LIVRO DE MINICURSOS SBCAS 2026

26º Simpósio Brasileiro de Computação Aplicada à Saúde

01 a 04 de junho de 2026
Ouro Preto, MG



Organização

Eduardo Luz
Rodrigo da Rosa Righi
Saul E. Delabrida Silva

UNISINOS
Universidade do Vale de Rio dos Sinos

UFOP
UNIVERSIDADE FEDERAL DE OURO PRETO

Realização

SBC
Sociedade Brasileira de Computação



SBCAS2026

26º Simpósio Brasileiro de Computação Aplicada à Saúde

Ouro Preto, MG, 01 a 04 de junho de 2026

Livro de Minicursos do XXVI Simpósio Brasileiro de Computação Aplicada à Saúde

Organização do Livro

Eduardo Luz (UFOP)

Rodrigo da Rosa Righi (UNISINOS)

Saul E. Delabrida Silva (UFOP)

Coordenação Geral do Simpósio

Andrea Gomes Campos (UFOP)

Realização

Universidade Federal de Ouro Preto – UFOP

Sociedade Brasileira de Computação – SBC

Porto Alegre

Sociedade Brasileira de Computação – SBC

2026

Esta obra está sob a licença Creative Commons Atribuição 4.0 (CC-BY). Você pode redistribuir este livro em qualquer suporte ou formato e copiar, remixar, transformar e criar a partir do conteúdo deste livro para qualquer fim, desde que cite a fonte.

Dados Internacionais de Catalogação na Publicação (CIP)

S612 Simpósio Brasileiro de Computação Aplicada à Saúde (26. : 01 – 04 junho 2026 : Porto Alegre)

Minicursos do SBCAS 2026 [recurso eletrônico] / organização: Eduardo Luz; Rodrigo da Rosa Righi; Saul E. Delabrida Silva. – Dados eletrônicos. – Porto Alegre: Sociedade Brasileira de Computação, 2026.

270 p. : il. : PDF ; 29 MB

Modo de acesso: World Wide Web.
ISBN 978-85-7669-670-4 (e-book)

1. Computação – Brasil – Evento. 2. Saúde aplicada. 3. Tecnologia em saúde. I. Luz, Eduardo. II. Righi, Rodrigo da Rosa. III. Silva, Saul E. Delabrida. IV. Sociedade Brasileira de Computação. V. Título.

CDU 004(063)

Ficha catalográfica elaborada por Annie Casali – CRB-10/2339

Biblioteca Digital da SBC – SBC OpenLib

Sociedade Brasileira de Computação
Av. Bento Gonçalves, 9500
Setor 4 | Prédio 43.412 | Sala 219 | Bairro Agronomia
Caixa Postal 15012 | CEP 91501-970
Porto Alegre - RS
(51) 99252-6018
sbc@sbc.org.br

Sociedade Brasileira de Computação – SBC

Presidência
Thais Vasconcelos Batista (UFRN), Presidente
Cristiano Maciel (UFMT), Vice-Presidente

Diretorias
Denis Lima do Rosário (UFPA), Diretor de Eventos e Comissões Especiais
Michelle Silva Wangham (UNIVALI), Diretora de Inovação
Alírio Santos de Sá (UFBA), Diretor de Comunicação
Eunice Pereira dos Santos Nunes (UFMT), Diretora de Secretarias Regionais
André Luís de Medeiros Santos (UFPE), Diretor de Planejamento e Programas Especiais
José Viterbo Filho (UFF), Diretor de Publicações
Ronaldo Alves Ferreira (UFMS), Diretor de Cooperação com Sociedades Científicas
Claudia Lage Rebello da Motta (UFRJ), Diretora de Educação
Leila Ribeiro (UFRGS), Diretora de Computação na Educação Básica
Renata de Matos Galante (UFRGS), Diretora Administrativa
Tanara Lauschner (UFAM), Diretora de Relações Profissionais
Francisco Dantas Medeiros Neto (UERN), Diretor de Finanças
Carlos Eduardo Ferreira (USP), Diretor de Competições Científicas

Diretorias Extraordinárias
Marcelo Antonio Marotta (UNB), Diretor de Tecnologia da Informação

Contato
Av. Bento Gonçalves, 9500
Setor 4 - Prédio 43.412 - Sala 219
Bairro Agronomia
91.509-900 – Porto Alegre RS

CNPJ: 29.532.264/0001-78
http://www.sbc.org.br

Comissão Especial de Computação Aplicada à Saúde (CE-CAS) - SBC

Rodrigo de Melo Souza Veras (UFPI) - Coordenador
Lina Maria Garcés Rodriguez (USP) - Vice-Coordenadora

Alexei Machado (UFMG)
Andrea Gomes Campos Bianchi (UFOP)
Cristiano André da Costa (Unisinos)
Débora Christina Muchaluat Saade (UFF)
Mariana Recamonde Mendoza (UFRGS)
Paulo Eduardo Ambrósio (UESC)
Rodrigo da Rosa Righi (Unisinos)
Sérgio Teixeira de Carvalho (UFG)

Mensagem da Coordenação Geral

É uma satisfação dar as boas-vindas aos leitores desta coletânea de minicursos do SBCAS 2026. Como coordenadora desta edição, é gratificante ver o simpósio consolidar-se em Ouro Preto como um espaço vital para a convergência entre o rigor científico e o impacto social. Este livro reflete o compromisso histórico do evento em fomentar a computação aplicada à saúde, trazendo temas que não apenas acompanham as tendências globais, mas que propõem soluções concretas para os desafios do ecossistema de saúde brasileiro.

Nesta edição, a seleção de tópicos demonstra o amadurecimento das nossas pesquisas, abrangendo desde a padronização de dados e aspectos de interoperabilidade até as fronteiras da Inteligência Artificial. Os leitores encontrarão aqui um mergulho profundo em tecnologias emergentes, como o Aprendizado Federado aplicado à triagem oncológica, a interpretabilidade de modelos com SHAP e o uso inovador de agentes inteligentes e aprendizado de máquina para a descoberta de fármacos. Cada capítulo foi estruturado para transitar da teoria à prática, garantindo que o conhecimento aqui compartilhado possa ser aplicado nos laboratórios e em aplicações reais nos setores hospitalar e industrial.

Gostaria de expressar meu sincero agradecimento aos autores e pesquisadores que dedicaram seu tempo a sistematizar esses conteúdos de forma didática e profunda. Em adição, aqui vão os meus agradecimentos aos professores que estiveram à frente da trilha de minicursos do SBCAS 2026. O SBCAS sempre foi movido pela colaboração e pela busca incessante pela inovação em prol do bem-estar humano. Espero que este material sirva como uma ferramenta valiosa para estudantes, acadêmicos e profissionais da área, inspirando novas investigações e parcerias que continuem a elevar o patamar da informática em saúde no Brasil. Tenham todos uma excelente leitura!

Andrea Gomes Campos (UFOP)

Coordenador Geral do SBCAS 2026

Mensagem da Coordenação de Minicursos

É com grande entusiasmo que damos as boas-vindas aos leitores deste volume, que reúne os minicursos selecionados para o XXVI Simpósio Brasileiro de Computação Aplicada à Saúde (SBCAS 2026). No dinâmico cenário da saúde digital, os minicursos desempenham um papel fundamental na formação de capital humano qualificado, servindo como ponte ágil entre a fundamentação teórica e a aplicação prática. Mais do que simples módulos de ensino, estas obras funcionam como catalisadores do conhecimento, permitindo que estudantes, pesquisadores e profissionais atualizem suas competências em um campo em que a inovação tecnológica avança em ritmo exponencial. Ao condensar experiências de especialistas em tópicos de alta relevância, este livro reafirma o compromisso do SBCAS com a democratização do saber e o fortalecimento da comunidade científica brasileira.

A seleção desta edição reflete fielmente os grandes temas que dominam a vanguarda tecnológica contemporânea, com foco robusto na Inteligência Artificial e na interoperabilidade de dados. O leitor encontrará discussões profundas sobre a construção de evidências através do modelo OMOP/OHDSI, o desenvolvimento de agentes de conversação e agentes inteligentes para o setor assistencial, e o uso de realidade virtual em treinamentos de saúde. Além disso, o volume aborda fronteiras críticas da ética e da eficiência algorítmica, como a interpretação de modelos com SHAP, o uso do Aprendizado Federado na triagem de câncer e o impacto do Aprendizado de Máquina na descoberta de novos fármacos. São temas que não apenas definem o "estado da arte", mas também propõem soluções concretas para os desafios reais enfrentados pelo sistema de saúde.

Por fim, este livro foi concebido como uma ferramenta versátil e valiosa para diferentes perfis de público. Para os estudantes, oferece um guia estruturado para ingressar em áreas de pesquisa emergentes; para os profissionais, apresenta metodologias e arquiteturas prontas para serem transpostas ao mercado e ao ambiente clínico; e para os entusiastas, oferece um panorama lúcido sobre como a computação está redefinindo o cuidado humano. Desejamos que estas páginas inspirem novas ideias e contribuam para o avanço da computação aplicada à saúde no Brasil, transformando o conhecimento técnico em impacto social e na melhoria da qualidade de vida da população. Boa leitura e um excelente simpósio!

Eduardo Luz (UFOP)

Rodrigo da Rosa Righi (UNISINOS)

Saul E. Delabrida Silva (UFOP)

Coordenadoras de Minicursos do SBCAS 2026

Sumário

Capítulo 1. OMOP/OHDSI Descomplicado: da Teoria à Construção de Evidências em Saúde.
Maria Tereza Fernandes Abrahão, Pablo Jorge Madril, Mateus de Lima Freitas. **1**

Capítulo 2. Interpretação de Modelos de Aprendizado de Máquina Aplicados à Saúde com SHAP.
Letícia Martins Raposo, Vinicius Navega Stelet. **42**

Capítulo 3. Construção de Agente de Conversação com Aplicação na Saúde: Do Problema Real à Solução Inteligente.
Amanda Morais Almeida, Diêgo Farias de Freitas, Paulo Eduardo Ambrósio **92**

Capítulo 4. Aprendizado Federado na Prática: Da Teoria à Implementação na Triagem do Câncer do Colo do Útero.
Leonardo Augusto Ferreira, Marcelo A. P. Xavier, Andrea G. Campos, Walmir M. Caminhas, Frederico Gadelha Guimarães. **134**

Capítulo 5. Agentes de Inteligência Artificial Aplicados à Saúde: Fundamentos, Arquiteturas e Implementação Prática.
James D. Sousa, Frank Cesár Lopes Véras. **183**

Capítulo 6. Aprendizado de Máquina Aplicado à Descoberta de Fármacos.
Karina dos Santos Machado, Adriano Velasque Werhli, Frederico Kremer, Rafael Junqueira Borges. **219**

Capítulo 1

OMOP/OHDSI Descomplicado: da Teoria à Construção de Evidências em Saúde

Maria Tereza Fernandes Abrahão, Pablo Jorge Madril, Mateus de Lima Freitas, Marcos Silva de Mendonça

Abstract

This chapter presents the growing role of observational research based on real-world data in the generation of reliable health evidence. In this context, the OMOP/OHDSI ecosystem is described as a well-established international infrastructure for the standardization, harmonization, and analysis of observational data from multiple sources. The text addresses the concepts of Real-World Data (RWD) and Real-World Evidence (RWE), as well as the scientific principles underpinning OHDSI. It presents the structure of the OMOP Common Data Model, including standardized tables and vocabularies, and outlines the implementation workflow with a focus on the structural and semantic data mapping processes. Also, it explores the use of Artificial Intelligence to support the definition and validation of clinical phenotypes. Finally, it discusses the impact of OMOP/OHDSI on multicenter research, the generation of real-world evidence, and the advancement of digital health.

Resumo

Este capítulo apresenta o papel crescente da pesquisa observacional baseada em dados do mundo real na geração de evidências confiáveis em saúde. Nesse contexto, descreve-se o ecossistema OMOP/OHDSI como uma infraestrutura internacional consolidada para a padronização, harmonização e análise de dados observacionais de múltiplas fontes. O texto aborda os conceitos de Real-World Data (RWD) e Real-World Evidence (RWE), bem como os princípios científicos da OHDSI. Apresenta-se a estrutura

 ©2026 SBC – Soc. Bras. de Computação

do OMOP Common Data Model, incluindo tabelas e vocabulários padronizados, além do fluxo de implementação com foco no mapeamento estrutural e semântico dos dados. Também explora o uso da Inteligência Artificial no apoio à definição e validação de fenótipos clínicos. Por fim, discute o impacto do OMOP/OHDSI na pesquisa multicêntrica, na produção de evidências do mundo real e no avanço da saúde digital.

1. Introdução

A pesquisa observacional em saúde vem ganhando destaque como uma abordagem essencial para a geração de evidências em larga escala, especialmente em contextos em que ensaios clínicos randomizados não são viáveis, éticos ou suficientes para responder às perguntas científicas necessárias. Esse tipo de pesquisa permite explorar dados do mundo real coletados rotineiramente em sistemas de saúde, favorecendo análises reprodutíveis, comparáveis e capazes de refletir cenários clínicos complexos e heterogêneos.

Nesse contexto, o ecossistema OMOP/OHDSI (*Observational Medical Outcomes Partnership / Observational Health Data Sciences and Informatics*) consolidou-se como o padrão global para a harmonização de dados clínicos e execução de estudos multicêntricos em saúde. O modelo de dados comum OMOP (CDM OMOP) oferece uma estrutura unificada que padroniza informações provenientes de diversas fontes, como prontuários eletrônicos, sistemas administrativos, registros farmacêuticos e bancos de dados de pesquisa, permitindo que instituições ao redor do mundo alinhem seus dados de forma consistente.

A comunidade OHDSI complementa essa infraestrutura ao disponibilizar metodologias, ferramentas analíticas e uma rede colaborativa internacional que possibilita a realização de estudos em larga escala, de maneira distribuída e tecnicamente robusta. Isso promove a comparabilidade entre instituições e países, fortalece a reprodutibilidade científica e acelera o processo de geração de insights clínicos, atuando como um sistema em que em cada etapa possui mecanismos de validação *a priori*, que garante a qualidade da pesquisa.

Nestes mecanismos de validação temos abertura para a incorporação de tecnologias de Inteligência Artificial e Deep Learning que permitem escalar o processo, sem interferir com os resultados, criando uma forma de adoção destas tecnologias com baixo risco.

Como resultado, o ecossistema OMOP/OHDSI tem desempenhado papel central na evolução da pesquisa observacional moderna, ampliando a capacidade de transformar dados em conhecimento e contribuindo diretamente para avanços em vigilância epidemiológica, farmacovigilância, predição de riscos e avaliação de efetividade de intervenções no mundo real.

O capítulo está estruturado como se segue. Primeiro, a seção 1.1 oferece uma introdução à pesquisa observacional, aos conceitos básicos de RWD e RWE. A seção 1.2 apresenta a iniciativa OHDSI, sua origem e os princípios científicos e a seção 1.3 o

 ©2026 SBC – Soc. Bras. de Computação

ecossistema OMOP/OHDSI. A seção 1.4 apresenta o modelo comum de dados, sua estrutura, tabelas e vocabulários e a seção 1.5 o fluxo de implementação para a preparação da base OMOP apresentando os mapeamentos dos dados, dos vocabulários e a ferramenta de Inteligência Artificial para detecção de fenótipos. A seção 1.6 apresenta as considerações finais e conclusões e a seção 1.7 as referências consultadas.

Na elaboração deste capítulo, consultou-se o livro da OHDSI, *The book of OHDSI*, de domínio público, sob a licença *Creative Commons Zero v1.0 Universal,* (16/04/2020). O livro é um documento vivo, mantido pela comunidade por meio de ferramentas de desenvolvimento de código aberto e evolui continuamente. A versão online, disponível gratuitamente[1], sempre representa a versão mais recente. Os exemplos foram elaborados a partir dos conhecimentos adquiridos pelos autores.

1.1 Introdução a pesquisa observacional

A pesquisa observacional é uma abordagem central para a compreensão de fenômenos em saúde a partir do uso de dados do mundo real, analisando informações tal como são geradas rotineiramente nos serviços de saúde, sem intervenção direta dos pesquisadores sobre a exposição ou o tratamento dos indivíduos. Diferentemente dos ensaios clínicos randomizados, que seguem protocolos rígidos e operam em ambientes controlados, a pesquisa observacional permite capturar a complexidade, a diversidade populacional e as condições reais da prática clínica, oferecendo uma visão mais próxima do cotidiano dos sistemas de saúde. Por essa razão, ela desempenha um papel estratégico na produção de evidências científicas, especialmente em cenários nos quais estudos experimentais seriam inviáveis, excessivamente onerosos ou eticamente inadequados (Hernán & Robins, 2025; Ricotta, EE, et al, 2025).

No contexto atual, marcado pela crescente digitalização dos sistemas de saúde, a disponibilidade de grandes volumes de dados estruturados abriu novas possibilidades para a pesquisa observacional. Esses dados, conhecidos como Real World Data (RWD), incluem informações provenientes de prontuários eletrônicos, registros administrativos, sistemas de faturamento, bancos de dados farmacêuticos, registros de doenças, entre outras fontes, que refletem a utilização real dos serviços de saúde e o comportamento clínico de grandes populações ao longo do tempo.

O uso sistemático desses dados permite avaliar desfechos clínicos em populações amplas e diversas, acompanhar trajetórias de cuidado ao longo do tempo e identificar padrões que dificilmente seriam observados em estudos altamente controlados.

A análise sistemática e metodologicamente rigorosa do RWD possibilita a geração de Real World Evidence (RWE), definida como a evidência clínica obtida a partir da análise rigorosa desses dados, contribuindo para avaliações de segurança, efetividade e uso de intervenções no mundo real (FDA, 2023; FDA, 2025).

Os estudos observacionais assumem diferentes delineamentos, como estudos de coorte, caso-controle, transversais e séries temporais. Cada um desses formatos responde

[1] The book OHDSI http://book.ohdsi.org

 ©2026 SBC – Soc. Bras. de Computação

a tipos específicos de perguntas cientificas, permitindo investigar associações entre exposições e desfechos, acompanhar trajetórias de cuidado e avaliar variações clínicas e assistenciais. Embora não tenham como principal objetivo estabelecer causalidade nos mesmos moldes dos ensaios clínicos randomizados, estudos observacionais bem conduzidos são essenciais para complementar evidências clínicas e orientar decisões em saúde pública, avaliação de tecnologias em saúde e farmacovigilância, especialmente quando aliados a métodos analíticos avançados e boas práticas metodológicas (Hernán & Robins, 2025; Sterrantino, 2024).

Entretanto, a pesquisa observacional está sujeita a uma série de vieses metodológicos que podem comprometer a validade interna e externa dos resultados caso não sejam devidamente identificados, controlados e discutidos. Entre os principais vieses, destaca-se o viés de seleção, que ocorre quando a população analisada não é representativa da população alvo, podendo distorcer associações observadas. O viés de informação refere-se a erros sistemáticos ou inconsistências no registro, mensuração ou classificação das exposições, desfechos ou covariáveis. Já o viés de confusão ocorre quando fatores externos estão associados simultaneamente à exposição e ao desfecho, levando a interpretações causais equivocadas se não forem adequadamente ajustados.

Adicionalmente, problemas recorrentes como dados ausentes, heterogeneidade na qualidade das fontes, mudanças nos sistemas de informação ao longo do tempo e variações nas práticas clínicas representam desafios frequentes em estudos baseados em dados do mundo real. O reconhecimento explícito dessas limitações é essencial tanto para a interpretação crítica dos achados quanto para o desenho de estudos observacionais mais robustos e transparentes, em consonância com as boas práticas metodológicas em epidemiologia e ciência de dados em saúde (Porta et al., 2014; Dekkers et al., 2012).

Para mitigar tais limitações, a adoção de boas práticas em pesquisa observacional é indispensável. Essas práticas incluem a definição clara da pergunta científica, a especificação prévia do protocolo analítico, o uso de métodos estatísticos apropriados para controle de confundimento, a transparência metodológica e a reprodutibilidade das análises. Iniciativas internacionais, como a declaração STROBE, fornecem diretrizes consolidadas para melhorar a qualidade e a transparência no relato de estudos observacionais, fortalecendo a credibilidade das evidências geradas (STROBE Initiative). A documentação completa das decisões metodológicas e a reprodutibilidade dos resultados são princípios centrais para garantir a robustez científica dos estudos baseados em RWD.

Em síntese, a pesquisa observacional desempenha um papel estratégico no uso de dados do mundo real para a geração de evidências relevantes, complementando os ensaios clínicos e ampliando a compreensão da efetividade das intervenções na prática cotidiana.

O uso adequado de RWD para gerar RWE, aliado à atenção aos vieses, à adoção de boas práticas metodológicas e à padronização dos dados, permite ampliar a capacidade de compreender fenômenos complexos, avaliar intervenções no mundo real e apoiar decisões mais informadas. Dessa forma, a pesquisa observacional consolida-se como um instrumento indispensável para o avanço do conhecimento, a melhoria da qualidade do

 ©2026 SBC – Soc. Bras. de Computação

cuidado e o fortalecimento dos sistemas de saúde, constituindo-se como a base para uma ciência mais reprodutível, colaborativa.

Nesse cenário, a padronização de dados em saúde emerge como um fator crítico para a qualidade, a comparabilidade e a escalabilidade da pesquisa observacional. Sistemas de saúde frequentemente utilizam diferentes estruturas de dados, formatos e terminologias clínicas, o que dificulta a integração de múltiplas fontes, compromete a reprodutibilidade dos estudos e limita a realização de análises multicêntricas. A ausência de padrões comuns aumenta o risco de erros analíticos, dificulta a interoperabilidade entre instituições e reduz a confiabilidade dos achados científicos. Em contrapartida, a adoção de modelos padronizados de dados, associada ao uso de vocabulários clínicos comuns e definições semânticas consistentes, contribui para maior qualidade analítica, facilita o compartilhamento de informações, estimula a colaboração entre pesquisadores e amplia o potencial de reutilização dos dados para diferentes finalidades de pesquisa e geração de evidências do mundo real (McDonald C.J. et al, 2022; Kahn et al., 2016; Hripcsak et al., 2015).

Além disso, a padronização é um pilar fundamental para a ciência aberta e colaborativa, permitindo que métodos e análises sejam replicados em diferentes contextos, sem a necessidade de compartilhamento direto de dados sensíveis.

Ao alinhar dados de diferentes origens sob uma mesma estrutura conceitual, torna-se possível executar análises distribuídas, comparar resultados entre instituições e replicar estudos em diferentes populações, preservando a privacidade dos dados. Isso promove estudos distribuídos, comparabilidade internacional e maior confiança nas evidências produzidas. Dessa forma, a padronização não apenas fortalece a pesquisa observacional do ponto de vista técnico, mas também acelera a produção de conhecimento científico com impacto direto na prática clínica e na formulação de políticas públicas (FDA, 2025).

1.2 A Iniciativa OHDSI: Origem e Princípios Científicos

A iniciativa Observational Health Data Sciences and Informatics (OHDSI) tem sua origem diretamente relacionada aos esforços conduzidos no âmbito do Observational Medical Outcomes Partnership (OMOP), um projeto iniciado formalmente em 2008 como uma parceria público-privada coordenada pela Foundation for the National Institutes of Health (FNIH), com forte participação da Food and Drug Administration (FDA), da indústria farmacêutica, da academia e de organizações de saúde (Overhage et al., 2012; Hripcsak et al., 2015). O surgimento do OMOP ocorreu em um contexto regulatório específico, marcado pela promulgação do FDA Amendments Act de 2007, que ampliou as exigências por sistemas mais eficazes de monitoramento pós comercialização de medicamentos. O projeto OMOP foi concebido com duração aproximada de cinco anos (2008–2013) e teve como objetivo central avaliar empiricamente a confiabilidade de métodos analíticos aplicados a dados observacionais em saúde, especialmente no contexto da farmacovigilância (Stang et al., 2010; Overhage et al., 2012). Diferentemente de iniciativas orientadas à obtenção direta de resultados clínicos, o OMOP concentrou seus esforços na comparação sistemática de abordagens

 ©2026 SBC – Soc. Bras. de Computação

metodológicas, examinando sua capacidade de identificar associações causais verdadeiras e minimizar falsas descobertas em bases de dados heterogêneas.

Uma das principais contribuições científicas decorrentes do OMOP foi o desenvolvimento do OMOP Common Data Model (CDM), criado para padronizar a estrutura e o conteúdo de dados observacionais provenientes de diferentes fontes, como prontuários eletrônicos e bases administrativas. A adoção de um modelo de dados comum e dos vocabulários padronizados permitiu a execução de análises comparáveis em múltiplas bases de dados, reduzindo a variabilidade estrutural e semântica e aumentando a reprodutibilidade dos estudos (Hripcsak et al., 2015). https://www.ohdsi.org/data-standardization/

Com o encerramento formal do projeto OMOP em 2013, seus principais ativos científicos, incluindo o Common Data Model, os vocabulários padronizados e os métodos analíticos desenvolvidos, passaram a ser mantidos e ampliados por uma nova iniciativa colaborativa: o OHDSI (Hripcsak et al., 2015). Diferentemente do OMOP, estruturado como um projeto com prazo definido, a OHDSI foi concebida como uma iniciativa contínua, aberta e global, orientada à evolução permanente de métodos e ferramentas para análise de dados observacionais em saúde.

Desde sua constituição, a OHDSI adotou como princípio estruturante a separação entre dados e métodos, promovendo um modelo no qual as instituições participantes mantêm seus dados sob custódia local, enquanto compartilham definições de coorte, protocolos analíticos e código-fonte (Hripcsak et al., 2015). Tal abordagem viabiliza a replicação sistemática de estudos em diferentes contextos assistenciais, fortalecendo a reprodutibilidade científica e favorecendo a validação externa dos resultados obtidos. Entre os princípios científicos fundamentais da iniciativa, destaca-se a transparência metodológica, que se manifesta na exigência de documentação explícita de hipóteses, critérios de inclusão e exclusão, estratégias analíticas e suposições estatísticas.

A OHDSI também reconhece explicitamente as limitações inerentes aos dados observacionais, promovendo o controle sistemático de vieses e fatores de confusão por meio de desenhos de estudo rigorosos e metodologias estatísticas avançadas (Schuemie et al., 2014).

Outro pilar conceitual da OHDSI é o paradigma de análise distribuída, no qual apenas o código analítico é compartilhado entre instituições, enquanto os dados permanecem localmente armazenados (Hripcsak et al., 2015). Esse modelo responde diretamente às exigências éticas e legais associadas à proteção de dados em saúde, permitindo a realização de estudos multicêntricos internacionais em conformidade com legislações como o General Data Protection Regulation (GDPR) e a Lei Geral de Proteção de Dados (LGPD), sem necessidade de transferência de dados sensíveis.

Sob uma perspectiva científico-metodológica, a iniciativa OHDSI desempenha um papel central na reconfiguração da pesquisa observacional, ao afirmá-la como uma fonte legítima, rigorosa e complementar de evidência clínica e populacional. Ao articular padronização estrutural e semântica dos dados, métodos analíticos explicitamente documentados e práticas de colaboração científica em larga escala, a OHDSI contribui

 ©2026 SBC – Soc. Bras. de Computação

para mitigar limitações historicamente atribuídas aos estudos observacionais, como baixa reprodutibilidade e elevado risco de vieses. Dessa forma, a iniciativa reduz a assimetria tradicional entre estudos observacionais e ensaios clínicos randomizados, não por equipará-los indiscriminadamente, mas por posicionar cada abordagem de maneira contextualizada, especialmente em cenários nos quais os ensaios randomizados são inviáveis, eticamente limitados ou insuficientes para representar a complexidade e a heterogeneidade da prática clínica real (Hripcsak et al., 2015).

Em síntese, a transição do OMOP (2008–2013) para o OHDSI representa a passagem de um projeto metodológico delimitado para um ecossistema científico sustentado, fundamentado em princípios de ciência aberta, rigor metodológico e colaboração global. Essa evolução consolidou as bases técnicas e conceituais para a produção contínua de evidências em saúde a partir de dados observacionais em escala populacional.

Ao estabelecer uma comunidade global estruturada de pesquisadores, instituições acadêmicas e bases de dados observacionais em saúde, com um centro de coordenação científica sediado na Universidade de Columbia, a iniciativa OHDSI viabilizou a colaboração científica em escala verdadeiramente internacional (Hripcsak et al., 2015). Atualmente, a comunidade OHDSI reúne mais de 4.700 colaboradores distribuídos por aproximadamente 88 países, integrando uma rede de mais de 540 fontes de dados padronizadas, que representam cerca de 970 milhões de registros de pacientes únicos mapeados ao OMOP Common Data Model (cerca de 12% da população mundial) (Columbia DBMI, 2025; OHDSI, 2026). Esse arranjo colaborativo e distribuído permite a execução de estudos observacionais multicêntricos, reprodutíveis e metodologicamente consistentes, transcendendo fronteiras geográficas e institucionais. Ao promover ciência aberta, padronização analítica e validação cruzada em muitos contextos assistenciais, a OHDSI vem transformando a forma como a pesquisa médica é conduzida, com impacto direto na geração de evidências para apoiar decisões clínicas, regulatórias e de políticas públicas, e com o objetivo explícito de contribuir para a melhoria da saúde em escala global.

1.3 Ecossistema OMOP/OHDSI

A interoperabilidade entre sistemas de informação em saúde é um requisito fundamental para o uso integrado e eficiente de dados provenientes de múltiplas fontes. Segundo Mucheroni (Mucheroni 2011), essa interoperabilidade depende essencialmente de dois fatores complementares:

- Interoperabilidade sintática (forma)
- Interoperabilidade semântica (conteúdo)

A interoperabilidade sintática refere-se à padronização da estrutura, formatos e modelos de dados, garantindo que diferentes sistemas consigam trocar informações de maneira técnica e operacional. Já a interoperabilidade semântica assegura que o significado dos dados seja preservado e compreendido de forma consistente entre os sistemas, permitindo interpretações corretas e comparáveis das informações trocadas.

 ©2026 SBC – Soc. Bras. de Computação

No contexto da pesquisa em saúde baseada em dados observacionais, esses dois níveis de interoperabilidade são indispensáveis para possibilitar comparações, integrações e análises estatísticas robustas entre conjuntos de dados heterogêneos, oriundos de diferentes instituições, regiões ou países.

Para viabilizar análises reprodutíveis e em larga escala de dados observacionais em saúde, a OHDSI se fundamenta em três componentes principais:

Modelo Comum de Dados (Common Data Model - CDM OMOP) O CDM OMOP define uma estrutura padronizada para o armazenamento de dados clínicos e administrativos, promovendo interoperabilidade sintática ao uniformizar tabelas, campos e relacionamentos. A especificação completa do modelo, incluindo a descrição detalhada das tabelas, campos, regras de preenchimento e versões suportadas, está disponível na documentação oficial da comunidade OHDSI: https://ohdsi.github.io/CommonDataModel/index.html

Vocabulários Padronizados - O uso de vocabulários controlados e padronizados (como SNOMED, LOINC, RxNorm, entre outros) assegura a interoperabilidade semântica, permitindo que os mesmos conceitos clínicos sejam representados de forma consistente, independentemente da origem dos dados. No ecossistema OMOP/OHDSI, esses vocabulários são integrados ao modelo por meio de um mecanismo unificado de conceitos e relacionamentos, permitindo a padronização semântica dos dados observacionais. Os vocabulários oficiais utilizados pelo CDM OMOP podem ser obtidos e atualizados por meio da plataforma Athena, disponível em: https://athena.ohdsi.org

Ferramentas para preparação do CDM e análises - A OHDSI disponibiliza um ecossistema de ferramentas que apoia desde o processo de transformação dos dados fonte para o CDM OMOP até a execução de análises estatísticas avançadas e a definição padronizada de estudos observacionais.

As ferramentas disponibilizadas pela OHDSI operam de maneira coesa e padronizada, permitindo que análises no nível do paciente sejam conduzidas de forma distribuída, transparente e reprodutível. Esse conjunto integrado de soluções viabiliza a realização de estudos comparáveis entre diferentes bases de dados, sem a necessidade de compartilhamento direto dos dados sensíveis.

Os principais pontos em destaque são:

- Padronização estrutural e semântica, reduzindo ambiguidades e aumentando a qualidade analítica dos dados.
- Reprodutibilidade científica, uma vez que métodos e códigos analíticos podem ser reutilizados em diferentes bases convertidas para o CDM OMOP.
- Escalabilidade e colaboração internacional, permitindo estudos multicêntricos com grande volume de dados.
- Transparência metodológica, com definição clara e compartilhável de coortes, critérios de inclusão e métodos estatísticos.
- Aceleração da geração de evidências do mundo real (Real World Evidence), apoiando decisões clínicas, regulatórias e estratégicas em saúde.

©2026 SBC – Soc. Bras. de Computação

No geral, as ferramentas da OHDSI desempenham um papel crucial na promoção da colaboração, reprodutibilidade e transparência na análise de dados observacionais de saúde, permitindo avanços significativos na pesquisa e na prática clínica. O conjunto de ferramentas disponibilizadas pela OHDSI auxiliam a preparação da base no modelo CDM OMOP, na validação do processo de mapeamento, na verificação da qualidade dos dados que compõem o CDM, na elaboração e análise dos diferentes tipos de estudos, facilitando a exploração dos dados e a geração de evidências (Hripcsak et al., 2015; Kahn et al., 2016). Podemos citar:

- Ferramentas para geração do banco CDM OMOP
  - ETL: White Rabbit, Rabbit-In-A-Hat[2]
  - Vocabulários: Athena[3]/Usagi[4]
  - Qualidade: Achilles[5] e *Data Quality Dashboard* (DQD)[6]
- Ferramentas de análise
  - Geração de Coortes/estudos - Atlas[7]
  - Análises estatísticas: HADES[8]

1.3.1 Onde achar: principais referências

As informações a respeito dos componentes da OHDSI podem ser classificadas em:

- ➢ Informações gerais: http://www.ohdsi.org - Site principal;
- ➢ Código e instalações: https://github.com/OHDSI/ - Aqui está disponibilizado o código fonte de todas as ferramentas. Em particular destacamos: *Common Data Model* (https://github.com/OHDSI/CommonDataModel), com a definição completa do modelo e as implementações para os diversos bancos suportados;
- ➢ Tutoriais e vídeos: Procure no Google por YouTube OHDSI (https://www.google.com/search?q=youtube+ohdsi), existe muita documentação e tutoriais em vídeo dos eventos anuais do grupo;
- ➢ Fórum: Para resolver dúvidas mais frequentes, consulte (http://forums.ohdsi.org/).
- ➢ Livro OHDSI: The book of OHDSI

1.4 Common Data Model: estrutura, tabelas essenciais e vocabulários

O OMOP Common Data Model (CDM) constitui a infraestrutura técnica fundamental que sustenta análises reprodutíveis e metodologicamente consistentes sobre dados do mundo real. Seu desenvolvimento responde diretamente à necessidade de lidar com a heterogeneidade estrutural e semântica dos dados de saúde, criando uma base comum que

[2] White Rabbit, Rabbit-In-A-Hat http://ohdsi.github.io/WhiteRabbit/index.html
[3] ATHENA http://athena.ohdsi.org
[4] Usagi http://ohdsi.github.io/Usagi/
[5] ACHILLES http://www.ohdsi.org/web/achilles
[6] DQD https://ohdsi.github.io/DataQualityDashboard/articles/DataQualityDashboard.html
[7] Atlas https://github.com/OHDSI/Atlas/wiki/Atlas-Setup-Guide
[8] HADES https://github.com/OHDSI/Hades

©2026 SBC – Soc. Bras. de Computação

permita comparabilidade analítica entre diferentes sistemas, instituições e contextos geográficos.

O CDM OMOP foi concebido com um objetivo central: permitir que dados observacionais de saúde, provenientes de fontes diversas como prontuários eletrônicos, bases administrativas e de faturamento, registros farmacêuticos, sistemas laboratoriais e registros de mortalidade, sejam analisados de forma padronizada por meio da aplicação de métodos analíticos comuns. Diferentemente de modelos voltados à interoperabilidade clínica em tempo real ou à troca transacional de informações, o CDM OMOP é explicitamente orientado à análise retrospectiva e longitudinal de dados populacionais. Ao padronizar simultaneamente a estrutura e o significado dos dados, o modelo possibilita que uma mesma pergunta científica seja investigada em diferentes contextos institucionais e sistemas de saúde, utilizando protocolos analíticos idênticos.

O desenho do CDM OMOP reflete princípios que atendem tanto a requisitos técnicos quanto a exigências científicas. O modelo adota padronização estrutural rigorosa, por meio da definição de esquemas de tabelas, campos e relacionamentos uniformes, o que permite a reutilização de códigos analíticos sem necessidade de adaptações específicas para cada base de dados. Paralelamente, incorpora padronização semântica mediante o uso de vocabulários clínicos internacionais, assegurando que conceitos médicos mantenham significado consistente entre diferentes instituições e países. Um princípio adicional fundamental é a separação entre a estrutura do modelo e o conteúdo semântico, característica que confere flexibilidade e permite a evolução controlada do modelo ao longo do tempo, sem comprometer análises previamente implementadas. O CDM OMOP é fortemente orientado à preservação da dimensão temporal dos eventos clínicos, viabilizando análises longitudinais e a reconstrução das trajetórias assistenciais completas dos indivíduos.

A arquitetura do CDM OMOP baseia-se em um modelo relacional organizado em domínios clínicos bem definidos, nos quais cada domínio representa um tipo específico de evento ou informação em saúde. Essa organização contempla informações demográficas dos indivíduos, períodos de observação, atendimentos ambulatoriais e internações, diagnósticos e condições clínicas, exposição a medicamentos, procedimentos clínicos e cirúrgicos, exames laboratoriais, medições clínicas e registros de óbito. Tal estrutura suporta tanto análises populacionais de grande escala quanto estudos em nível individual, mantendo coerência clínica e temporal dos dados.

A estrutura prática do CDM OMOP foi concebida para padronizar dados de saúde de diferentes origens, permitindo análises consistentes, reprodutíveis e comparáveis entre instituições, países e sistemas distintos. Uma vez que os dados são transformados e carregados no modelo OMOP, passam a obedecer a uma lógica estrutural muito bem definida, voltada principalmente para a análise clínica e epidemiológica em larga escala.

Estrutura centrada no paciente

O princípio fundamental do CDM OMOP é ser patient-centric. Isso significa que o paciente é o eixo central de toda a modelagem de dados. Todas as tabelas clínicas

 ©2026 SBC – Soc. Bras. de Computação

possuem relacionamento direto ou indireto com a tabela PERSON, por meio do campo PERSON_ID, que identifica de forma única cada indivíduo no banco de dados.
Essa abordagem permite reconstruir a linha do tempo clínica do paciente, organizando eventos como visitas, diagnósticos, tratamentos e medições de forma cronológica. Datas de início e fim (por exemplo, *start_date* e *end_date*) são elementos essenciais do modelo, garantindo análises longitudinais, estudos de trajetória de cuidado, identificação de recorrência de eventos e avaliação de desfechos ao longo do tempo.

Organização temporal dos eventos clínicos
No CDM OMOP, eventos não são registros isolados: eles fazem parte de uma sequência temporal. Cada condição diagnosticada, medicamento administrado ou exame realizado está associado a:

- Um paciente (*PERSON_ID*),
- Um período no tempo,
- Um contexto assistencial, geralmente representado por uma **visita** (*VISIT_OCCURRENCE*).

Essa estrutura possibilita análises como:

- Progressão de doenças,
- Comparação entre períodos pré e pós-intervenção,
- Identificação de padrões de uso de medicamentos,
- Avaliação de respostas clínicas a tratamentos.

Principais tabelas clínicas
PERSON: A tabela PERSON armazena as informações demográficas básicas do indivíduo, como sexo, data de nascimento, data de óbito (quando aplicável), raça e etnia, sempre codificadas por meio de vocabulários padronizados. Essa padronização é essencial para estudos populacionais e comparações entre bases distintas.
VISIT_OCCURRENCE: representa os contatos do paciente com o sistema de saúde, como internações, atendimentos ambulatoriais, consultas de emergência ou teleatendimentos. Ela funciona como um elemento organizador dos demais eventos clínicos, permitindo que diagnósticos, procedimentos, medicamentos e exames sejam associados a um mesmo episódio de cuidado.
CONDITION_OCCURRENCE: registra os diagnósticos e condições clínicas identificadas no paciente. Cada registro indica quando a condição foi observada, o tipo de diagnóstico (principal, secundário, histórico, suspeito, confirmado) e, sempre que possível, o contexto da visita. As condições são codificadas com terminologias padronizadas, como SNOMED CT, o que aumenta a interoperabilidade e a precisão analítica.
DRUG_EXPOSURE: descreve a exposição do paciente a medicamentos, incluindo prescrições, dispensações ou administrações. Essa tabela permite representar informações como princípio ativo, dose, duração do tratamento e via de administração. Ela é

 ©2026 SBC – Soc. Bras. de Computação

fundamental para análises de segurança medicamentosa, adesão terapêutica, estudos de efetividade e farmacovigilância.

MEASUREMENT: concentra dados quantitativos e qualitativos provenientes de exames laboratoriais, sinais vitais e outros testes clínicos. Além do valor medido, são armazenados unidades, métodos, intervalos de referência e datas, possibilitando avaliações clínicas detalhadas, análises de tendência e estratificação de risco.

OBSERVATION: utilizada para registrar achados clínicos diversos que não se enquadram diretamente em diagnósticos, medicamentos ou medições. Exemplos incluem hábitos de vida, fatores sociais, respostas a questionários, escala de dor ou informações clínicas observacionais relevantes. Essa tabela amplia a capacidade analítica do modelo, incorporando dados contextuais e complementares.

A figura a 1.1 ilustra as principais tabelas clínicas do modelo OMOP.

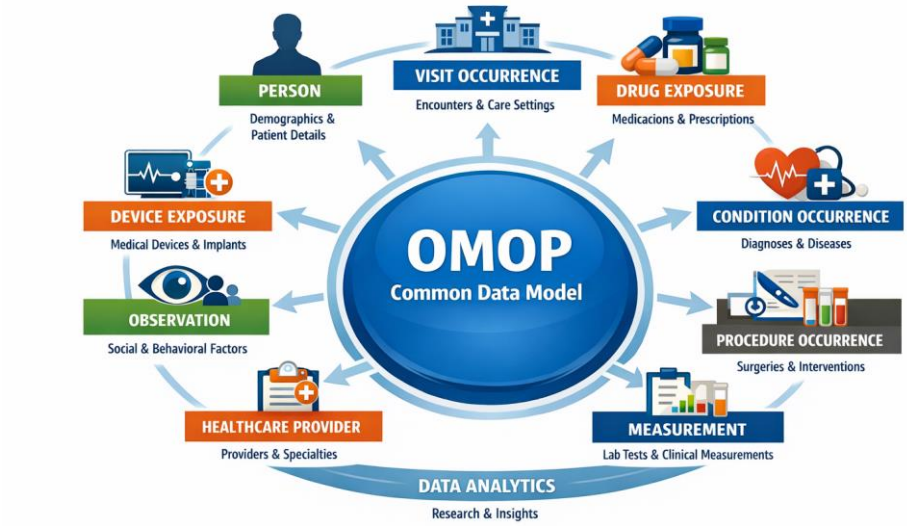


Figura 1.1 Visão geral das principais tabelas clínicas do CDM OMOP

Padronização e escalabilidade

Em conjunto, a estrutura do CDM OMOP oferece uma base sólida, coerente e orientada ao paciente, preparada para suportar análises descritivas, preditivas e causais, sendo amplamente utilizada em projetos de saúde digital, real world evidence e pesquisa clínica avançada.

Um dos aspectos mais distintivos do CDM OMOP é o suporte explícito à interoperabilidade semântica por meio do uso de vocabulários clínicos padronizados, integrados no chamado OMOP Standardized Vocabulary. Durante o processo de conversão dos dados, códigos locais utilizados pelos sistemas de origem são mapeados para conceitos padronizados, ao mesmo tempo em que os códigos originais são preservados. Essa estratégia garante rastreabilidade, flexibilidade analítica e comparabilidade internacional, promovendo harmonização semântica mesmo em ambientes altamente heterogêneos.

 ©2026 SBC – Soc. Bras. de Computação

A adoção do CDM OMOP envolve um processo estruturado de extração, transformação e carga (ETL) dos dados de origem, que inclui a análise do modelo original, o mapeamento estrutural para o CDM, o mapeamento semântico para vocabulários padronizados, a validação técnica e clínica dos dados transformados e a avaliação sistemática da qualidade dos dados. Embora esse processo seja tecnicamente complexo e demande conhecimento especializado, ele representa um investimento estratégico, pois viabiliza a reutilização contínua dos dados para múltiplos estudos, sem necessidade de reestruturações adicionais.

Um dos impactos mais relevantes do CDM OMOP na pesquisa observacional reside em sua contribuição para a reprodutibilidade científica. Ao garantir que bases de dados distintas compartilhem a mesma estrutura e semântica, o modelo permite que protocolos analíticos padronizados sejam executados em diferentes contextos, produzindo resultados diretamente comparáveis. Essa característica é particularmente importante para estudos multicêntricos internacionais, validação externa de achados científicos, análises distribuídas em ambientes regulados e auditorias metodológicas, posicionando o CDM OMOP como uma ponte entre dados fragmentados e práticas científicas sistematizadas.

Apesar de seus benefícios amplamente reconhecidos, o CDM OMOP apresenta desafios relevantes. A complexidade do processo de ETL, a necessidade de governança robusta de dados e a dependência da qualidade dos dados de origem são limitações frequentemente discutidas na literatura. Além disso, a adoção institucional do modelo exige maturidade organizacional, capacitação técnica e compromisso estratégico de longo prazo. Esses desafios, entretanto, não diminuem o valor científico do modelo, mas evidenciam que sua implementação bem-sucedida depende da integração entre fatores técnicos, humanos e institucionais.

Ao fornecer dados longitudinalmente estruturados e semanticamente consistentes, o CDM OMOP também se consolida como uma base sólida para inovação analítica em saúde digital. Suas aplicações incluem modelos preditivos clínicos, aprendizado de máquina e inteligência artificial, avaliação de tecnologias em saúde, estudos de efetividade no mundo real, saúde populacional e iniciativas de medicina personalizada. Dessa forma, o CDM OMOP transcende sua função original como modelo de dados e se afirma como uma infraestrutura estratégica para a pesquisa observacional baseada em dados e evidências. Na prática, CDM OMOP **f**unciona como uma *camada de padronização* que transforma dados de saúde heterogêneos em uma estrutura comum, permitindo análises comparáveis, reprodutíveis e em larga escala.

Em síntese, o CDM OMOP representa um marco na evolução da pesquisa observacional em saúde. Ao integrar padronização estrutural, interoperabilidade semântica e suporte metodológico, o modelo viabiliza análises reprodutíveis, escaláveis e cientificamente rigorosas sobre dados do mundo real. Inserido no ecossistema OHDSI, o CDM OMOP não apenas enfrenta os desafios técnicos da heterogeneidade de dados, mas contribui para a redefinição de práticas científicas, promovendo colaboração global,

 ©2026 SBC – Soc. Bras. de Computação

transparência e geração consistente de evidências, reforçando sua relevância estratégica para o presente e o futuro da ciência em saúde.

O ponto de partida para iniciativas de análise de dados em saúde é, em geral, um cenário caracterizado por dados de fonte heterogêneas. As organizações de saúde armazenam informações clínicas e administrativas em diversos sistemas distintos, como prontuários eletrônicos do paciente (EHR/PEP), sistemas administrativos e de faturamento, registros laboratoriais, bases de dados de farmácia e bancos de sinistros.

Esses diferentes repositórios refletem a diversidade de objetivos operacionais e contextos institucionais, resultando em dados que variam significativamente quanto à estrutura, incluindo tabelas, campos e relacionamentos, bem como quanto à codificação, que pode envolver classificações padronizadas como CID-10, códigos locais ou mesmo texto livre. Além disso, há variações importantes em termos de granularidade, completude e qualidade da informação.

Nesse contexto, o CDM OMOP não tem o objetivo de substituir os sistemas de origem. Seu papel é criar uma representação padronizada dos dados, preservando os registros originais, mas organizando-os de forma consistente para viabilizar análises comparáveis, reprodutíveis e escaláveis sobre dados observacionais em saúde.

O OMOP adota um princípio fundamental que sustenta todo o seu modelo analítico: toda análise é realizada exclusivamente sobre conceitos padronizados (*standard concepts*). Esse princípio é essencial para garantir a interoperabilidade semântica e a comparabilidade dos dados entre diferentes instituições e sistemas de origem.

Na prática, isso significa que cada dado clínico armazenado no CDM OMOP mantém duas representações complementares. A primeira é o conceito fonte (*source concept*), que corresponde ao código ou à descrição original utilizada no sistema de origem, preservando fielmente a informação conforme foi registrada. A segunda é o conceito padrão (*standard concept*), selecionado a partir dos vocabulários padronizados da OHDSI, e que representa o significado clínico unificado daquele dado.

Essa dupla representação permite que o CDM OMOP concilie dois objetivos fundamentais: preservar a rastreabilidade e a integridade dos dados originais e, ao mesmo tempo, viabilizar análises consistentes, reprodutíveis e independentes da codificação de origem. Como resultado, diferentes bases de dados podem ser analisadas de forma uniforme, mesmo quando utilizam vocabulários distintos nos sistemas fonte.

O mapeamento de vocabulários é o processo que viabiliza a interoperabilidade semântica no CDM OMOP. Na prática, ele garante que conceitos clínicos oriundos de diferentes sistemas, códigos e idiomas passem a ter o mesmo significado computacional, permitindo comparações e análises consistentes entre bases distintas. Sistemas de saúde utilizam múltiplos vocabulários e codificações. Esses códigos representam o mesmo conceito clínico com rótulos diferentes ou representam conceitos parecidos, mas não idênticos. Sem mapeamento, não é possível garantir que por exemplo, o diagnóstico de "Diabetes tipo 2" em um sistema seja tratado como o mesmo evento em outro.

A OHDSI mantém um repositório central de vocabulários integrados, que constitui a base para a interoperabilidade semântica no CDM OMOP. Esse repositório

 ©2026 SBC – Soc. Bras. de Computação

consolida e organiza diferentes sistemas de codificação amplamente utilizados na área da saúde, definindo papéis claros entre vocabulários padrão e vocabulários fonte.

Nesse contexto, alguns vocabulários são adotados como padrões para análise, como o SNOMED CT, utilizado para representar condições clínicas, o RxNorm, empregado na padronização de medicamentos, e o LOINC, utilizado para exames laboratoriais e medições clínicas. Outros sistemas de codificação, como ICD-10, ICD-9, CPT e diversos códigos locais, são tratados como vocabulários fonte, preservando a forma original com que os dados foram registrados nos sistemas de origem.

Cada conceito presente nesse repositório possui um CONCEPT_ID único, que funciona como identificador universal dentro do modelo OMOP. Além disso, todo conceito pertence a um domínio específico, como *Condition*, *Drug*, *Measurement* ou *Procedure*, definindo o tipo de evento clínico ao qual ele se refere. Os conceitos também são conectados por relacionamentos semânticos explícitos, que descrevem, por exemplo, equivalências, hierarquias ou mapeamentos entre códigos fonte e conceitos padrão.

Essa organização estruturada dos vocabulários permite que dados provenientes de sistemas distintos sejam interpretados de forma uniforme, garantindo consistência semântica, reprodutibilidade das análises e comparabilidade entre diferentes bases de dados observacionais.

1.5 Mapeamentos dos dados e dos vocabulários

O mapeamento dos dados para o CDM OMOP é o processo de transformação dos dados brutos de sistemas de origem (prontuário eletrônico, sistemas administrativos, laboratórios, farmácias etc.) para a estrutura padronizada e semântica do OMOP Common Data Model. Esse processo é fundamental para garantir interoperabilidade, comparabilidade e reuso analítico dos dados. De forma prática, o mapeamento ocorre em etapas bem definidas, que combinam conhecimento técnico, clínico e semântico. As etapas estão descritas a seguir:

1. Entendimento da fonte de dados (source data)

Esta é a etapa inicial e uma das mais críticas no processo de mapeamento para o CDM OMOP. Antes de qualquer transformação técnica, é indispensável conhecer profundamente o sistema de origem, pois é nele que estão as regras implícitas, as práticas clínicas locais e as limitações que influenciarão todo o processo de ETL. Um mapeamento bem-sucedido depende diretamente da qualidade desse diagnóstico inicial.

Esse entendimento começa pela análise detalhada do modelo de dados da fonte, incluindo a identificação das tabelas existentes, seus campos, os relacionamentos entre elas e a forma como os dados são organizados. É fundamental compreender onde cada informação clínica ou administrativa está armazenada e como diferentes tabelas se conectam, evitando interpretações equivocadas ou associações incorretas durante a transformação.

Paralelamente, é necessário avaliar os tipos de dados utilizados, como datas, campos textuais, códigos estruturados ou textos livres. Datas podem ter granularidade

 ©2026 SBC – Soc. Bras. de Computação

variada ou formatos inconsistentes, textos livres podem conter informações clínicas relevantes não estruturadas, e códigos podem seguir padrões distintos ou misturar conceitos diferentes em um mesmo campo. Identificar essas características desde o início evita perdas de informação e facilita a definição de regras adequadas de transformação.

Outro ponto essencial é o levantamento dos padrões locais de codificação. Sistemas de origem podem utilizar classificações como CID-10 para diagnósticos, catálogos internos para medicamentos, tabelas proprietárias para procedimentos ou combinações desses padrões. Compreender quais terminologias são usadas, como elas são mantidas e se existem variações locais é indispensável para o posterior mapeamento semântico para os vocabulários padronizados do OMOP.

A etapa também envolve uma análise cuidadosa da qualidade dos dados, identificando campos nulos, registros inconsistentes, duplicidades e lacunas frequentes. Essas características impactam diretamente as decisões de transformação e ajudam a definir regras de limpeza, consolidação ou inferência que serão aplicadas nas fases seguintes do ETL.

Durante esse processo de entendimento, busca-se responder a perguntas fundamentais, como onde exatamente estão registrados os diagnósticos, de que forma as visitas e episódios de cuidado são representados, se os dados de medicamentos correspondem a prescrições, dispensações ou administrações, e se existem datas confiáveis de início e fim para os eventos clínicos. As respostas a essas questões orientam tanto o mapeamento estrutural quanto a aplicação de regras clínicas e de negócio.

Identificar padrões locais de codificação é um passo essencial para garantir um mapeamento correto e semanticamente consistente para o CDM OMOP. Esse processo exige uma combinação de análise técnica, interpretação clínica e investigação operacional do sistema de origem. Na prática, ele ocorre por meio de várias abordagens complementares.

O primeiro passo é a análise exploratória do banco de dados de origem. Nessa etapa, examinam-se tabelas e campos que armazenam códigos clínicos, como diagnósticos, procedimentos, medicamentos e exames. A inspeção dos nomes das colunas, do tipo de dado e da distribuição dos valores frequentemente revela indícios claros de padrões utilizados, como códigos no formato CID-10, sequências numéricas internas ou combinações alfanuméricas proprietárias. Consultas simples de contagem e frequência ajudam a identificar quais códigos aparecem com maior recorrência e se seguem algum padrão reconhecível.

Em seguida, é fundamental analisar tabelas de domínio ou cadastros mestres. Muitos sistemas mantêm tabelas auxiliares que descrevem os códigos utilizados, com campos como descrição, categoria, status ou data de vigência. Essas tabelas costumam indicar explicitamente se um código pertence a um padrão internacional (como CID-10) ou se trata de um identificador interno criado pela própria instituição. Quando existem múltiplos catálogos para um mesmo tipo de informação, isso já sinaliza a presença de padrões locais distintos coexistindo.

 ©2026 SBC – Soc. Bras. de Computação

Outro aspecto importante é a avaliação do contexto de uso dos códigos. O mesmo código pode representar coisas diferentes dependendo de onde é utilizado. Por exemplo, um código numérico pode significar um diagnóstico em uma tabela médica e um motivo administrativo em outra. Analisar como os códigos se relacionam com visitas, especialidades, tipos de atendimento ou faturamento ajuda a entender seu real significado clínico e evita interpretações equivocadas.

A consulta a especialistas do domínio, como equipes clínicas, de faturamento, farmácia ou TI do sistema de origem, é uma etapa indispensável. Muitas regras de codificação são implícitas e nunca foram formalmente documentadas. Profissionais que utilizam o sistema no dia a dia conseguem explicar, por exemplo, se determinados códigos representam diagnósticos confirmados ou suspeitos, se medicamentos correspondem a prescrições ou a dispensações, ou se certos códigos internos são equivalentes a padrões internacionais específicos.

Também é importante investigar documentações internas e manuais do sistema, quando disponíveis. Especificações técnicas, dicionários de dados, guias de integração ou materiais de treinamento costumam descrever os padrões de codificação adotados, inclusive exceções e extensões locais. Mesmo documentação desatualizada pode fornecer pistas valiosas sobre a origem e a intenção dos códigos.

Por fim, a identificação de padrões locais se consolida por meio de testes de mapeamento piloto. Ao tentar mapear um conjunto de códigos para os vocabulários OMOP (como SNOMED, RxNorm ou LOINC), torna-se evidente quais códigos possuem correspondência direta que exigem interpretação clínica e quais não encontram equivalência. Esse exercício ajuda a classificar os códigos em padrões reconhecidos, extensões locais ou lacunas semânticas.

Em conjunto, essas práticas permitem identificar com clareza quais padrões de codificação estão presentes na fonte de dados, como eles são utilizados e quais cuidados precisam ser adotados no processo de ETL. Esse entendimento é crucial para evitar perdas de significado, reduzir erros de mapeamento e garantir que os dados transformados para o CDM OMOP reflitam fielmente a realidade clínica capturada nos sistemas de origem.

Em síntese, este é o alicerce de todo o processo de transformação para o CDM OMOP e garante que os dados sejam interpretados corretamente, preserva o significado clínico original e evita erros que poderiam comprometer análises futuras. Sem essa etapa bem executada, há alto risco de perdas de informação, distorções semânticas e conclusões analíticas incorretas.

2. Mapeamento estrutural (ETL – Extract, Transform, Load)

O mapeamento estrutural, realizado por meio do processo de ETL (Extract, Transform, Load), é a etapa responsável por alinhar fisicamente os dados de origem ao desenho lógico do OMOP Common Data Model, que possui tabelas e campos rigidamente pré-definidos. Diferentemente do mapeamento semântico, que trata do significado dos dados, o mapeamento estrutural responde à organização e ao posicionamento correto de cada informação dentro do modelo, garantindo conformidade com a arquitetura do CDM.

 ©2026 SBC – Soc. Bras. de Computação

A figura 1.2 ilustra o modelo CDM OMOP completo.

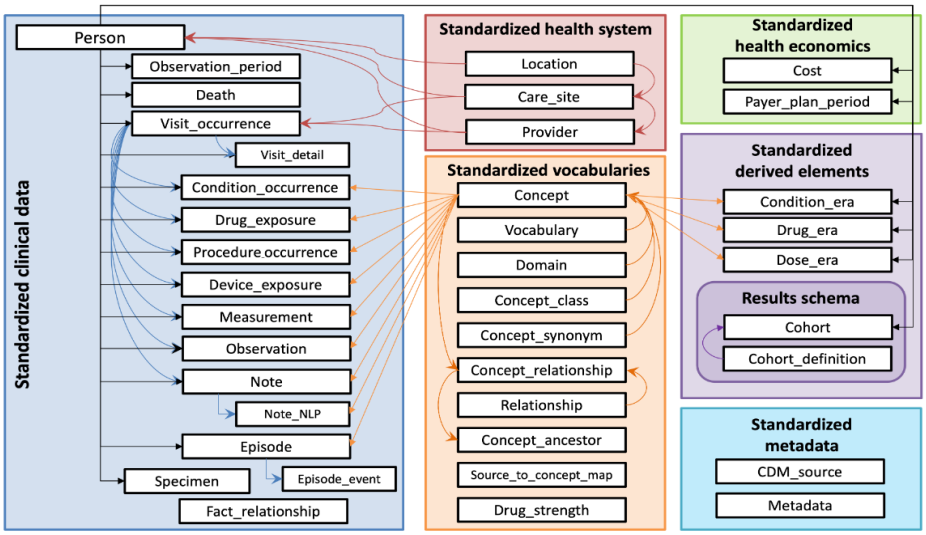


Figura 1.2 Visão geral da estrutura do *CDM OMOP V5* [9]

Nesse processo, define-se de forma explícita qual tabela do sistema de origem alimenta qual tabela do OMOP e como cada atributo da origem deve ser transformado e carregado nos campos correspondentes do CDM. Em outras palavras, trata-se de decidir "onde cada pedaço de dado deve morar dentro do OMOP". Por exemplo, informações cadastrais de pacientes são direcionadas para a tabela **PERSON**, enquanto registros de consultas, atendimentos ambulatoriais e internações são convertidos em registros da **VISIT_OCCURRENCE**. Da mesma forma, diagnósticos clínicos passam a compor a tabela **CONDITION_OCCURRENCE**, dados relacionados ao uso de medicamentos são estruturados na **DRUG_EXPOSURE**, e exames laboratoriais ou sinais vitais são representados na **MEASUREMENT**.

Durante essa etapa, não ocorre apenas a realocação direta de dados entre tabelas. São aplicadas diversas transformações estruturais essenciais para garantir consistência e integridade do modelo. Entre elas está a padronização de datas, assegurando formatos homogêneos e coerentes com o CDM, bem como a resolução de inconsistências comuns nos sistemas de origem, como datas parciais ou campos com múltiplos significados. Também ocorre a normalização de identificadores, na qual chaves oriundas de sistemas distintos são traduzidas para identificadores únicos e consistentes dentro do OMOP.

Um ponto central do mapeamento estrutural é a criação e gerenciamento dos identificadores do CDM, como PERSON_ID, VISIT_OCCURRENCE_ID e demais chaves primárias e estrangeiras. Esses identificadores garantem que todos os eventos clínicos estejam corretamente relacionados ao paciente e, quando aplicável, ao contexto da visita, permitindo análises longitudinais e preservando a integridade referencial do

[9] Fonte: https://ohdsi.github.io/CommonDataModel/

©2026 SBC – Soc. Bras. de Computação

banco. A geração desses IDs segue regras claras e documentadas, pois qualquer inconsistência nessa etapa compromete todo o ecossistema analítico.

Em conjunto, o mapeamento estrutural assegura que os dados transformados respeitem o desenho técnico do CDM OMOP e estejam preparados para receber a camada semântica e as validações de qualidade. Ele é a base que permite que as ferramentas da comunidade OHDSI funcionem corretamente e que análises padronizadas sejam executadas de forma consistente, escalável e reprodutível.

3. Mapeamento semântico (vocabulários padronizados)

O mapeamento semântico por meio de vocabulários padronizados é um dos componentes mais críticos e conceitualmente relevantes do CDM OMOP. Diferentemente de uma simples padronização estrutural, essa etapa garante que os dados carregados compartilhem o mesmo significado clínico, independentemente da origem, do sistema ou do país onde foram gerados. Sem essa padronização terminológica, análises comparáveis e reprodutíveis seriam inviáveis.

Na prática, os códigos locais existentes nos sistemas de origem, muitas vezes criados para fins assistenciais, administrativos ou financeiros, precisam ser convertidos em conceitos padrão (standard concepts) definidos pelo OMOP. Esses conceitos padrão pertencem a vocabulários clínicos internacionais amplamente aceitos, que funcionam como uma "língua franca" para os dados de saúde. Exemplos comuns incluem o mapeamento de códigos CID-10 para SNOMED CT no caso de condições clínicas, a conversão de códigos locais de medicamentos para RxNorm, a padronização de exames laboratoriais em LOINC e a representação de procedimentos clínicos em SNOMED CT ou outros vocabulários suportados pelo modelo.

Esse processo de mapeamento é sustentado pelas tabelas centrais de vocabulário do OMOP. A tabela **CONCEPT** armazena todos os conceitos disponíveis, incluindo seus identificadores, nomes, domínios e a indicação de quais são conceitos padrão. Já a tabela **CONCEPT_RELATIONSHIP** descreve as relações entre conceitos, como mapeamentos de equivalência, hierarquia ou associação entre códigos não padronizados e seus correspondentes padrão. Complementarmente, durante o ETL, costumam ser criadas tabelas auxiliares de mapeamento, que documentam de forma explícita como cada código da fonte foi traduzido para o universo OMOP, servindo como referência técnica e de governança.

No entanto, a tabela **CONCEPT_RELATIONSHIP** é um dos elementos centrais da camada semântica do CDM OMOP, pois é responsável por descrever como os conceitos clínicos e administrativos se relacionam entre si dentro do ecossistema de vocabulários padronizados. Enquanto a tabela **CONCEPT** define cada conceito individualmente, indicando seu identificador, nome, domínio e se ele é padrão ou não, a **CONCEPT_RELATIONSHIP** explicita o significado das conexões entre esses conceitos, permitindo interoperabilidade semântica real entre diferentes sistemas e terminologias.

 ©2026 SBC – Soc. Bras. de Computação

Na prática, essa tabela funciona como uma grande rede de relacionamentos direcionais, na qual cada registro conecta um conceito de origem a um conceito de destino por meio de um tipo de relacionamento bem definido. Além dos identificadores dos conceitos envolvidos, cada relacionamento possui um período de validade, garantindo que mudanças nos vocabulários ao longo do tempo sejam corretamente representadas e respeitadas durante análises e processos de ETL.

Um dos usos mais importantes da **CONCEPT_RELATIONSHIP** é o mapeamento de códigos não padronizados para conceitos padrão. Códigos locais, CID-10 ou terminologias proprietárias não são utilizados diretamente nas análises OMOP. Em vez disso, eles são conectados a conceitos padrão por meio de relacionamentos como "Maps to", que indica a equivalência semântica entre um conceito não padrão e seu correspondente padrão. Dessa forma, mesmo que dados distintos tenham origens diferentes, eles passam a ser interpretados de maneira uniforme pelas ferramentas analíticas do OMOP.

Além do mapeamento, a **CONCEPT_RELATIONSHIP** também sustenta hierarquias clínicas e relações semânticas complexas. Relacionamentos como "Is a" permitem identificar que um conceito específico pertence a uma categoria mais ampla, viabilizando análises que incluem automaticamente todos os subtipos de uma condição, procedimento ou medicamento. No domínio de medicamentos, por exemplo, relações como "Has ingredient" conectam produtos farmacêuticos aos seus princípios ativos, permitindo análises tanto no nível de formulação quanto no nível de substância.

Ferramentas da comunidade OHDSI, como ATLAS e ACHILLES, dependem fortemente da CONCEPT_RELATIONSHIP para funcionar corretamente. Ao definir uma coorte ou uma análise, essas ferramentas utilizam os relacionamentos para incluir conceitos equivalentes ou descendentes, garantindo consistência e reprodutibilidade entre diferentes bases OMOP. É essa lógica que permite que uma mesma definição analítica seja executada em múltiplas instituições e produza resultados comparáveis, mesmo quando os dados de origem são heterogêneos.

Por fim, a CONCEPT_RELATIONSHIP é parte de um conjunto de vocabulários que evolui continuamente. Por isso, seus relacionamentos são versionados e possuem validade temporal, o que reforça a importância de alinhar o processo de ETL à versão específica dos vocabulários utilizados. Em síntese, essa tabela é o que transforma o OMOP de um modelo estrutural em um modelo verdadeiramente semântico, permitindo tradução entre terminologias, navegação hierárquica e análises padronizadas em escala global.

Nem sempre, porém, existe uma correspondência direta e perfeita entre um código local e um conceito padrão. Nesses casos, o OMOP oferece diferentes estratégias para preservar o dado sem comprometer a integridade analítica. Um código pode ser mapeado para um conceito mais genérico, quando não há equivalência exata, mantendo o significado clínico essencial. Alternativamente, o dado pode ser mantido como um non-standard concept, permitindo sua preservação e rastreabilidade, mesmo que ele não seja utilizado diretamente nas análises padronizadas. Por fim, quando não há mapeamento

 ©2026 SBC – Soc. Bras. de Computação

adequado, a situação deve ser explicitamente documentada como uma lacuna semântica, garantindo transparência sobre os limites da base de dados.

Em conjunto, o mapeamento semântico transforma dados locais e heterogêneos em informações semanticamente interoperáveis, capazes de sustentar estudos multicêntricos, análises preditivas e geração de Real World Evidence. É essa camada semântica que diferencia o OMOP de um simples modelo relacional e o torna uma base sólida para análises científicas confiáveis e comparáveis em escala global.

4. Aplicação de regras clínicas e de negócio

A aplicação de regras clínicas e de negócio é uma etapa essencial no processo de transformação dos dados para o CDM OMOP, pois nem toda informação proveniente do sistema de origem pode ser transferida de forma direta ou literal. Os dados operacionais refletem realidades locais, diferentes práticas assistenciais e limitações dos sistemas transacionais, o que exige interpretações cuidadosas para que o significado clínico seja corretamente representado no modelo padronizado.

Nesse contexto, são definidas e aplicadas regras de transformação que traduzem a realidade da fonte para a lógica analítica do OMOP. Um exemplo frequente é a necessidade de definir qual diagnóstico é primário ou secundário, especialmente em cenários onde o sistema de origem não diferencia explicitamente essa informação. Essa definição é crucial para diversas análises clínicas e epidemiológicas, como estudos de causa principal de internação ou estratificação de pacientes.

Outro ponto comum é a unificação de registros duplicados. Sistemas de origem podem gerar múltiplos registros para um mesmo evento clínico devido a correções, reprocessamentos ou registros paralelos. As regras de negócio determinam quando esses registros devem ser consolidados, evitando supercontagem de eventos e distorções analíticas.

A inferência de datas de término também é uma prática recorrente, sobretudo para medicamentos, internações ou condições clínicas onde apenas a data de início está disponível. Nesses casos, regras clínicas bem definidas permitem estimar períodos de exposição ou duração de eventos de forma consistente, sempre com critérios claros e documentados.

Além disso, muitas fontes de dados não classificam explicitamente o tipo de visita. Cabe às regras de transformação identificarem se um atendimento deve ser representado como internação, atendimento ambulatorial ou emergência, a partir de atributos como especialidade, local de atendimento, duração ou tipo de faturamento. Essa classificação é fundamental para análises que dependem do contexto assistencial.

Todas essas regras têm impacto direto nas análises futuras e, por isso, devem ser claramente documentadas e rigorosamente versionadas. A documentação garante transparência, reprodutibilidade e auditabilidade, enquanto o versionamento permite acompanhar evoluções, comparar resultados entre cargas e manter consistência ao longo do tempo. Dessa forma, a aplicação consciente de regras clínicas e de negócio transforma

 ©2026 SBC – Soc. Bras. de Computação

dados brutos em informações analiticamente confiáveis, preservando o significado clínico e a robustez do CDM OMOP.

5. A carga para o CDM OMOP (Load)

Corresponde à etapa em que os dados, já devidamente transformados do ponto de vista estrutural e semântico, são inseridos nas tabelas finais do modelo. Nesse momento, todo o trabalho prévio de entendimento da fonte, aplicação de regras de negócio e mapeamento para vocabulários padronizados se materializa em um banco de dados que segue rigorosamente o desenho do OMOP Common Data Model.

Durante a carga, é essencial respeitar as chaves primárias, que garantem a unicidade dos registros, e as chaves estrangeiras, que asseguram a integridade referencial entre as tabelas. Relacionamentos como aqueles entre PERSON, VISIT_OCCURRENCE e as tabelas clínicas (por exemplo, CONDITION_OCCURRENCE, DRUG_EXPOSURE e MEASUREMENT) devem estar corretamente estabelecidos, permitindo que os eventos clínicos sejam corretamente associados ao paciente e ao contexto assistencial correspondente.

Outro aspecto crítico dessa etapa é o respeito aos tipos de dados definidos pelo modelo, como formatos de datas, campos numéricos, identificadores e flags conceituais.

A conformidade com esses padrões evita inconsistências técnicas, falhas em ferramentas analíticas e problemas de interoperabilidade. Além disso, a carga deve observar regras específicas do CDM, como o uso correto de identificadores de conceitos, a distinção entre conceitos padrão e não padrão e o preenchimento adequado de campos obrigatórios.

Garantir a integridade referencial é um requisito fundamental na implementação do CDM OMOP, pois assegura que os relacionamentos entre pacientes, visitas e eventos clínicos sejam consistentes, rastreáveis e confiáveis ao longo de todo o banco de dados. Na prática, isso envolve uma combinação de boas práticas de modelagem, regras técnicas, processos de ETL bem definidos e validações sistemáticas.

O primeiro passo para garantir integridade referencial é o controle rigoroso dos identificadores. Cada tabela do OMOP possui uma chave primária única (por exemplo, person_id, visit_occurrence_id, condition_occurrence_id) que deve ser gerada de forma consistente durante o ETL. Esses identificadores não podem ser reutilizados nem alterados após a carga, pois servem como base para todos os relacionamentos entre tabelas.

Em paralelo, é essencial assegurar que todas as chaves estrangeiras apontem para registros válidos. Por exemplo, todo person_id presente em tabelas como condition_occurrence, drug_exposure ou measurement deve existir previamente na tabela PERSON. Da mesma forma, quando um evento clínico referência uma visita (visit_occurrence_id), essa visita precisa existir na tabela VISIT_OCCURRENCE. Para isso, o fluxo de carga deve respeitar uma ordem lógica, carregando primeiro as tabelas centrais (como PERSON e VISIT_OCCURRENCE) e só depois as tabelas de eventos.

 ©2026 SBC – Soc. Bras. de Computação

Outra prática fundamental é a padronização e persistência dos mapeamentos de origem. Durante o ETL, identificadores do sistema fonte deve ser consistentemente traduzidos para os identificadores OMOP por meio de tabelas de correspondência (lookup tables). Essas tabelas garantem que o mesmo paciente, visita ou evento seja sempre representado pelo mesmo identificador no CDM, evitando duplicidades ou quebras de relacionamento.

A validação automática da integridade também é indispensável. Ferramentas da comunidade OHDSI, como o Data Quality Dashboard (DQD), executam centenas de verificações que identificam registros órfãos, referências inválidas ou violações das regras estruturais do CDM. O ACHILLES, por sua vez, ajuda a detectar indiretamente problemas de integridade ao expor inconsistências de volume, lacunas inesperadas e padrões atípicos nos dados.

Além disso, é altamente recomendável implementar restrições de integridade no próprio banco de dados, sempre que tecnicamente possível. Constraints de chave primária e estrangeira, validações de tipo e regras de não-nulidade reforçam a integridade em nível físico, reduzindo o risco de cargas incorretas ou manuais não controlados.

Por fim, a integridade referencial depende fortemente de governança e documentação. Todas as regras de geração de identificadores, relacionamentos permitidos, exceções conhecidas e decisões de modelagem devem estar documentadas e versionadas. Isso garante que futuras evoluções do ETL ou do CDM mantenham coerência com as cargas anteriores.

A ordem correta de carga dos dados (load order) é um ponto crítico para o funcionamento adequado das *constraints* no CDM OMOP. Para que as chaves estrangeiras sejam respeitadas e a integridade referencial seja mantida, a carga deve seguir rigorosamente a hierarquia lógica do modelo, iniciando pelas tabelas centrais e, somente depois, avançando para as tabelas de eventos clínicos.

O processo deve começar pela tabela PERSON, que representa o paciente e é a base para todos os demais relacionamentos. Em seguida, deve-se carregar a tabela VISIT_OCCURRENCE, responsável por registrar os episódios de cuidado e servir de contexto para muitos eventos clínicos. Somente após essas duas etapas é que as tabelas de eventos clínicos devem ser populadas, como CONDITION_OCCURRENCE, DRUG_EXPOSURE, MEASUREMENT, PROCEDURE_OCCURRENCE e OBSERVATION, todas dependentes direta ou indiretamente das tabelas anteriores.

Seguir essa ordem é essencial porque os registros das tabelas de eventos fazem referência a pacientes e, frequentemente, a visitas já existentes. Caso a carga seja executada fora dessa sequência, o banco rejeitará os dados por violação de chave estrangeira, tornando o processo de ETL instável e comprometendo a integridade do CDM. Portanto, respeitar o load order não é apenas uma boa prática, mas um pré-requisito para um ambiente OMOP consistente, robusto e governado.

Ao final do processo de carga, o banco de dados já passa a refletir fielmente a arquitetura lógica e semântica do OMOP, tornando-se plenamente compatível com as ferramentas da comunidade OHDSI e apto para validações, análises descritivas e estudos

 ©2026 SBC – Soc. Bras. de Computação

observacionais. Essa etapa marca a transição definitiva dos dados operacionais para um ativo analítico padronizado, pronto para uso científico e analítico em larga escala.

6. Validação e controle de qualidade

A validação e o controle de qualidade são etapas indispensáveis após a carga dos dados no CDM OMOP. Nessa fase, é fundamental assegurar que os dados estejam completos, que sejam clinicamente plausíveis, que mantenham coerência temporal entre os eventos e que respeitem rigorosamente as regras estruturais e semânticas do CDM. Esse processo garante que a transformação dos dados preservou tanto a integridade técnica quanto o significado clínico das informações. Para isso, normalmente são utilizadas ferramentas consolidadas da comunidade OHDSI.

O Data Quality Dashboard (DQD) aplica um amplo conjunto de validações padronizadas, avaliando conformidade, plausibilidade e completude de cada tabela e atributo do modelo. Já o ACHILLES é amplamente utilizado para análises descritivas e exploratórias, permitindo identificar padrões, outliers e possíveis inconsistências nos dados carregados. Em conjunto, essas ferramentas fornecem evidências objetivas da qualidade do banco, garantindo que ele esteja pronto para análises reprodutíveis, confiáveis e comparáveis em diferentes contextos e instituições.

7. Documentação e governança

A documentação e a governança são etapas fundamentais no processo de mapeamento para o CDM OMOP. Todo o trabalho realizado durante a transformação dos dados deve ser cuidadosamente documentado, incluindo as decisões clínicas adotadas, as regras de transformação aplicadas, as limitações conhecidas dos dados e o nível de cobertura semântica alcançada durante o mapeamento. Esse registro garante que o significado original dos dados, bem como as escolhas feitas ao longo do processo, seja compreendido e rastreáveis.

Essa documentação é essencial para assegurar a auditabilidade do processo, permitindo verificar a consistência e a qualidade dos dados ao longo do tempo. Além disso, ela viabiliza a evolução do modelo, facilitando ajustes, refinamentos e reprocessamentos futuros. Do ponto de vista científico, contribui para a transparência e reprodutibilidade dos estudos, e, do ponto de vista organizacional, permite o reuso do trabalho por outras equipes, reduzindo esforços, aumentando a confiabilidade e promovendo escalabilidade nas iniciativas baseadas em dados.

A governança de dados no contexto do CDM OMOP é sustentada por um conjunto integrado de ferramentas que atuam de forma complementar para garantir padronização, qualidade, rastreabilidade e transparência ao longo de todo o ciclo de vida dos dados. As ferramentas da comunidade OHDSI desempenham um papel central nesse processo, especialmente o ATLAS, que permite a definição, versionamento e compartilhamento de coortes, conceitos e estudos observacionais, assegurando que as análises sejam reproduzíveis e auditáveis. O ACHILLES complementa esse ecossistema ao gerar perfis descritivos dos dados, contribuindo para a compreensão da estrutura e da distribuição das

 ©2026 SBC – Soc. Bras. de Computação

informações carregadas no CDM, enquanto o Data Quality Dashboard aplica verificações sistemáticas de conformidade, completude e plausibilidade, fortalecendo a governança da qualidade dos dados.

Além das ferramentas específicas do OMOP, a governança é ampliada por soluções de gestão de metadados e documentação, como catálogos de dados e repositórios colaborativos. Essas ferramentas são essenciais para registrar regras de ETL, decisões clínicas, limitações conhecidas, versões do CDM e dos vocabulários utilizados, criando uma memória institucional sólida e facilitando o reuso das informações por diferentes equipes. O controle de versão, geralmente apoiado por plataformas como Git e seus derivados, garante rastreabilidade técnica, permitindo acompanhar mudanças em scripts, mapeamentos semânticos e regras de negócio, além de viabilizar auditorias e a evolução controlada do modelo.

A governança semântica também é um pilar fundamental, apoiada pelos vocabulários oficiais do OMOP e por mecanismos de gestão dos mapeamentos entre códigos locais e conceitos padrão, assegurando consistência clínica e alinhamento com terminologias internacionais. Complementarmente, ferramentas de monitoramento operacional e observabilidade dos pipelines de dados reforçam a governança ao permitir a detecção de falhas, desvios de qualidade e impactos entre versões de carga. Em conjunto, essas ferramentas garantem que o CDM OMOP não seja apenas um repositório técnico, mas um ativo confiável, sustentável e preparado para suportar análises científicas e estratégias analíticas avançadas em escala.

1.5.1 Como o mapeamento dos vocabulários acontece na prática

O mapeamento de vocabulários no contexto do CDM OMOP ocorre principalmente durante o processo de ETL (Extract, Transform, Load) e envolve etapas bem definidas que garantem a interoperabilidade semântica dos dados.

Na primeira etapa, realiza-se a identificação do código fonte presente no sistema de origem. Isso inclui o reconhecimento do código original, como, por exemplo, o CID-10 E11.9 (Diabetes tipo 2 sem complicações), a identificação do vocabulário de origem ao qual esse código pertence e a contextualização do dado dentro do cenário clínico, como diagnóstico, medicamento ou exame laboratorial.

Essas informações são preservadas no CDM OMOP por meio de campos específicos que asseguram a rastreabilidade do dado original. O valor do código é registrado no campo *_source_value, enquanto o identificador do conceito correspondente no vocabulário fonte é armazenado no campo *_source_concept_id. Dessa forma, o modelo mantém um vínculo explícito entre o dado padronizado e sua origem.

Após a identificação do código fonte, o próximo passo é a busca do conceito padrão correspondente, que será utilizado nas análises. Esse mapeamento é realizado, em geral, por meio de relacionamentos semânticos do tipo *"Maps to"*, definidos e mantidos no repositório de vocabulários da OHDSI, especificamente na tabela CONCEPT_RELATIONSHIP.

 ©2026 SBC – Soc. Bras. de Computação

Por exemplo, o código CID-10 E11.9 é mapeado para o conceito SNOMED CT 44054006, que representa *Type 2 diabetes mellitus*. Esse conceito SNOMED CT passa a ser o conceito padrão armazenado no campo analítico do CDM OMOP e é sobre ele que todas as consultas, definições de coorte e análises estatísticas serão realizadas. Esse mecanismo assegura que diferentes códigos de origem, ainda que provenientes de sistemas distintos, sejam interpretados de maneira uniforme, possibilitando análises consistentes, comparáveis e reprodutíveis em ambientes distribuídos.

No CDM OMOP, cada evento clínico armazenado passa a conter, de forma explícita, tanto a representação padronizada quanto a informação original do sistema de origem. O campo condition_concept_id registra o conceito padrão, geralmente proveniente de vocabulários como o SNOMED CT, e é esse identificador que serve de base para todas as análises e definições de coortes. Paralelamente, o campo condition_source_concept_id armazena o conceito original, como um código CID-10, enquanto o campo condition_source_value preserva o código ou texto exatamente como registrado no sistema fonte.

Dessa forma, o OMOP garante que nenhuma informação original seja perdida, ao mesmo tempo em que assegura que as análises sejam realizadas de maneira uniforme, utilizando conceitos padronizados e semanticamente consistentes.

1.5.2 Mapeamento automático versus mapeamento manual

O processo de mapeamento de vocabulários pode ocorrer de maneira automática ou manual, dependendo da disponibilidade e da qualidade dos relacionamentos existentes nos vocabulários padronizados.

O mapeamento automático utiliza relacionamentos previamente definidos nos vocabulários do ecossistema OHDSI, especialmente aqueles do tipo *"Maps to"*. Esse tipo de mapeamento apresenta boa cobertura para codificações amplamente utilizadas, como CID, LOINC e RxNorm, sendo, em geral, um processo rápido, escalável e consistente entre diferentes bases de dados.

Já o mapeamento manual torna-se necessário em situações mais complexas, como quando não há uma correspondência direta nos vocabulários padrão, quando os dados utilizam códigos locais ou quando os conceitos clínicos são extremamente específicos. Nesses casos, especialistas clínicos e analistas de dados avaliam cuidadosamente o significado clínico do código de origem, selecionam o conceito padrão mais adequado e documentam as decisões tomadas. Essa documentação é essencial para garantir rastreabilidade, transparência e governança dos dados, especialmente em estudos multicêntricos ou regulatórios.

1.5.3 Análises padronizadas sobre o CDM

Uma vez que os dados são transformados e organizados no CDM OMOP, abre-se a possibilidade de realizar análises padronizadas e reprodutíveis, eliminando a necessidade de reescrever códigos específicos para cada fonte de dados. Essa padronização permite que diferentes bases, originalmente heterogêneas em estrutura, terminologia e

 ©2026 SBC – Soc. Bras. de Computação

granularidade, passem a falar a mesma "linguagem analítica", reduzindo significativamente o esforço técnico e aumentando a confiabilidade dos resultados.

Nesse contexto, a definição de coortes torna-se mais clara e consistente, pois é baseada em conceitos clínicos padronizados e amplamente validados pela comunidade internacional. Assim, critérios como diagnósticos, procedimentos, medicamentos e eventos clínicos seguem vocabulários comuns, possibilitando a criação de coortes complexas, como, por exemplo, "pacientes adultos com diabetes tipo 2 em uso de metformina", de forma uniforme e comparável entre instituições e países. Isso não apenas melhora a qualidade das análises, mas também favorece a transparência e a replicabilidade dos estudos.

As ferramentas desenvolvidas pela comunidade OHDSI operam diretamente sobre o CDM OMOP, o que viabiliza a execução de uma ampla gama de estudos analíticos avançados. Entre eles, destacam-se os estudos de segurança de medicamentos, voltados à identificação de eventos adversos e riscos associados a terapias; as análises de efetividade comparativa, que permitem avaliar diferentes intervenções em condições do mundo real; a caracterização detalhada de coortes, essencial para compreender perfis populacionais; e o desenvolvimento de modelos preditivos, que apoiam a tomada de decisão clínica e estratégica por meio de técnicas de ciência de dados e aprendizado de máquina.

Um dos maiores diferenciais desse ecossistema é a possibilidade de reutilização integral dos scripts analíticos. O mesmo código pode ser executado em um hospital no Brasil, em uma seguradora de saúde na Europa ou em uma base nacional dos Estados Unidos, mantendo a lógica analítica e os critérios de estudo inalterados. Dessa forma, não é necessário mover ou centralizar os dados, o que fortalece a governança, a privacidade e a conformidade regulatória. Apenas o código analítico é compartilhado, promovendo colaboração global, escalabilidade e geração de evidências robustas em larga escala.

1.5.4 Funcionamento em rede distribuída

O funcionamento do OMOP no contexto de redes distribuídas é um dos seus pilares mais importantes e o que viabiliza, na prática, a colaboração entre múltiplas instituições de forma segura, escalável e regulatoriamente adequada. Esse modelo é baseado em uma arquitetura federada, na qual cada organização mantém total controle sobre seus próprios dados, preservando sua autonomia institucional e suas responsabilidades legais. Os dados clínicos, administrativos ou assistenciais não são transferidos ou centralizados em um repositório único; eles permanecem armazenados localmente, dentro dos ambientes seguros de cada instituição participante.

Nesse contexto, o que efetivamente circula entre os participantes da rede é o código analítico padronizado, desenvolvido de acordo com o CDM OMOP e com as ferramentas da comunidade OHDSI. Esse código é compartilhado e executado localmente em cada base de dados, respeitando exatamente a mesma lógica analítica, definições de coorte, critérios de inclusão e exclusão, métricas e modelos estatísticos. Após a execução, apenas os resultados agregados, previamente definidos e anonimizados, são consolidados

 ©2026 SBC – Soc. Bras. de Computação

para análise conjunta, sem exposição de dados identificáveis ou sensíveis em nível individual.

Esse modelo federado traz ganhos diretos e estratégicos. Do ponto de vista regulatório, ele permite a conformidade com legislações rigorosas de proteção de dados, como a LGPD no Brasil, o GDPR na União Europeia e o HIPAA nos Estados Unidos, uma vez que não há compartilhamento de dados pessoais entre instituições ou países. A governança da informação permanece local, enquanto a colaboração científica ocorre de forma coordenada e transparente, baseada em métodos comuns e auditáveis.

Além disso, o funcionamento em rede distribuída torna viáveis estudos multicêntricos globais, envolvendo hospitais, sistemas de saúde, seguradoras, agências regulatórias e centros de pesquisa, mesmo quando esses participantes estão sujeitos a diferentes contextos legais e operacionais. Todos contribuem para a geração de evidências utilizando seus próprios dados, mas respondendo às mesmas perguntas de pesquisa, o que fortalece a comparabilidade e a robustez dos achados.

Por fim, essa abordagem possibilita a escala massiva de análises com dados do mundo real, algo fundamental para avaliações de segurança de medicamentos, efetividade comparativa, vigilância pós-comercialização, estudos epidemiológicos e modelos preditivos. Ao combinar grandes volumes de dados distribuídos geograficamente, sem comprometer privacidade ou soberania institucional, o OMOP viabiliza uma ciência colaborativa em larga escala, alinhada às necessidades da saúde baseada em evidências e à transformação digital dos sistemas de saúde.

1.5.5 A conexão com a IA: OHDSI Keeper

Numa época com destaque aos avanços contínuos nas tecnologias de Deep Learning e Inteligência Artificial (IA), é fácil esquecer que nada disso seria possível sem dados e evidências confiáveis que alimentem e treinem estes algoritmos.

Para gerar evidências confiáveis precisamos superar os obstáculos que se apresentam no caminho dela, representados como preocupações tanto metodológicas quanto técnicas. A figura 1.3 apresenta o caminho e as perguntas para as quais devemos dar resposta.

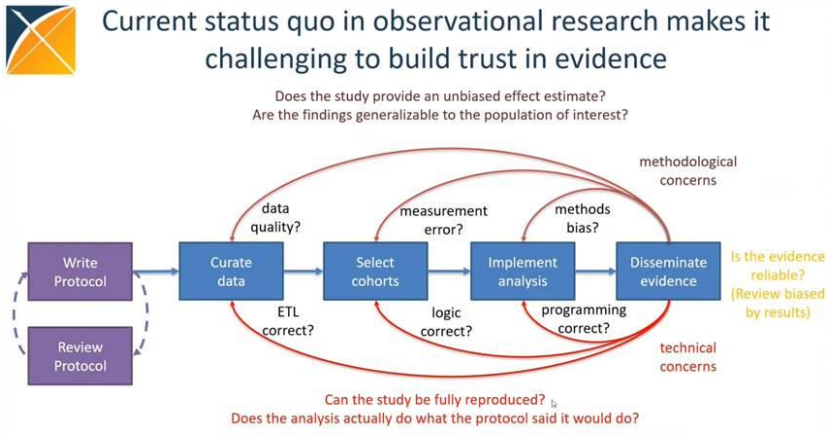


Figura 1.3 O caminho da evidência confiável e seus obstáculos

 ©2026 SBC – Soc. Bras. de Computação

Evidências confiáveis devem ser **repetíveis**, o que significa que os pesquisadores devem esperar produzir resultados idênticos ao aplicar a mesma análise aos mesmos dados para qualquer questão específica. Implícita nesse requisito mínimo está a noção de que a evidência é o resultado da execução de um processo definido com uma entrada específica e deve estar livre de intervenção manual ou tomada de decisão posterior ao longo do processo.

Idealmente, evidências confiáveis devem ser **reproduzíveis**, de modo que um pesquisador diferente possa realizar a mesma tarefa de executar uma determinada análise em um determinado banco de dados e esperar produzir um resultado idêntico ao do primeiro pesquisador. A reprodutibilidade exige que o processo seja totalmente especificado, geralmente em formato legível por humanos e executável por computador, de forma que nenhuma decisão do estudo fique a critério do investigador.

A solução mais eficiente para alcançar repetibilidade e reprodutibilidade é usar rotinas analíticas padronizadas que tenham entradas e saídas definidas e aplicar esses procedimentos a bancos de dados com controle de versão.

É mais provável que tenhamos confiança nas nossas evidências se pudermos demonstrar que elas são **replicáveis**, ou seja, se a mesma questão abordada usando a mesma análise com dados semelhantes produzir resultados semelhantes. A Figura 1.4 apresenta estes conceitos.

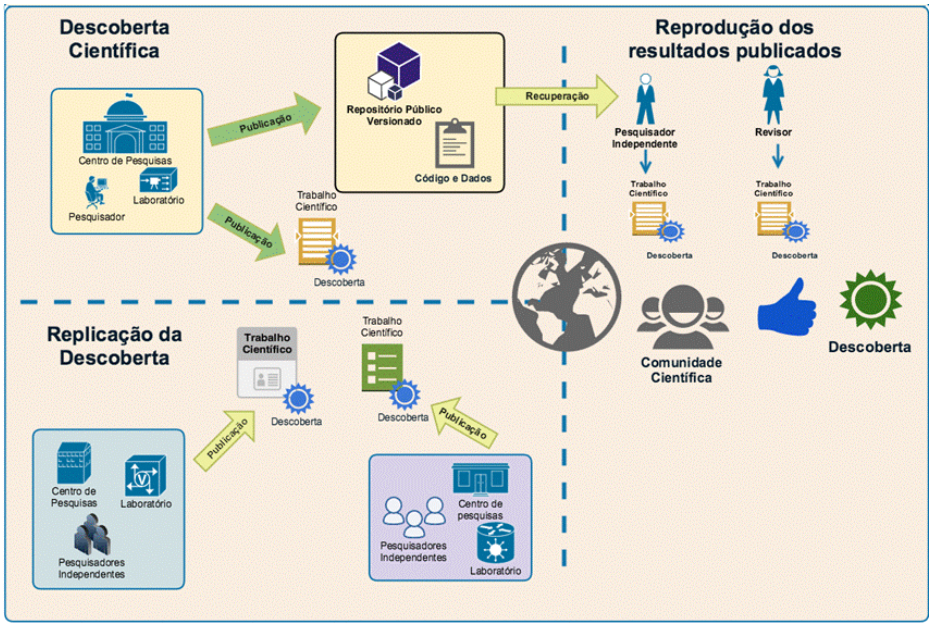


Figura 1.4 Pesquisa reprodutível - Replicação e reprodução

No contexto da predição em nível de paciente, a replicabilidade destaca o valor da validação externa e a capacidade de avaliar o desempenho de um modelo treinado em um banco de dados, observando sua acurácia discriminativa e calibração quando aplicado a um banco de dados diferente.

©2026 SBC – Soc. Bras. de Computação

Em circunstâncias em que análises idênticas são realizadas em diferentes bancos de dados e ainda apresentam resultados consistentemente semelhantes, aumentamos nossa confiança na **generalização** das evidências. Um valor fundamental da rede de pesquisa OHDSI é a diversidade representada por diferentes populações, regiões geográficas e processos de coleta de dados.

Evidências confiáveis devem ser **robustas**, o que significa que as conclusões não devem ser excessivamente sensíveis às escolhas subjetivas que podem ser feitas em uma análise. Se houver métodos estatísticos alternativos que possam ser considerados potencialmente razoáveis para um determinado estudo, pode ser reconfortante verificar se os diferentes métodos produzem resultados semelhantes ou, inversamente, alertar caso sejam encontrados resultados discordantes. (Madigan, Ryan e Schuemie 2013) Para a estimativa do efeito em nível populacional, as análises de sensibilidade podem incluir escolhas de alto nível no desenho do estudo, como a aplicação de um desenho de coorte comparativa ou de séries de casos autocontrolados, ou podem se concentrar em considerações analíticas inerentes ao desenho, como a realização de pareamento por escore de propensão, estratificação ou ponderação como estratégia de ajuste de fatores de confusão na estrutura de coorte comparativa.

Por último, mas talvez o mais importante, as evidências devem ser **calibradas**. Não basta ter um sistema gerador de evidências que produza respostas para perguntas desconhecidas se o desempenho desse sistema não puder ser verificado. Espera-se que um sistema fechado tenha características operacionais conhecidas, que devem poder ser medidas e comunicadas como contexto para a interpretação de quaisquer resultados produzidos pelo sistema. Artefatos estatísticos devem poder ser demonstrados empiricamente como tendo propriedades bem definidas, como um intervalo de confiança de 95% com probabilidade de cobertura de 95% ou uma coorte com probabilidade prevista de 10% com uma proporção observada de eventos em 10% da população. Um estudo observacional deve sempre ser acompanhado por diagnósticos que testem as premissas relativas ao delineamento, aos métodos e aos dados. Esses diagnósticos devem se concentrar na avaliação das principais ameaças à validade do estudo: viés de seleção, fatores de confusão e erros de medição. Os controles negativos têm se mostrado uma ferramenta poderosa para identificar e mitigar erros sistemáticos em estudos observacionais (Schuemie et al. 2016; Schuemie, Hripcsak, et al. 2018; Schuemie, Ryan, et al. 2018). A figura 1.5 resume os atributos desejáveis de uma evidência robusta.

Desired attribute	Question	Researcher	Data	Analysis		Result
Repeatable	Identical	Identical	Identical	Identical	=	Identical
Reproducible	Identical	Different	Identical	Identical	=	Identical
Replicable	Identical	Same or different	Similar	Identical	=	Similar
Generalizable	Identical	Same or different	Different	Identical	=	Similar
Robust	Identical	Same or different	Same or different	Different	=	Similar
Calibrated	Similar (controls)	Identical	Identical	Identical	=	Statistically consistent

Figura 1.5 Atributos da evidência robusta

 ©2026 SBC – Soc. Bras. de Computação

O sistema da OHDSI

O processo de converter RWD (Real World Data) em RWE (Real World Evidence) levou a OHDSI a construir um sistema de ciência aberta no lugar de gerar apenas componentes que precisam ser combinados posteriormente.

Este sistema, mostrado na figura 1.6, é o que possibilita a colaboração entre os parceiros da rede OHDSI e é a solução para criar confiança na geração de evidências do mundo real (RWE).

Ele consiste das seguintes etapas:

- Avaliação da qualidade dos dados
- Desenvolvimento e avaliação de Fenótipos
- Avaliação da confiabilidade das análises

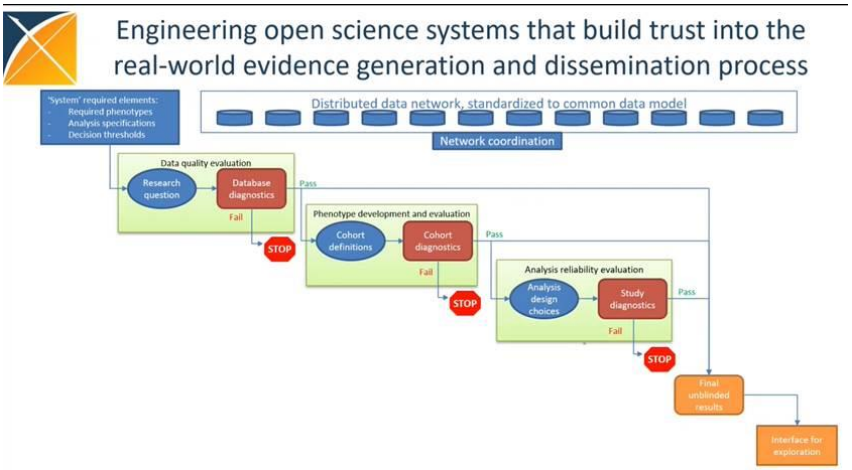


Figura 1.6 Sistema OHDSI de geração de evidências

Para assegurar confiança no processo de conversão de RWD (Real Worl Data) em RWE (Real World Evidence) é necessário estabelecer critérios e controles em cada etapa que validem os resultados e não deixem o processo continuar caso estes critérios não sejam cumpridos.

A solução para assegurar a confiabilidade na produção de evidências, reside na criação de sistemas que tenham entradas e saídas definidas de forma consistente, que possam ser avaliadas objetivamente a priori, mitigando a incidência de vieses e a subjetividade inerente ao processo de interpretação.

Tal processo basicamente exige que, a priori, se defina o conjunto de questões de pesquisa, seja uma única questão sobre uma exposição e um desfecho, ou um conjunto de questões, como por exemplo, para estudar a segurança e a eficácia comparativa dos tratamentos para diabetes.

Seja qual for o conjunto de questões de pesquisa, precisamos desenvolver critérios para avaliar a qualidade dos dados, decidindo sobre critérios objetivos de

 ©2026 SBC – Soc. Bras. de Computação

aprovação/reprovação, quando um banco de dados é ou não adequado para uma determinada questão.

E para aqueles que são adequados, precisamos passar por um processo para desenvolver e avaliar fenótipos, para dispor de critérios objetivos de aprovação/reprovação sobre quais desfechos e exposições podem ou não ser estudados de forma confiável em nossos bancos de dados.

E para aqueles que são aprovados, precisamos ter critérios para avaliar a confiabilidade de nossa análise. Precisamos provar que nossos métodos são imparciais, que realmente produzem características operacionais aceitáveis. E se produzirem, somente nesse momento então, estaremos no ponto de estarmos confiantes em gerar resultados.

O desenvolvimento e avaliação de um sistema que gera evidências de qualidade é o objetivo e desafio principal da OHDSI como comunidade.

Como parte desse sistema, um conjunto de componentes em R foi desenvolvido para auxiliar as tarefas de validação de cada uma das etapas. O projeto HADES (Health-Analytics Data to Evidence Suite) agrupa todos estes componentes.

Um dos componentes é o ACHILLES (Automated Characterization of Health Information at Large-scale Longitudinal Evidence Systems) (Huser et al. 2018). O ACHILLES é uma ferramenta de software que fornece caracterização e visualização de um banco de dados em conformidade com o CDM e gera métricas que permitem validar a qualidade dele.

Validação de coortes: OHDSI Keeper

A validação de fenótipos é importante porque nem sempre uma seleção de uma coorte pode se basear apenas na presença ou ausência de um determinado diagnóstico. Temos que lembrar que a pesquisa observacional utiliza dados coletados com outras finalidades que não a pesquisa.

O uso secundário do dado assistencial se constitui como uma fonte de informação importante para pesquisa de desfechos, porém apresenta uma série de desafios metodológicos peculiares a este tipo de fonte de dados (Abrahão M.T, et al, 2019)

Muitas vezes dados de exames laboratoriais ou outras evidências como prescrições de medicamentos específicos indicam um possível candidato a uma coorte mesmo que o diagnóstico correspondente não tenha sido registrado. Desfechos classificados incorretamente podem enviesar os resultados das análises.

É recomendado revisar cada caso potencial (FDA - regulatory information) e no caso de isso ser inviável, selecionar uma amostra para mensurar o desempenho operacional (PPV, sensibilidade) do fenótipo e realizar análises quantitativas de viés para ajustar os resultados.

Precisamos garantir métodos confiáveis para medir o poder de seleção da nossa definição de coorte (fenótipo), comparar com um Gold Standard e calcular o seu Valor

 ©2026 SBC – Soc. Bras. de Computação

Preditivo Positivo (PPV), quer dizer, quantos casos a nossa corte selecionou do Gold Standard (PPV = True Positives / (True Positives + False Positives)). A Figura 1.7 ilustra a matriz de confusão para cálculo do PPV.

		Gold Standard: True	Gold Standard: False
Cohort Definition	True	True Positive	False Positive
	False	False Negative	True Negative

Figura 1.7 Matriz de confusão

Em particular, para desenvolver e principalmente validar fenótipos e cuidar destas situações, foi construído o Knowledge-Enhanced Electronic Profile Review (KEEPER).

O **OHDSI Keeper** é uma ferramenta (pacote R) desenvolvida pela comunidade OHDSI, projetada para otimizar e escalar o processo de validação de casos em estudos observacionais de saúde (OHDSI 2023 Plenary). Ele auxilia na validação de fenótipos (definição de coortes), permitindo a revisão de perfis de pacientes. Ele suporta a avaliação da concordância entre a revisão humana e métodos automatizados (como LLMs ou algoritmos) utilizando métricas como o Kappa de Cohen.

Tradicionalmente, validar se um paciente realmente teve uma condição específica (como diabetes mellitus tipo II), exige que revisores humanos leiam prontuários médicos complexos, o que é lento e caro. O Keeper atua como uma ponte eficiente entre os dados estruturados e a revisão detalhada.

Os princípios do KEEPER são: (Schuemie, 2025)

- Adesão ao raciocínio clínico: O KEEPER aplica princípios e etapas gerais do raciocínio clínico diagnóstico.
- Padronização: Tanto a entrada quanto a saída são padronizadas entre fontes de dados e diagnósticos.
- Redução da dimensionalidade: Extrair apenas informações relevantes.

Funcionalidades do Keeper:

- Extrai dados em nível de paciente para:
  a) uma amostra aleatória de pacientes em uma coorte ou
  b) pacientes em uma lista especificada pelo usuário
  e os formata de acordo com os princípios do KEEPER.
- Permite a revisão de perfis de pacientes por humanos por meio de um aplicativo Shiny interativo.
- Permite a revisão de perfis de pacientes por ferramentas de IA (LLMs).

Roteiro do funcionamento do OHDSI Keeper:

1. **Definição de Conceitos (Concept Sets)**: O profissional define quais diagnósticos, sintomas e medicamentos são relevantes para o fenótipo que deseja validar.

 ©2026 SBC – Soc. Bras. de Computação

2. **Extração de dados do Modelo OMOP**: O sistema utiliza dados já padronizados no CDM OMOP. Extrai informações relevantes de um paciente, como diagnósticos, medicamentos, procedimentos e medições laboratoriais.
3. **Geração de Relatórios Padronizados**: Em vez de obrigar o revisor a navegar por sistemas de prontuário eletrônico (PEP) nativos e confusos, o Keeper gera um "resumo" estruturado e padronizado do histórico do paciente.
4. **Apoio à Decisão com LLMs**: Versões modernas do Keeper estão integradas a Modelos de Linguagem de Grande Escala (LLMs). Isso permite que a ferramenta analise os dados e forneça uma "sugestão" de diagnóstico para o revisor, apontando evidências nos dados que confirmam ou descartam o caso.
5. **Aumento de Escala**: Ao fornecer um formato de saída padronizado e o auxílio de IA, o Keeper permite que os pesquisadores validem centenas de casos em uma fração do tempo que levaria uma revisão manual tradicional, mantendo um nível de concordância similar ao de especialistas humanos (Schuemie, 2025).
6. **Fluxo de Consenso:**
   Para estudos de alta qualidade (FDA - regulatory information), recomenda-se:
   - **Revisão em Dupla**: Dois revisores analisam os mesmos IDs de pacientes.
   - **Adjudicação**: Um terceiro revisor (mais experiente) resolve as divergências quando os dois primeiros discordam.
7. Para centralizar as decisões de múltiplos revisores, o **OHDSI Keeper** geralmente utiliza uma estrutura de banco de dados ou arquivos de log compartilhados. Isso permite o cálculo da taxa de concordância (inter-rater reliability).

Principais Benefícios do Keeper

- **Eficiência**: Reduz a necessidade de exploração extensiva de prontuários brutos.
- **Escalabilidade**: Permite a realização de estudos com coortes muito maiores.
- **Reprodutibilidade**: Como os dados vêm do modelo OMOP, a metodologia de validação pode ser aplicada em diferentes instituições que usem o mesmo padrão. (Abrahão, M. T., 2019)

A primeira etapa, a definição e seleção de conceitos (Concept Sets), é quem guia a qualidade final do fenótipo. Para estruturar os conjuntos de conceitos, precisa focar em capturar os elementos que um médico usaria para confirmar um diagnóstico. O Keeper não precisa apenas do código da doença, mas de todo o contexto clínico ao redor dela para facilitar a revisão. (Schuemie, 2025; Ostropolets, 2023).

Estrutura Recomendada para o Keeper

É recomendado dividir os Concept Sets em categorias que reflitam o raciocínio clínico: (Schuemie, 2025)

1. **Apresentação Clínica (Sintomas/Sinais)**: Inclua conceitos de queixas e achados físicos relacionados (ex: "dor abdominal", "febre").
2. **Histórico e Comorbidades**: Fatores de risco ou condições prévias que tornam o diagnóstico plausível.

 ©2026 SBC – Soc. Bras. de Computação

3. **Procedimentos Diagnósticos**: Exames de imagem, biópsias ou testes laboratoriais específicos (ex: "colonoscopia", "níveis de troponina").
4. **Tratamentos e Medicamentos**: Medicamentos usados para tratar a condição, que servem como evidência de suporte.
5. **Diagnósticos Diferenciais**: Conceitos que podem "confundir" o revisor e precisam ser monitorados para exclusão.

Como Criar os Concept Sets

Existem três formas principais de gerar esses conjuntos:

- **Uso do Atlas OHDSI**: É a forma mais visual. A partir de uma busca dos termos no Atlas (Atlas Tutorial) seleciona os conceitos padrão (Standard Concepts) e exporta no formato JSON para usar no R.
- **Geração Automática (Função generateKeeperConceptSets)** (OHDSI Keeper Doc.): O próprio pacote Keeper possui uma função experimental que usa LLMs para sugerir os conjuntos de conceitos baseados no nome da doença.
- **Via R (Capr)**: O pacote Capr (Concept Sets in Capr) permite criar conjuntos de conceitos programaticamente usando IDs do OMOP.

A estrutura dos Concept Sets deve abranger os seguintes critérios:

1. Diagnóstico Principal (Critério de Entrada)

Incluir o código principal e seus descendentes para capturar variações (ex: DM2 com complicações renais).

2. Laboratórios (Evidência Objetiva)

Estes são cruciais para o Keeper mostrar ao revisor se existe algum exame que indique presença do diagnóstico. Por exemplo, Hemoglobina Glicada para diagnóstico de diabetes.

3. Medicamentos (Tratamento)

A presença desses fármacos ajuda a confirmar que o paciente está sendo tratado especificamente para uma doença em particular.

4. Diagnósticos Diferenciais (Exclusão)

Importante para o Keeper sinalizar se o paciente pode ter sido classificado incorretamente.

O Keeper então processará esses dados e gerará uma interface (ou planilhas) onde o revisor médico poderá marcar "Sim", "Não" ou "Inconclusivo" para cada paciente.

Na pasta de saída podemos encontrar:

- **Arquivos HTML Individuais**: Cada paciente terá uma página exclusiva mostrando a jornada com o diagnóstico selecionado.
- **Timeline Visual**: Uma linha do tempo com os diagnósticos e as prescrições correspondentes.

©2026 SBC – Soc. Bras. de Computação

- **Tabela de Laboratórios**: Os exames relevantes aparecerão destacados no topo da lista.

Tudo isto facilita e padroniza o processo de revisão de casos, a validação por revisores e a apresentação de resultados.

Para gerar o **relatório de concordância** (Kappa de Cohen ou concordância percentual), o **OHDSI Keeper**, através da função:
computeCohortOperatingCharacteristics, analisa as divergências entre os votos dos revisores armazenados na sua tabela de resultados, juntamente com sensibilidade, especificidade, valor preditivo positivo e AUC.

O uso de LLM no sistema de adjudicação do Keeper obteve uma performance razoável, com um grau de concordância com o Gold Standard semelhante aos revisores humanos (Schuemie, 2025).

O uso combinado do KEEPER com LLMs facilita a rápida avaliação de grandes volumes de casos, permitindo o cálculo do PPV e da sensibilidade para inúmeros fenótipos e fontes de dados. Devido a preocupações com a privacidade dos dados dos pacientes, para não transmitir perfis para fora da instituição, são utilizados LLMs hospedados localmente.

Embora o uso de Modelos de Linguagem de Grande Escala (LLMs) na prática clínica ainda seja controverso, este caso de uso que visa aumentar a confiabilidade das evidências a partir de dados observacionais, parece promissor e de baixo risco.

1.6 Considerações finais e conclusões

Em síntese, a pesquisa observacional baseada em dados do mundo real assume um papel cada vez mais estratégico na produção de evidências científicas aplicáveis à prática clínica e à gestão em saúde. Nesse cenário, o ecossistema OMOP/OHDSI destaca-se como uma iniciativa fundamental para enfrentar os desafios de heterogeneidade, escalabilidade e reprodutibilidade inerentes aos dados observacionais, ao oferecer um modelo comum de dados, vocabulários padronizados e ferramentas analíticas consolidadas. Ao converter dados heterogêneos, provenientes de diferentes sistemas e formatos, em uma estrutura comum e padronizada, o OMOP cria uma base sólida para análises consistentes e comparáveis.

Essa padronização viabiliza a realização de análises reprodutíveis, estudos multicêntricos e modelos preditivos em larga escala, permitindo que diferentes instituições utilizem a mesma lógica analítica sobre bases distintas. Dessa forma, o CDM OMOP se estabelece como um pilar essencial para iniciativas de Real World Evidence, projetos de saúde digital e estratégias de análises avançadas.

 ©2026 SBC – Soc. Bras. de Computação

Numa época com destaque aos avanços contínuos nas tecnologias de aprendizado de máquina e Inteligência Artificial (IA), fica fácil esquecer que nada disso seria possível sem dados e evidencias confiáveis que alimentem e treinem estes algoritmos.

A iniciativa da OHDSI se integra em perfeita sincronia com o avanço destas tecnologias fornecendo o que elas mais precisam: dados padronizados e evidências confiáveis, para poder realizar o potencial destes avanços. Se hoje podemos integrar a IA no fluxo do atendimento com confiabilidade, é porque o esforço da OHDSI para melhorar a qualidade de informação médica começou há mais de 10 anos atrás.

Veremos num futuro próximo, sistemas médicos que incluem estas tecnologias de IA no dia a dia do atendimento clínico, sem perceber o imenso valor e importância da infraestrutura da OHDSI gerando evidências confiáveis e seguras para manter o treinamento da IA atualizado.

A Inteligência Artificial no ecossistema OHDSI é utilizada como apoio à detecção, definição e validação de fenótipos clínicos, permitindo identificar padrões complexos em grandes volumes de dados observacionais padronizados no CDM OMOP. Por meio de técnicas de aprendizado de máquina, a IA auxilia na descoberta de fenótipos, na construção de modelos preditivos mais sensíveis e específicos e na avaliação do desempenho das definições fenotípicas. Integrada ao ATLAS, a IA pode sugerir conjuntos de conceitos e regras temporais, sempre com validação por especialistas clínicos. Além disso, possibilita análises distribuídas e reprodutíveis, respeitando a governança e a privacidade dos dados. Dessa forma, a IA atua como um acelerador da fenotipagem computacional, complementando o rigor metodológico da OHDSI e fortalecendo a geração de evidências do mundo real.

Por fim, conclui-se que o ecossistema OMOP/OHDSI representa uma infraestrutura essencial para o presente e o futuro da pesquisa observacional em saúde. Sua relevância tende a se intensificar à medida que sistemas de saúde incorporam, de forma cada vez mais integrada, análises avançadas e soluções baseadas em inteligência artificial no cuidado assistencial. Muitas dessas aplicações ocorrerão de maneira quase invisível ao usuário final, mas apoiadas em uma base sólida, confiável e sustentável de padronização de dados e geração de evidências, justamente o legado e a contribuição contínua da comunidade OHDSI para a ciência e a saúde global.

1.7 Referências bibliográficas

Abrahão, M. T., Nobre, M. R., Madril, P. J., O estado da arte em pesquisa observacional de dados de saúde: A iniciativa OHDSI 2019.
https://books-sol.sbc.org.br/index.php/sbc/catalog/book/29
https://books-sol.sbc.org.br/index.php/sbc/catalog/view/29/98/248

ATLAS Tutorial: Explore Concept Sets
Video: https://www.youtube.com/watch?v=mfjxNwn3KkM

 ©2026 SBC – Soc. Bras. de Computação

Ricotta EE, Bustos Carrillo FA, Angelli-Nichols S, Barugahare J, Benton A, Carlson CJ, et al. Observational research in epidemic settings: a roadmap to reform. BMJ Global Health. 2025;10:e017981. https://doi.org/10.1136/bmjgh-2024-017981

Concept Sets in Capr: https://ohdsi.github.io/Capr/articles/Capr-conceptSets.html

Columbia DBMI: OHDSI Sets Path for Enhancing Trust in Science, Engaging Global Community at 2025 Symposium. COLUMBIA UNIVERSITY, Department of Biomedical Informatics (DBMI). 2025.
Disponível em: https://www.dbmi.columbia.edu/ohdsi-2025/

Dekkers OM, Egger M, Altman DG, Vandenbroucke JP. Distinguishing case series from cohort studies. Ann Intern Med. 2012 Jan 3;156(1 Pt 1):37-40. doi: 10.7326/0003-4819-156-1-201201030-00006. PMID: 22213493.

FDA: Food and Drug Administration. Real-World Evidence. 2023–2025.
https://www.fda.gov/science-research/science-and-research-special-topics/real-world-evidence

FDA – regulatory information: Real-World Data: Assessing Electronic Health Records and Medical Claims Data to Support Regulatory Decision-Making for Drug and Biological Products Guidance for Industry
Final version: https://www.fda.gov/media/152503/download
More info: https://www.fda.gov/regulatory-information/search-fda-guidance-documents/real-world-data-assessing-electronic-health-records-and-medical-claims-data-support-regulatory

Hernán, M. A., & Robins, J. M. Causal Inference: What If. Chapman & Hall/CRC, 2025. https://miguelhernan.org/whatifbook chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://static1.squarespace.com/static/675db8b0dd37046447128f5f/t/691fb7706ce66332f0b44467/1763686256720/hernan robins_WhatIf_21nov25.pdf

Hripcsak G, Duke JD, Shah NH, Reich CG, Huser V, Schuemie MJ, Suchard MA, Park RW, Wong IC, Rijnbeek PR, van der Lei J, Pratt N, Norén GN, Li YC, Stang PE, Madigan D, Ryan PB. Observational Health Data Sciences and Informatics (OHDSI): Opportunities for Observational Researchers. Stud Health Technol Inform. 2015;216:574-8. PMID: 26262116; PMCID: PMC4815923.

 ©2026 SBC – Soc. Bras. de Computação

Huser V, Kahn MG, Brown JS, Gouripeddi R. Methods for examining data quality in healthcare integrated data repositories. Pac Symp Biocomput. 2018;23:628-633. PMID: 29218922.

Kahn MG, Callahan TJ, Barnard J, Bauck AE, Brown J, Davidson BN, Estiri H, Goerg C, Holve E, Johnson SG, Liaw ST, Hamilton-Lopez M, Meeker D, Ong TC, Ryan P, Shang N, Weiskopf NG, Weng C, Zozus MN, Schilling L. A Harmonized Data Quality Assessment Terminology and Framework for the Secondary Use of Electronic Health Record Data. EGEMS (Wash DC). 2016 Sep 11;4(1):1244. doi: 10.13063/2327-9214.1244. PMID: 27713905; PMCID: PMC5051581.

Madigan D, Ryan PB, Schuemie M. Does design matter? Systematic evaluation of the impact of analytical choices on effect estimates in observational studies. Ther Adv Drug Saf. 2013 Apr;4(2):53-62. doi: 10.1177/2042098613477445. PMID: 25083251; PMCID: PMC4110833.

McDonald CJ, Humphreys BL. The U.S. National Library of Medicine and standards for electronic health records: One thing led to another. Inf Serv Use. 2022 May 10;42(1):81-94. doi: 10.3233/ISU-210142. PMID: 35600128; PMCID: PMC9108563.

OHDSI (OBSERVATIONAL HEALTH DATA SCIENCES AND INFORMATICS). Who We Are. 2026. Disponível em: https://www.ohdsi.org/who-we-are/

OHDSI Keeper. Repositório Github: https://github.com/OHDSI/Keeper/ Documentação: https://ohdsi.github.io/Keeper/ Manual: https://raw.githubusercontent.com/OHDSI/Keeper/main/extras/Keeper.pdf

OHDSI 2023 Plenary: Improving the reliability and scale of case validation Anna Ostropolets, Martijn Schuemie, Patrick Ryan.
Apresentação:
https://www.ohdsi.org/wp-content/uploads/2023/10/OHDSI2023-Plenary-1.pdf
video: https://www.youtube.com/watch?v=scUssf863TI

Ostropolets A, Hripcsak G, Husain SA, Richter LR, Spotnitz M, Elhussein A, Ryan PB. Scalable and interpretable alternative to chart review for phenotype evaluation using standardized structured data from electronic health records. J Am Med Inform Assoc. 2023 Dec 22;31(1):119-129. doi: 10.1093/jamia/ocad202. PMID: 37847668; PMCID: PMC10746303.

 ©2026 SBC – Soc. Bras. de Computação

Overhage JM, Ryan PB, Reich CG, Hartzema AG, Stang PE. Validation of a common data model for active safety surveillance research. J Am Med Inform Assoc. 2012 Jan-Feb;19(1):54-60. doi: 10.1136/amiajnl-2011-000376. Epub 2011 Oct 28. PMID: 22037893; PMCID: PMC3240764.

Porta, M. (Ed.). A Dictionary of Epidemiology. Oxford University Press, 2014. ISBN: 9780197663639

Schuemie MJ, Ryan PB, DuMouchel W, Suchard MA, Madigan D. Interpreting observational studies: why empirical calibration is needed to correct p-values. Stat Med. 2014 Jan 30;33(2):209-18. doi: 10.1002/sim.5925. Epub 2013 Jul 30. PMID: 23900808; PMCID: PMC4285234.

Schuemie MJ, Hripcsak G, Ryan PB, Madigan D, Suchard MA. Robust empirical calibration of p-values using observational data. Stat Med. 2016 Sep 30;35(22):3883-8. doi: 10.1002/sim.6977. PMID: 27592566; PMCID: PMC5108459.

Schuemie MJ, Hripcsak G, Ryan PB, Madigan D, Suchard MA. Empirical confidence interval calibration for population-level effect estimation studies in observational healthcare data. Proc Natl Acad Sci U S A. 2018 Mar 13;115(11):2571-2577. doi: 10.1073/pnas.1708282114. PMID: 29531023; PMCID: PMC5856503.

Schuemie MJ, Ryan PB, Hripcsak G, Madigan D, Suchard MA. Improving reproducibility by using high-throughput observational studies with empirical calibration. Philos Trans A Math Phys Eng Sci. 2018 Sep 13;376(2128):20170356. doi: 10.1098/rsta.2017.0356. PMID: 30082302; PMCID: PMC6107542.

Schuemie, M.J., Ostropolets, A., Zhuk, A. et al. Standardized patient profile review using large language models for case adjudication in observational research. npj Digit. Med. 8, 18 (2025). https://doi.org/10.1038/s41746-025-01433-4
Apresentação:http://www.airis.or.kr/file/S4_4_Martijn%20Schuemie.pdf
Video: https://www.youtube.com/watch?v=_fl6v46BjIA&t=3734s

Stang PE, Ryan PB, Racoosin JA, et al. Advancing the science for active surveillance: rationale and design for the Observational Medical Outcomes Partnership. Annals of Internal Medicine. 2010 Nov;153(9):600-606. DOI: 10.7326/0003-4819-153-9-201011020-00010. PMID: 21041580.

Sterrantino, A. Observational studies: practical tips for avoiding common statistical pitfalls. The Lancet Regional Health - Southeast Asia, 2024; 25 https://www.thelancet.com/journals/lansea/article/PIIS2772-3682(24)00065-9/fulltext

 ©2026 SBC – Soc. Bras. de Computação

STROBE Initiative. Strengthening the Reporting of Observational Studies in Epidemiology. https://www.strobe-statement.org/

 ©2026 SBC – Soc. Bras. de Computação

Capítulo 2

Interpretação de Modelos de Aprendizado de Máquina Aplicados à Saúde com SHAP

Letícia Martins Raposo, Vinicius Navega Stelet

Abstract

The growing adoption of complex machine learning models in healthcare has substantially enhanced the predictive capability of computational systems while intensifying challenges related to interpretability and transparency of automated decisions. In this context, explainable artificial intelligence methods have become essential to foster trust, enable clinical validation, and support the responsible use of such models in sensitive domains. This chapter presents the theoretical, methodological, and practical foundations of predictive model interpretation through SHAP (SHapley Additive exPlanations), a method grounded in Cooperative Game Theory. Core concepts of interpretability and black-box models are discussed, followed by the mathematical formalization of Shapley values, the axiomatic properties of SHAP, and its formulation as an additive explanation model. The chapter examines local and global interpretations, implementation in R and Python with a cardiovascular risk case study, visualization strategies, and result analysis, as well as limitations, methodological challenges, best practices, and ethical and regulatory implications for the responsible use of SHAP in health computing.

Resumo

A crescente adoção de modelos complexos de aprendizado de máquina em saúde amplia a capacidade preditiva dos sistemas computacionais, mas intensifica os desafios de interpretabilidade e transparência das decisões automatizadas. Nesse contexto, métodos de inteligência artificial explicável tornaram-se fundamentais para promover confiança, validação clínica e uso responsável desses modelos em domínios sensíveis. Este capítulo apresenta os fundamentos teóricos, metodológicos e práticos da interpretação de modelos preditivos por meio do SHAP (SHapley Additive exPlanations), método fundamentado na Teoria dos Jogos Cooperativos. São discutidos os conceitos centrais de interpretabilidade e modelos caixa-preta, a formalização matemática dos valores de Shapley, as propriedades

 ©2026 SBC – Soc. Bras. de Computação

axiomáticas do SHAP e sua formulação como modelo de explicação aditivo. O capítulo examina interpretações locais e globais, implementação em R e Python com estudo de caso em risco cardiovascular, estratégias de visualização e análise de resultados, além de limitações, desafios metodológicos, boas práticas e implicações éticas e regulatórias para o uso responsável do método em computação aplicada à saúde.

2.1. Introdução à Interpretabilidade em Aprendizado de Máquina para Saúde

Nas últimas décadas, o aprendizado de máquina tem se consolidado como uma ferramenta central na área da saúde, impulsionado pela crescente disponibilidade de dados clínicos digitais, pelo avanço do poder computacional e pelo desenvolvimento de algoritmos capazes de lidar com grandes volumes de dados heterogêneos. Modelos preditivos baseados em *ensembles*, redes neurais profundas e técnicas avançadas de otimização têm demonstrado desempenho competitivo em tarefas específicas, embora com variação importante entre contextos, dados e critérios de validação [Miotto et al. 2018, Rajkomar et al. 2018, Topol 2019].

Esses avanços se manifestam em múltiplos domínios da informática em saúde. Em prontuários eletrônicos, modelos preditivos são empregados para identificar pacientes de alto risco, antecipar reinternações e otimizar a alocação de recursos hospitalares [Goldstein et al. 2017, Mahmoudi et al. 2020]. Em imagens médicas, redes neurais convolucionais atingem desempenho comparável ou superior ao de especialistas humanos na detecção de retinopatia diabética [Bajwa et al. 2023], na classificação de lesões dermatológicas [Esteva et al. 2017] e na análise de exames radiológicos [Chen et al. 2024]. Na análise de biossinais, algoritmos de aprendizado de máquina têm sido aplicados com sucesso à detecção precoce de arritmias [Hannun et al. 2019] e de crises epilépticas [Daoud and Bayoumi 2019].

Apesar desses resultados promissores, o aumento do desempenho preditivo é frequentemente acompanhado por uma redução da transparência dos modelos. Muitos dos algoritmos mais eficazes operam como *modelos caixa-preta*, cujos mecanismos internos são difíceis de compreender mesmo para especialistas em ciência de dados [Lipton 2018, Binzagr 2024]. No contexto da saúde, essa opacidade é particularmente problemática, uma vez que decisões automatizadas podem influenciar diretamente diagnósticos, condutas terapêuticas e desfechos clínicos.

Nesse cenário, a interpretabilidade em aprendizado de máquina emerge como um campo de pesquisa fundamental para aplicações em saúde. Neste capítulo, adotamos *interpretabilidade* como a capacidade de compreender, em termos humanamente significativos, como um modelo relaciona entradas e saídas, e *explicabilidade* como o conjunto de métodos empregados para tornar esse comportamento inteligível, especialmente em modelos complexos. De forma mais específica, a interpretabilidade pode ser entendida como o grau em que um ser humano — seja um profissional de saúde, pesquisador ou gestor — consegue compreender as razões pelas quais um modelo produz determinada saída a partir de suas entradas clínicas [Stiglic et al. 2020]. Embora não exista uma definição única e universalmente aceita, o conceito está intimamente relacionado à possibilidade de atribuir significado clínico às decisões do modelo e de estabelecer relações compreensíveis entre variáveis de entrada e predições [Molnar 2020].

A relevância da interpretabilidade em aplicações médicas vai além de considerações

 ©2026 SBC – Soc. Bras. de Computação

estritamente técnicas. Em contextos clínicos sensíveis, como triagem em unidades de emergência, estratificação de risco para doenças crônicas, diagnóstico assistido por computador e sistemas de alerta precoce em unidades de terapia intensiva, compreender os fatores que sustentam uma predição é tão importante quanto a própria acurácia do modelo [Tonekaboni et al. 2019]. Explicações interpretáveis são essenciais para promover confiança, possibilitar auditoria clínica, apoiar a tomada de decisão compartilhada e facilitar a adoção segura de sistemas baseados em aprendizado de máquina.

Além disso, a interpretabilidade desempenha papel central em debates éticos e regulatórios. Modelos opacos podem perpetuar vieses e desigualdades sem que tais problemas sejam facilmente detectáveis, como demonstrado em estudos que identificaram vieses raciais em algoritmos amplamente utilizados em sistemas de saúde [Obermeyer et al. 2019]. Paralelamente, legislações recentes, como o Regulamento Geral sobre a Proteção de Dados (GDPR, em inglês) na União Europeia, a Lei Geral de Proteção de Dados Pessoais (LGPD) no Brasil e regulamentações específicas para *Software as a Medical Device* (SaMD) por agências como FDA e ANVISA, reforçam a necessidade de transparência, explicabilidade e justificativa das decisões automatizadas em saúde [Goodman and Flaxman 2017, U.S. Food and Drug Administration 2021].

Nesse contexto, métodos de explicabilidade *post hoc* têm ganhado destaque por permitirem interpretar modelos complexos sem comprometer seu desempenho preditivo. Entre esses métodos, o SHAP (*SHapley Additive exPlanations*) destaca-se por sua fundamentação teórica na Teoria dos Jogos Cooperativos e por fornecer explicações consistentes, locais e globais para uma ampla classe de modelos [Lundberg and Lee 2017]. Sua capacidade de atribuir contribuições individuais às variáveis de entrada o torna particularmente útil em dados tabulares estruturados, frequentes em prontuários eletrônicos, desde que a interpretação produzida seja compreendida como explicação do comportamento do modelo e não como evidência causal sobre o fenômeno clínico.

O argumento central deste capítulo é que o SHAP pode ampliar a transparência e a auditabilidade de modelos preditivos em saúde, sem eliminar limitações relacionadas à qualidade dos dados, ao viés algorítmico, à dependência das escolhas metodológicas e à diferença entre associação preditiva e interpretação causal. Com esse enquadramento, o capítulo parte da caracterização dos modelos caixa-preta e da motivação para métodos de explicabilidade *post hoc*, avança para os fundamentos matemáticos dos valores de Shapley e sua formulação no SHAP, e então apresenta sua implementação prática em R e Python em um estudo de caso sobre risco cardiovascular. As seções finais discutem limitações, boas práticas e implicações éticas e regulatórias para o uso responsável do método em contextos clínicos.

2.2. Modelos Caixa-Preta e Métodos de Explicação

O uso crescente de modelos de aprendizado de máquina em saúde levanta uma questão central: em que medida decisões automatizadas podem ser consideradas confiáveis quando seus mecanismos internos não são transparentes para os usuários humanos? Esta seção examina o problema da opacidade em modelos complexos, suas implicações no contexto clínico e as principais estratégias desenvolvidas no campo de *Explainable Artificial Intelligence* (XAI), culminando na justificativa do uso do método SHAP.

 ©2026 SBC – Soc. Bras. de Computação

2.2.1. Modelos caixa-preta e opacidade estrutural

Modelos caixa-preta são sistemas cujos processos internos, embora formalmente especificados, não são acessíveis à interpretação humana em razão de sua complexidade estrutural, interações não lineares e alta dimensionalidade [Burrell 2016, Bodria et al. 2021, Hassija et al. 2023]. Em aplicações clínicas, essa opacidade manifesta-se de diferentes formas. Redes neurais profundas aplicadas à análise de imagens médicas envolvem milhões de parâmetros distribuídos em múltiplas camadas convolucionais, tornando inviável a interpretação direta de seus pesos [Mienye et al. 2025]. Modelos baseados em arquiteturas *transformer*, utilizados em processamento de linguagem natural sobre registros clínicos, apresentam complexidade ainda maior, sendo empregados em tarefas como codificação automatizada e detecção de eventos adversos [Yang et al. 2022, Yuan et al. 2025]. De modo análogo, métodos de *ensemble*, como florestas aleatórias e algoritmos de *gradient boosting*, combinam múltiplos modelos base, dificultando a análise interpretativa de seu comportamento agregado [Stiglic et al. 2020].

A opacidade de modelos complexos não implica ausência de rigor metodológico, mas sim uma limitação epistêmica na tradução de seus mecanismos internos em explicações compreensíveis. Essa limitação possui implicações diretas na prática clínica: evidências indicam que profissionais de saúde tendem a resistir ou reinterpretar recomendações provenientes de sistemas cujo funcionamento não compreendem, mesmo quando estes apresentam alto desempenho preditivo [Tonekaboni et al. 2019, Tun et al. 2024].

2.2.2. Interpretabilidade intrínseca e seus limites

Uma resposta tradicional a esse problema consiste no uso de modelos interpretáveis por construção, cuja estrutura permite a compreensão direta de suas decisões. Exemplos clássicos na medicina incluem escores de risco como o SCORE2, o CHA_2DS_2–VASc e o SOFA, nos quais os parâmetros possuem interpretações clínicas claras [working group and risk collaboration 2021, Lip et al. 2010, Vincent et al. 1996]. Contudo, a adoção desses modelos impõe um compromisso entre interpretabilidade e capacidade preditiva.

Em cenários caracterizados por alta dimensionalidade e relações não lineares — como prontuários eletrônicos e séries temporais clínicas — modelos simples frequentemente apresentam desempenho preditivo inferior [Patharkar et al. 2024]. Esse compromisso torna-se particularmente problemático em contextos nos quais tanto a acurácia quanto a interpretabilidade possuem valor crítico.

2.2.3. Explicabilidade em modelos preditivos e métodos pós-hoc

Como resposta a esse problema, métodos de explicação pós-hoc têm sido amplamente desenvolvidos com o objetivo de tornar modelos complexos mais interpretáveis sem modificar sua estrutura interna. Tais métodos produzem explicações a partir da relação entre entradas e saídas do modelo, tratando-o, em muitos casos, como uma função caixa-preta [Hassija et al. 2023, Markus et al. 2021].

Do ponto de vista regulatório, essas abordagens são particularmente relevantes para atender exigências de transparência e prestação de contas em sistemas automatizados [Goodman and Flaxman 2017, U.S. Food and Drug Administration 2021]. No

 ©2026 SBC – Soc. Bras. de Computação

contexto clínico, permitem a integração de recomendações automatizadas ao raciocínio médico, além de possibilitar a identificação de vieses, inconsistências e potenciais riscos associados ao uso dessas tecnologias [Tonekaboni et al. 2019, Amann et al. 2020, Obermeyer et al. 2019].

Entretanto, a adoção de métodos pós-hoc introduz uma tensão fundamental: as explicações produzidas constituem aproximações do comportamento do modelo, e não descrições fiéis de seus mecanismos internos. Como consequência, explicações de baixa fidelidade podem induzir interpretações equivocadas, com implicações clínicas potencialmente graves [Rudin 2019]. Esse problema evidencia a necessidade de critérios analíticos rigorosos para classificar, avaliar e selecionar métodos de explicação.

Nesse contexto, a literatura em XAI converge para a identificação de duas dimensões analíticas centrais: o **escopo da explicação** e o **grau de dependência do modelo**. Essas dimensões estruturam diferentes formas de acesso ao comportamento de modelos complexos e permitem compreender as limitações e potencialidades das abordagens existentes.

No que se refere ao escopo, distinguem-se **explicações globais** e **locais**. Explicações globais buscam caracterizar o comportamento médio do modelo ao longo da distribuição de dados, permitindo identificar padrões gerais, importância média de variáveis e relações funcionais predominantes. Essas abordagens são úteis para avaliar plausibilidade clínica, detectar vieses sistemáticos e apoiar validação externa, mas podem obscurecer comportamentos raros ou não lineares relevantes em subpopulações específicas [Hakkoum et al. 2024, Greenwell 2017, Lundberg et al. 2020].

Por outro lado, **explicações locais** focalizam instâncias individuais, decompondo predições específicas em contribuições das variáveis de entrada ou aproximando o modelo em uma vizinhança restrita do espaço de dados. Essa perspectiva é particularmente relevante em decisões clínicas centradas no paciente, nas quais a justificativa individual é essencial [Hakkoum et al. 2024, Salih et al. 2023, Vimbi et al. 2024]. Contudo, tais explicações podem apresentar instabilidade e limitada representatividade global, exigindo cautela em sua interpretação [Zafar and Khan 2021].

Paralelamente, a distinção entre métodos **específicos do modelo** (*model-specific*) e **agnósticos ao modelo** (*model-agnostic*) constitui uma segunda dimensão analítica relevante. Métodos específicos exploram propriedades internas de determinadas arquiteturas (por exemplo, árvores de decisão ou redes neurais convolucionais), frequentemente alcançando maior fidelidade explicativa e eficiência computacional dentro de seu domínio [Roshinta and Gábor 2024, Belle and Papantonis 2020, Lundberg et al. 2020]. No entanto, sua aplicabilidade é limitada à arquitetura para a qual foram desenvolvidos.

Em contraste, métodos agnósticos operam exclusivamente sobre a relação entrada–saída, ampliando sua aplicabilidade a diferentes modelos e contextos [Elshawi et al. 2019, Ribeiro et al. 2016]. Essa flexibilidade, contudo, é frequentemente acompanhada por maior custo computacional, possíveis problemas de estabilidade e limitações na captura de propriedades estruturais profundas do modelo [Roshinta and Gábor 2024, Lundberg et al. 2020].

À luz dessas dimensões, as principais estratégias de explicação podem ser compreendidas como respostas parciais a um problema comum: tornar modelos complexos

 ©2026 SBC – Soc. Bras. de Computação

inteligíveis sem comprometer sua capacidade preditiva. Métodos baseados em importância de variáveis, como importância por permutação ou medidas derivadas de árvores, oferecem interpretações globais intuitivas, mas são sensíveis à correlação entre preditores e não capturam heterogeneidade individual [Greenwell and Boehmke 2020, Hooker et al. 2019]. Abordagens baseadas em modelos substitutos locais, como o *Local Interpretable Model-agnostic Explanations* (LIME), procuram explicar predições individuais por meio de aproximações locais [Ribeiro et al. 2016], mas sua dependência de perturbações aleatórias e da definição de vizinhança pode comprometer a estabilidade e a fidelidade das explicações [Zafar and Khan 2021, Tan et al. 2023].

Por fim, técnicas de visualização, como gráficos de dependência parcial e mapas de saliência, desempenham papel relevante na comunicação dos resultados, mas não resolvem, isoladamente, os desafios associados à fidelidade e à consistência das explicações [Greenwell 2017, Selvaraju et al. 2020, Brenning 2021, Singh et al. 2025]. Em particular, métodos globais podem mascarar efeitos condicionais importantes, enquanto abordagens locais frequentemente carecem de robustez.

2.2.4. SHAP como abordagem teoricamente fundamentada

As limitações discutidas anteriormente — incluindo a tensão entre interpretabilidade e fidelidade, a instabilidade de explicações locais, a sensibilidade a correlações entre variáveis e a ausência de garantias teóricas consistentes — convergem para a necessidade de métodos que conciliem propriedades frequentemente conflitantes: (i) fidelidade ao modelo original, (ii) consistência teórica, (iii) aplicabilidade a diferentes arquiteturas e (iv) capacidade de produzir explicações coerentes em níveis local e global.

É nesse contexto que métodos baseados em valores de Shapley se destacam como uma abordagem formalmente fundamentada. Derivados da teoria dos jogos cooperativos, os valores de Shapley oferecem um esquema de atribuição de importância a variáveis que satisfaz um conjunto de propriedades axiomáticas desejáveis, incluindo eficiência (a soma das contribuições corresponde à predição do modelo), simetria (variáveis com contribuições equivalentes recebem o mesmo valor) e aditividade [Lundberg and Lee 2017, Shapley 1953, Bifarin 2022].

O SHAP (*SHapley Additive exPlanations*) operacionaliza esses princípios no contexto de modelos de aprendizado de máquina, fornecendo uma decomposição aditiva da predição em contribuições atribuíveis a cada variável. Diferentemente de métodos como o LIME, que dependem de aproximações locais potencialmente instáveis, o SHAP estabelece uma conexão direta entre a explicação e propriedades formais do modelo, o que contribui para maior consistência e interpretabilidade [Lundberg et al. 2020, Ribeiro et al. 2016, Kelodjou et al. 2024]. Além disso, a estrutura aditiva dos valores de Shapley permite uma integração natural entre análises locais (explicações de instâncias individuais) e globais (agregação das contribuições ao longo da população).

Do ponto de vista computacional, avanços como o algoritmo TreeSHAP tornam viável a aplicação eficiente dessa abordagem em modelos amplamente utilizados na prática clínica, como árvores de decisão e métodos de *gradient boosting*, preservando exatidão polinomial na estimação dos valores [Lundberg et al. 2020, Nohara et al. 2021, Luo et al. 2024].

 ©2026 SBC – Soc. Bras. de Computação

Apesar de suas vantagens, o SHAP não deve ser interpretado como uma solução definitiva. Embora ofereça garantias teóricas sob certas condições, sua interpretação depende de pressupostos sobre a distribuição e independência das variáveis, e diferentes escolhas metodológicas podem levar a explicações distintas. Ainda assim, o SHAP representa um avanço importante ao combinar fundamentos matemáticos bem definidos com aplicabilidade prática em problemas de interpretabilidade.

Dessa forma, o SHAP pode ser entendido como uma resposta teoricamente fundamentada às limitações das abordagens anteriores, oferecendo um quadro unificado para a análise explicativa de modelos complexos. As seções seguintes apresentam seus fundamentos matemáticos e suas aplicações no contexto da saúde.

2.3. Fundamentos da Teoria dos Jogos e Valores de Shapley

A interpretação de modelos preditivos pelo SHAP baseia-se na Teoria dos Jogos Cooperativos, que fornece um modelo formal para dividir, de forma justa, o resultado de uma cooperação entre diferentes participantes. No contexto de aprendizado de máquina, essa ideia é utilizada para decompor a predição de um modelo em contribuições individuais associadas a cada variável clínica, preservando propriedades matemáticas bem definidas [Shapley 1953, Nohara et al. 2021].

2.3.1. Jogos Cooperativos e Função Característica

Um jogo cooperativo é definido por um par (N, v), em que $N = \{1, 2, \ldots, n\}$ representa o conjunto de jogadores e $v : 2^N \to \mathbb{R}$ é a função característica, que associa um valor a cada subconjunto $S \subseteq N$. Esse subconjunto, chamado de *coalizão*, representa um grupo de jogadores considerados em conjunto. Em aprendizado de máquina aplicado à saúde, os "jogadores" correspondem às variáveis clínicas utilizadas no modelo [Nohara et al. 2021, Ghasemi et al. 2024, Vimbi et al. 2024, Ghosh and Khandoker 2024]. No contexto clínico, isso significa tratar cada variável observada como participante formal da predição produzida pelo modelo, e não como portadora, por si só, de significado causal ou terapêutico [Hettikankanamage et al. 2025].

Por exemplo, em um modelo de risco cardiovascular, os jogadores podem incluir idade, sexo, pressão arterial, colesterol, presença de diabetes e hábitos como tabagismo [working group and risk collaboration 2021]. A função característica $v(S)$ representa o desempenho do modelo quando apenas o conjunto de variáveis S está disponível. Assume-se que $v(\emptyset) = 0$, ou seja, na ausência de informação clínica, o modelo retorna apenas uma predição base, como a prevalência do desfecho na população [Ghosh and Khandoker 2024, Binzagr 2024, Covert et al. 2020, Shapley 1953].

Nesse contexto, o problema de explicabilidade pode ser formulado como um problema de alocação: quanto cada variável contribui para alterar a predição em relação a esse valor base?

Entretanto, calcular $v(S)$ na prática exige lidar com variáveis ausentes, o que implica realizar imputação ou marginalização. Essa escolha não é neutra: diferentes estratégias podem levar a explicações distintas. Em termos práticos, isso significa que a forma como as variáveis não observadas são tratadas pode alterar a explicação obtida. Além disso, ao

 ©2026 SBC – Soc. Bras. de Computação

avaliar o modelo em combinações de variáveis que não ocorrem naturalmente nos dados, podem surgir cenários clinicamente implausíveis, o que dificulta a interpretação dos resultados [Hettikankanamage et al. 2025, Contreras et al. 2024, Hooshyar and Yang 2024].

2.3.2. O Problema da Alocação Justa de Contribuições

A questão central passa a ser: como distribuir o valor total $v(N)$ entre as variáveis de forma justa? Em termos práticos, isso corresponde a decompor uma predição em contribuições atribuíveis a cada variável [Nohara et al. 2021, Ghosh and Khandoker 2024, Salih et al. 2023].

Em aplicações clínicas, a alocação das contribuições possui implicações diretas para a interpretação e a tomada de decisão. Considere um modelo de predição de sepse em UTI que estima uma probabilidade de 78% para um paciente cuja probabilidade base é de 15%. Surge, então, a questão: como atribuir os 63 pontos percentuais de aumento a variáveis como lactato elevado, taquicardia e hipotensão? A decomposição dessa diferença em contribuições individuais constitui exatamente o tipo de explicação local necessário em contextos de alto risco clínico, nos quais a compreensão dos fatores que influenciam a predição é tão importante quanto o valor predito em si [Stenwig et al. 2022, Hu et al. 2022, Hu et al. 2024a].

No entanto, o conceito de "contribuição justa" é definido em termos matemáticos, não clínicos. Duas variáveis podem apresentar contribuições simétricas ao modelo preditivo e, ainda assim, possuir significados clínicos inteiramente distintos — uma associação espúria e um mecanismo fisiopatológico estabelecido podem, em princípio, receber contribuições equivalentes. Isso não invalida o método, mas delimita seu escopo: os valores atribuídos descrevem o comportamento do modelo, mas não necessariamente refletem relações causais no mundo real [Salih et al. 2023, Bifarin 2022, Hettikankanamage et al. 2025].

Os valores de Shapley se destacam porque são a única solução que satisfaz simultaneamente um conjunto de axiomas formais de equidade e consistência [Shapley 1953, Salih et al. 2023, Lundberg and Lee 2017], o que justifica sua adoção no SHAP.

2.3.3. Definição Formal dos Valores de Shapley

Os valores de Shapley, propostos por Shapley em 1953 [Shapley 1953], fornecem uma forma única de distribuir o valor total entre os jogadores. O valor $\phi_i(v)$ associado à variável i é dado por:

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(|N| - |S| - 1)!}{|N|!} \left[v(S \cup \{i\}) - v(S) \right] \tag{1}$$

Intuitivamente, esse valor ($\phi_i(v)$) corresponde à contribuição média da variável i ao longo de todas as possíveis combinações de variáveis. O termo $v(S \cup \{i\}) - v(S)$ mede quanto a inclusão de i melhora a predição, enquanto o fator combinatório pondera essa contribuição considerando todas as ordens possíveis de entrada das variáveis [Shapley 1953, Ghosh and Khandoker 2024].

 ©2026 SBC – Soc. Bras. de Computação

Outra forma de interpretar esse valor ($\phi_i(v)$) é probabilística: ele representa a contribuição média de uma variável quando as variáveis são consideradas em ordem aleatória. No entanto, essa interpretação pressupõe independência entre variáveis, o que raramente ocorre em dados clínicos. Quando há correlação entre preditores, parte da contribuição pode refletir efeitos compartilhados entre variáveis [Hu et al. 2024a, Belle and Papantonis 2020].

2.3.4. Axiomas Caracterizadores

Os valores de Shapley são definidos por quatro axiomas principais:

Eficiência. $\sum_{i \in N} \phi_i(v) = v(N)$. A soma das contribuições reproduz exatamente a predição do modelo. No SHAP, essa propriedade corresponde à *local accuracy* [Lundberg and Lee 2017, Auzine et al. 2024]: a explicação é completa no sentido contábil. Contudo, completude contábil não implica completude causal; a eficiência garante que a predição seja integralmente decomposta, mas não que cada parcela corresponda a um mecanismo clínico independente [Hettikankanamage et al. 2025, Bifarin 2022].

Simetria. Se i e j contribuem igualmente para todas as coalizões, então $\phi_i(v) = \phi_j(v)$. Matematicamente, isso é uma propriedade de equidade formal [Shapley 1953, Bifarin 2022]. Clinicamente, porém, duas variáveis com efeito preditivo idêntico podem ter relevância terapêutica ou fisiopatológica inteiramente distinta. A simetria garante consistência na atribuição de contribuições, mas não substitui o julgamento clínico na interpretação de seu significado [Hettikankanamage et al. 2025].

Jogador nulo. Se a inclusão de i não altera nenhuma coalizão, então $\phi_i(v) = 0$. Variáveis sem poder preditivo não recebem contribuições espúrias — no SHAP, essa ideia corresponde à propriedade de *missingness* [Alabi et al. 2023, Auzine et al. 2024]. Note-se que esse axioma é definido em relação ao modelo, não ao fenômeno clínico: uma variável pode ser um fator causal estabelecido e, ainda assim, receber contribuição nula se sua informação for inteiramente redundante com outros preditores presentes no modelo.

Aditividade. Para dois jogos v e w, $\phi_i(v+w) = \phi_i(v) + \phi_i(w)$. Em termos simples, a contribuição de cada variável em um modelo combinado é igual à soma de suas contribuições em cada modelo separado. Isso permite analisar modelos complexos por partes, mantendo consistência nas explicações. No contexto clínico, em modelos *ensemble*, a explicação final pode ser entendida como a soma das explicações de cada modelo individual [Shapley 1953, Lundberg et al. 2019].

Esses axiomas garantem uma solução formalmente consistente para o problema de atribuição, mas não asseguram que as explicações reflitam relações causais.

2.3.5. Complexidade Computacional e Motivação para Aproximações

O cálculo exato dos valores de Shapley em (1) exige a avaliação de todas as possíveis combinações de variáveis, isto é, 2^n coalizões. Esse crescimento é exponencial: com 20 variáveis clínicas, são necessárias mais de um milhão de avaliações do modelo, enquanto

 ©2026 SBC – Soc. Bras. de Computação

com 30 variáveis esse número ultrapassa um bilhão, tornando a abordagem inviável em aplicações reais [Nohara et al. 2021, Ghosh and Khandoker 2024, Salih et al. 2023]. Diante dessa limitação, foram desenvolvidas estratégias que tornam o SHAP aplicável na prática, destacando-se o KernelSHAP e o TreeSHAP.

O KernelSHAP é um método agnóstico ao modelo que estima os valores de Shapley por meio da amostragem de diferentes coalizões e ajuste de uma regressão linear ponderada [Lundberg et al. 2019, Olsen and Jullum 2025]. Sua principal vantagem é a generalidade, permitindo sua aplicação a uma ampla variedade de modelos, incluindo redes neurais, máquinas de vetor de suporte e *ensembles*. No entanto, essa flexibilidade tem custo: o método pode exigir um número elevado de avaliações do modelo para explicar uma única instância e, por depender de amostragem, introduz variabilidade nas estimativas. Em cenários com muitas variáveis ou forte dependência entre preditores, diferentes execuções podem gerar explicações distintas, o que exige cuidado na configuração dos parâmetros e na análise de robustez [Kelodjou et al. 2024, Olsen and Jullum 2025]. Extensões como DeepSHAP e GradientSHAP procuram reduzir o custo computacional em modelos mais complexos, mas mantêm o caráter aproximado da abordagem [Lundberg and Lee 2017].

O TreeSHAP, por sua vez, explora a estrutura de modelos baseados em árvores para calcular valores SHAP exatos de forma eficiente, com complexidade $O(TLD^2)$ por instância, onde T é o número de árvores, L o número médio de folhas e D a profundidade máxima [Lundberg et al. 2020]. Em modelos de *gradient boosting*, amplamente utilizados em dados clínicos, esse cálculo pode ser realizado em milissegundos por paciente, o que viabiliza sua aplicação em larga escala [Nohara et al. 2021, Luo et al. 2024, Hu et al. 2024a]. Ainda assim, a exatidão computacional não elimina limitações interpretativas: os valores obtidos refletem o comportamento do modelo treinado, incluindo possíveis vieses, padrões espúrios e dependências entre variáveis. Em particular, a presença de multicolinearidade pode levar à redistribuição das contribuições entre variáveis correlacionadas, dificultando a interpretação individual de seus efeitos [Salih et al. 2023, Hu et al. 2024a, Hettikankanamage et al. 2025].

Em síntese, enquanto o KernelSHAP oferece maior flexibilidade ao custo de variabilidade e maior demanda computacional, o TreeSHAP proporciona eficiência e determinismo, mas permanece dependente da estrutura do modelo e dos dados. Essa distinção é central para a interpretação dos resultados, uma vez que os valores SHAP devem ser analisados considerando tanto o método de estimação quanto o modelo subjacente. Em aplicações com dados tabulares de prontuários eletrônicos, nas quais modelos de *gradient boosting* são predominantes, o TreeSHAP tende a ser a escolha mais viável, permitindo a obtenção de explicações locais e globais em escala clínica.

2.4. SHAP: Formulação Teórica e Propriedades

O SHAP (*SHapley Additive Explanations*), proposto por Lundberg e Lee em 2017, utiliza os valores de Shapley para explicar modelos de aprendizado de máquina. A ideia central é decompor a predição de um modelo em contribuições individuais associadas a cada variável, permitindo interpretar de forma transparente como cada fator influencia o resultado [Lundberg and Lee 2017, Nohara et al. 2021, Ghosh and Khandoker 2024].

Essa abordagem fornece uma base matemática rigorosa para atribuir importância

 ©2026 SBC – Soc. Bras. de Computação

às variáveis, sendo aplicável a diferentes tipos de modelos preditivos. Nesta seção, são apresentados os principais elementos da formulação do SHAP, bem como suas propriedades teóricas, incluindo um exemplo ilustrativo no contexto de risco cardiovascular.

Apesar de sua fundamentação formal, é importante destacar que a interpretação dos resultados não é automática. As explicações produzidas pelo SHAP dependem de suposições sobre a distribuição dos dados e sobre como as variáveis interagem entre si no modelo. Assim, mesmo sendo matematicamente consistentes, essas explicações podem variar conforme as escolhas metodológicas adotadas [Hettikankanamage et al. 2025, Belle and Papantonis 2020].

2.4.1. SHAP como Modelo de Explicação Aditivo

O SHAP pertence à classe dos modelos de explicação aditivos, nos quais a predição de um modelo é representada como a soma de contribuições individuais associadas a cada variável [Lundberg and Lee 2017, Binzagr 2024, Alabi et al. 2023]. Seja $f : \mathbb{R}^p \to \mathbb{R}$ um modelo preditivo com p variáveis de entrada. Para uma instância $x \in \mathbb{R}^p$, a explicação SHAP é dada por:

$$f(x) = \phi_0 + \sum_{i=1}^{p} \phi_i(x), \tag{2}$$

onde $\phi_0 = \mathbb{E}[f(X)]$ é o **valor base**, isto é, a predição média do modelo na população de referência, e $\phi_i(x)$ representa a **contribuição SHAP** da variável i, indicando quanto essa variável altera a predição em relação a esse valor base [Nohara et al. 2021, Alabi et al. 2023, Binzagr 2024]. Essa propriedade, conhecida como *local accuracy*, garante que a soma das contribuições individuais seja exatamente igual à predição do modelo, sem deixar parte não explicada [Lundberg and Lee 2017, Auzine et al. 2024, Alabi et al. 2023].

Na prática, isso significa que a predição é "quebrada" em partes atribuídas a cada variável. Essa forma de decomposição facilita a interpretação, pois permite entender como cada fator contribui para o resultado final. No entanto, essa representação também envolve simplificações. Ao distribuir a predição entre variáveis individuais, o SHAP assume que os efeitos podem ser separados dessa maneira, mesmo quando existem interações complexas entre elas [Lundberg et al. 2020, Lundberg et al. 2019].

Por exemplo, em um modelo de risco cardiovascular, a combinação de hipertensão e diabetes pode ter um efeito conjunto relevante. No SHAP, esse efeito é dividido entre as variáveis com base em contribuições médias. Embora essa divisão seja matematicamente bem definida, ela nem sempre corresponde a uma interpretação clínica direta.

Assim, o caráter aditivo do SHAP traz duas consequências importantes: por um lado, torna as explicações mais claras e consistentes; por outro, pode simplificar interações complexas, que na prática não são facilmente separáveis em contribuições independentes.

2.4.2. Definição da Função Característica

Para aplicar os valores de Shapley em modelos preditivos, é necessário definir a chamada função característica. No contexto do SHAP, essa função é dada por:

 ©2026 SBC – Soc. Bras. de Computação

$$v_x(S) = \mathbb{E}\left[f(X) \mid X_S = x_S\right], \tag{3}$$

onde $S \subseteq N$ representa um subconjunto de variáveis observadas, x_S são os valores dessas variáveis para a instância x, e a expectativa é calculada considerando a distribuição das variáveis não observadas $X_{\bar{S}}$ [Lundberg and Lee 2017, Covert et al. 2020]. Em termos simples, $v_x(S)$ responde à seguinte pergunta: *qual seria a predição esperada do modelo se conhecêssemos apenas as variáveis em S?*

Essa definição envolve uma escolha importante: como lidar com as variáveis ausentes ao calcular essa expectativa. Na prática, isso significa decidir como "preencher" os valores de $X_{\bar{S}}$. Uma abordagem comum utiliza a distribuição marginal dos dados, isto é, cada variável ausente é considerada de forma independente das demais. Embora simples, essa estratégia pode levar o modelo a avaliar combinações de características que não ocorrem na prática clínica. Por exemplo, ao fixar um valor elevado de pressão arterial sistólica e estimar a pressão diastólica de forma independente, o modelo pode gerar pares de valores que não são fisiologicamente plausíveis [Salih et al. 2023, Hooker et al. 2019, Hooshyar and Yang 2024].

Esse problema se torna ainda mais relevante em dados clínicos, nos quais muitas variáveis estão naturalmente correlacionadas, como pressão sistólica e diastólica, creatinina e ureia, ou índice de massa corporal e circunferência abdominal. Quando essas relações são ignoradas, o modelo passa a ser avaliado em regiões do espaço de dados que não foram observadas durante o treinamento, o que pode comprometer a interpretação das contribuições atribuídas [Salih et al. 2023, Hu et al. 2024a, Hooker et al. 2019].

Uma alternativa consiste em utilizar a distribuição condicional das variáveis ausentes, isto é, considerar $X_{\bar{S}}$ condicionado a $X_S = x_S$. Nesse caso, os valores ausentes são gerados de forma consistente com as variáveis observadas. Retomando o exemplo anterior, ao fixar uma pressão sistólica elevada, a pressão diastólica seria estimada dentro de uma faixa compatível com esse valor, preservando relações fisiológicas conhecidas. Essa abordagem, conhecida como SHAP condicional, tende a produzir cenários mais realistas [Olsen and Jullum 2025, Covert et al. 2020].

No entanto, essa escolha também altera o significado das explicações. Enquanto a abordagem marginal busca isolar o efeito de cada variável, a abordagem condicional passa a atribuir contribuições levando em conta o contexto das demais variáveis, o que pode redistribuir a importância entre preditores correlacionados. Além disso, a implementação é mais complexa e pode não preservar integralmente algumas propriedades teóricas do SHAP original [Lundberg and Lee 2017, Salih et al. 2023, Olsen and Jullum 2025, Hettikankanamage et al. 2025].

Assim, a escolha entre utilizar a distribuição marginal ou condicional não é apenas técnica: ela define o tipo de cenário hipotético considerado na explicação e influencia diretamente o significado dos valores SHAP obtidos.

2.4.3. Cálculo dos Valores SHAP

A contribuição de cada variável no SHAP é dada pelo valor de Shapley aplicado à função característica v_x. Para uma variável i e uma instância x, esse valor é definido por:

 ©2026 SBC – Soc. Bras. de Computação

$$\phi_i(x) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!\,(p-|S|-1)!}{p!} \left[v_x(S \cup \{i\}) - v_x(S) \right], \tag{4}$$

onde $\phi_i(x)$ representa a contribuição da variável i para a predição, calculada como a média de seu efeito marginal ao ser adicionada a diferentes combinações das demais variáveis [Lundberg and Lee 2017, Nohara et al. 2021, Binzagr 2024, Bifarin 2022]. O valor base da predição corresponde a $\phi_0 = v_x(\emptyset) = \mathbb{E}[f(X)]$.

Em termos intuitivos, esse cálculo considera todas as formas possíveis de incluir a variável i no modelo e mede quanto ela altera a predição em cada caso. O valor SHAP final é a média desses efeitos, ponderada de forma a garantir propriedades de equidade e consistência.

Quando há correlação entre variáveis (colinearidade), a interpretação se torna mais delicada. Nesse caso, o método distribui o efeito conjunto entre as variáveis correlacionadas, com base na média de suas contribuições marginais. Isso significa que, se duas variáveis carregam informações semelhantes, a contribuição atribuída a cada uma depende de como o modelo aprendeu essa relação durante o treinamento. Como resultado, a decomposição continua sendo única do ponto de vista matemático, mas sua interpretação clínica pode ser ambígua: parte do efeito atribuído a uma variável pode, na prática, refletir a influência de outra variável correlacionada [Nohara et al. 2021, Salih et al. 2023, Hu et al. 2024a].

Como o cálculo direto dessa expressão é inviável em modelos com muitas variáveis, métodos como TreeSHAP e KernelSHAP são utilizados para estimar os valores SHAP de forma eficiente em aplicações práticas [Nohara et al. 2021, Ghosh and Khandoker 2024, Luo et al. 2024, Lundberg and Lee 2017, Lundberg et al. 2020].

2.4.4. Exemplo Clínico: Estratificação de Risco Cardiovascular

Para ilustrar o uso dos valores SHAP, considere um modelo de *gradient boosting* treinado para prever a presença de doença cardiovascular a partir de variáveis demográficas, clínicas e comportamentais. Entre os preditores estão idade, sexo, pressão arterial sistólica e diastólica, índice de massa corporal (IMC), níveis de colesterol e glicemia, além de indicadores de tabagismo, consumo de álcool e atividade física. O valor base do modelo, $\phi_0 \approx -0{,}02$, corresponde à predição média no conjunto de treinamento, expressa em *log-odds*.

Considere agora um paciente do sexo masculino, com 52,4 anos, pressão arterial sistólica de 140 mmHg, diastólica de 90 mmHg, IMC de 30,1 kg/m^2, colesterol e glicemia normais (`cholesterol` = `gluc` = 1), não tabagista, não etilista e sedentário. Para esse paciente, o SHAP decompõe a predição da seguinte forma:

$$\begin{aligned} f(x) = -0{,}02 + \underbrace{1{,}32}_{\text{PAS=140}} + \underbrace{0{,}18}_{\text{PAD=90}} + \underbrace{0{,}14}_{\text{Sedentarismo}} \\ + \underbrace{(-0{,}06)}_{\text{Idade}} + \underbrace{(-0{,}04)}_{\text{Colesterol}} + \underbrace{0{,}01}_{\text{IMC}} + \underbrace{(-0{,}01)}_{\text{Demais}} \approx 1{,}54 \end{aligned}$$

 ©2026 SBC – Soc. Bras. de Computação

Essa decomposição mostra como cada variável contribui para o valor final da predição. A pressão arterial sistólica elevada é o principal fator, respondendo pela maior parte do aumento do risco. A pressão diastólica e o sedentarismo também contribuem positivamente, mas com menor intensidade. Por outro lado, o fato de o paciente ter colesterol normal e idade intermediária reduz parcialmente o risco previsto [Ponce-Bobadilla et al. 2024, Stenwig et al. 2022].

No entanto, essa interpretação deve ser feita com cautela. A pressão sistólica e a diastólica são variáveis fisiologicamente relacionadas, e o SHAP distribui o efeito conjunto entre elas com base em como o modelo aprendeu essa relação. Assim, a maior contribuição atribuída à pressão sistólica não significa necessariamente que seu efeito seja independente ou mais "importante" do ponto de vista clínico. Mudanças na distribuição dos dados ou no próprio modelo podem alterar essa divisão das contribuições sem modificar a predição final [Salih et al. 2023].

Portanto, os valores SHAP devem ser entendidos como uma descrição de como o modelo utiliza as variáveis para gerar a predição, e não como uma representação direta dos mecanismos causais envolvidos no risco cardiovascular.

2.4.5. Considerações sobre Escala de Explicação

Em modelos de classificação binária, os valores SHAP podem ser expressos em duas escalas: probabilidade ou *log-odds* [Lundberg and Lee 2017, Ponce-Bobadilla et al. 2024]. Embora essa escolha não altere a ordem de importância das variáveis, ela influencia a forma como os resultados são interpretados e comunicados.

Na **escala de probabilidade**, as contribuições indicam diretamente quanto cada variável aumenta ou reduz a probabilidade do desfecho, $p(x) = \mathbb{P}(Y = 1 \mid x)$. Essa forma é mais intuitiva e, por isso, costuma ser mais adequada para comunicação clínica. No entanto, como a relação entre *log-odds* e probabilidade é não linear, a decomposição pode não ser estritamente aditiva. Isso significa que o impacto de uma variável pode parecer maior ou menor dependendo do nível de risco basal do paciente, mesmo que seu efeito real no modelo seja constante [Ponce-Bobadilla et al. 2024].

Na **escala de *log-odds***, a decomposição é exatamente aditiva:

$$\text{logit}(p(x)) = \phi_0 + \sum_{j=1}^{p} \phi_j^{(\text{logit})}(x) \tag{5}$$

Nessa escala, cada contribuição pode ser interpretada de forma multiplicativa em termos de *odds ratio* (e^{ϕ_j}), o que é particularmente útil em análises epidemiológicas e na comparação entre fatores de risco [Ponce-Bobadilla et al. 2024, Lundberg and Lee 2017, Nohara et al. 2021].

Na prática, uma abordagem comum é utilizar a escala de *log-odds* para análise técnica, especialmente ao estudar interações entre variáveis, e a escala de probabilidade para comunicação com profissionais de saúde. Em qualquer caso, é importante explicitar qual escala está sendo utilizada, pois isso afeta a interpretação dos resultados. As propriedades teóricas do SHAP permanecem válidas em ambas as escalas, desde que a função explicada

 ©2026 SBC – Soc. Bras. de Computação

seja claramente definida [Lundberg and Lee 2017, Ponce-Bobadilla et al. 2024].

A seção seguinte discute como essas contribuições podem ser organizadas em análises locais e globais voltadas ao suporte à decisão clínica, retomando essas limitações no contexto de aplicações práticas.

2.5. Interpretação Local e Global com SHAP

O SHAP permite interpretar modelos preditivos em dois níveis complementares: o nível local e o nível global. No nível local, o objetivo é explicar uma predição específica, ou seja, entender como cada variável contribuiu para o resultado de um indivíduo. Já no nível global, busca-se compreender o comportamento geral do modelo, identificando padrões médios, variáveis mais influentes e possíveis relações entre os preditores [Bodria et al. 2021, Hakkoum et al. 2024].

No contexto da saúde, essa distinção é especialmente importante. Explicações locais são úteis para apoiar decisões clínicas individualizadas, permitindo que o profissional entenda por que um paciente foi classificado como de alto ou baixo risco. Por outro lado, análises globais ajudam a avaliar o modelo como um todo, identificar fatores de risco em nível populacional e atender a exigências regulatórias relacionadas à transparência e validação [Salih et al. 2023, Ponce-Bobadilla et al. 2024, Bachmann 2025, Joachim et al. 2026].

Uma característica importante do SHAP é que esses dois níveis estão diretamente conectados. As explicações globais são obtidas a partir da agregação das explicações locais, o que garante coerência entre a interpretação de casos individuais e o comportamento médio do modelo [Lundberg et al. 2020, Bodria et al. 2021]. Isso diferencia o SHAP de métodos como o LIME, que se concentram apenas em explicações locais e não oferecem, de forma direta, uma visão global consistente [Alabi et al. 2023, Vimbi et al. 2024].

2.5.1. Interpretação Local: Explicando Decisões Individuais

A interpretação local no SHAP tem como objetivo explicar a predição feita para um paciente específico. Isso é feito decompondo o valor previsto $f(x)$ em contribuições individuais de cada variável, de acordo com a expressão:

$$f(x) = \phi_0 + \sum_{i=1}^{p} \phi_i(x) \tag{6}$$

Nessa decomposição, ϕ_0 representa o valor base do modelo (a predição média), enquanto cada $\phi_i(x)$ indica quanto a variável i aumenta ou reduz a predição para aquele paciente.

Na prática, isso permite identificar, para cada paciente, quais fatores estão mais associados ao aumento ou à redução do risco e qual é a magnitude de cada efeito. Esse tipo de explicação é particularmente útil na clínica, pois conecta diretamente o resultado do modelo com características observáveis do paciente [Stenwig et al. 2022, Hu et al. 2024a, Luo et al. 2024].

Essa capacidade de rastrear a origem da predição é especialmente importante em sistemas de alerta precoce. Por exemplo, em uma unidade de terapia intensiva (UTI), um

 ©2026 SBC – Soc. Bras. de Computação

alerta de sepse pode ser explicado por contribuições elevadas de variáveis como lactato e pressão arterial baixa. Nesse caso, o modelo não apenas sinaliza o risco, mas também indica quais fatores estão mais associados a esse alerta, oferecendo ao profissional de saúde uma justificativa clara e imediatamente interpretável [Hu et al. 2022].

Por serem calculados em relação a um mesmo valor base e obedecerem aos axiomas de Shapley, os valores SHAP são diretamente comparáveis entre indivíduos [Lundberg and Lee 2017, Binzagr 2024]. Em outras palavras, o efeito de uma variável tem o mesmo significado quantitativo em todos os indivíduos, o que permite analisar padrões de forma consistente.

Essa propriedade possibilita diferentes aplicações relevantes na prática clínica. Uma delas é a identificação de fenótipos clínicos: ao comparar os padrões de contribuição entre pacientes, é possível identificar subgrupos com perfis de risco distintos, como aqueles em que o risco é mais influenciado por fatores renais ou por fatores cardiovasculares [Luo et al. 2024, Bifarin 2022, Lundberg et al. 2020].

Outra aplicação é o monitoramento longitudinal. Ao acompanhar um mesmo paciente ao longo do tempo, pode-se observar como as contribuições das variáveis mudam, indicando alterações no perfil de risco [Ponce-Bobadilla et al. 2024, Patharkar et al. 2024]. Isso pode ajudar, por exemplo, a avaliar a evolução clínica ou a resposta a intervenções.

Além disso, a comparabilidade dos valores SHAP permite identificar casos atípicos. Quando um paciente apresenta um padrão de explicação muito diferente dos demais, isso pode indicar tanto um perfil clínico incomum quanto possíveis limitações do modelo, como falta de representatividade desse tipo de caso nos dados de treinamento [Salih et al. 2023, Auzine et al. 2024, Liu et al. 2022].

2.5.2. Interpretação Global por Agregação Estatística

As interpretações globais no SHAP são obtidas a partir da agregação dos valores individuais calculados para cada paciente em uma amostra representativa [Lundberg and Lee 2017, Ponce-Bobadilla et al. 2024]. Em vez de analisar um caso isolado, o objetivo aqui é entender como o modelo se comporta, em média, na população.

Duas medidas são particularmente importantes nesse contexto. A primeira é a **importância global média**, que indica o quanto, em média, cada variável contribui para a predição, independentemente da direção do efeito. Ela é definida como:

$$I_i = \mathbb{E}\left[|\phi_i(X)|\right] \approx \frac{1}{n}\sum_{j=1}^{n} |\phi_i(x_j)| \tag{7}$$

Essa métrica permite identificar quais variáveis têm maior impacto no modelo como um todo [Wang et al. 2024, Bifarin 2022].

A segunda medida avalia a **direcionalidade do efeito**. Em vez de usar a média aritmética das contribuições brutas — que, por definição matemática do SHAP, se aproxima de zero ao longo do conjunto de dados de referência [Nohara et al. 2021] —, a direção é frequentemente quantificada pela correlação (por exemplo, coeficiente de Spearman) entre os valores da variável e os seus respectivos valores SHAP [Joachim et al. 2026]. Essa

 ©2026 SBC – Soc. Bras. de Computação

medida indica se o aumento de uma variável tende a aumentar ou reduzir o risco.

A análise conjunta da importância média global e da direcionalidade é útil para identificar padrões mais complexos. Quando o efeito de uma variável é consistente na população (por exemplo, sempre aumentando o risco), a correlação tende a ser forte. No entanto, quando há heterogeneidade, isto é, quando a variável aumenta o risco em alguns pacientes e reduz em outros, a importância global média pode permanecer alta, enquanto a medida de direcionalidade se aproxima de zero devido ao cancelamento entre efeitos positivos e negativos [Ponce-Bobadilla et al. 2024, Joachim et al. 2026]. Esse comportamento indica que o efeito da variável não é uniforme na população.

Além disso, essas medidas permitem comparar os resultados do modelo com o conhecimento epidemiológico existente e explorar padrões mais complexos, como relações não lineares e interações entre variáveis, especialmente quando combinadas com técnicas de visualização apropriadas [Ghasemi et al. 2024, Bifarin 2022]. Para além das métricas agregadas, a análise da **distribuição completa dos valores SHAP** fornece uma visão mais rica do comportamento do modelo. Uma alta dispersão nas contribuições de uma variável pode indicar diferentes fenômenos: relações não lineares (como a presença de limiares clínicos), interações com outras variáveis (por exemplo, quando o efeito de um fator depende da presença de outro) ou heterogeneidade real e presença de efeitos raros de alta magnitude entre pacientes [Ponce-Bobadilla et al. 2024, Cha et al. 2021, Lundberg et al. 2020].

Esses padrões de heterogeneidade e não linearidade podem ser explorados por meio de visualizações, que serão apresentadas mais adiante. Os *beeswarm plots* permitem observar a distribuição das contribuições de cada variável em toda a amostra, enquanto os *dependence plots* mostram como essas contribuições variam em função do valor da variável e de possíveis interações [Ponce-Bobadilla et al. 2024, Lundberg et al. 2020]. Essas visualizações tornam visíveis relações não lineares e efeitos condicionais que não são capturados por métricas baseadas em médias.

Por fim, a análise global pode ser aprofundada por meio da **estratificação por subgrupos clínicos relevantes**. Comparar, por exemplo, pacientes cirúrgicos e clínicos, diferentes faixas etárias ou a presença de comorbidades permite avaliar se o modelo se comporta de forma consistente entre populações distintas. Essa etapa é fundamental para detectar vieses, limitações de generalização e possíveis diferenças na relevância das variáveis entre grupos [Hakkoum et al. 2024, Lundberg et al. 2020].

2.5.3. Implicações para Validação e Auditoria Regulatória

O uso do SHAP tem se tornado cada vez mais relevante em processos de validação clínica e auditoria regulatória. Embora ainda não exista uma padronização formal consolidada, diferentes aplicações já demonstram como essas análises podem apoiar a avaliação e o monitoramento de modelos preditivos.

Uma primeira aplicação importante é a **validação de plausibilidade clínica**. Por meio de análises globais, é possível verificar se o modelo utiliza as variáveis de forma consistente com o conhecimento médico estabelecido. Em estudos multicêntricos, por exemplo, o SHAP tem sido utilizado para confirmar que os principais preditores identificados pelo modelo correspondem a fatores de risco reconhecidos na literatura. Quando surgem incon-

 ©2026 SBC – Soc. Bras. de Computação

sistências relevantes, isso pode indicar problemas nos dados, presença de confundimento ou falhas no próprio modelo [Prendin et al. 2023, Zeng et al. 2025, Kheder et al. 2025].

Outra aplicação central é a **detecção de vieses algorítmicos**. Ao analisar as contribuições das variáveis em diferentes subgrupos — como sexo, raça ou condição socioeconômica — é possível identificar padrões discriminatórios. Esse tipo de análise também ajuda a distinguir entre efeitos biológicos reais e variáveis que funcionam como proxies de desigualdade no acesso ao sistema de saúde [Momenzadeh et al. 2022, Hettikankanamage et al. 2025, Obermeyer et al. 2019, Chen et al. 2021].

O SHAP também tem sido utilizado na **documentação regulatória**, especialmente em sistemas classificados como SaMD. Nesses casos, análises locais e globais são incluídas na documentação técnica para permitir a verificação externa do comportamento do modelo e facilitar a reprodutibilidade dos resultados [Singh et al. 2025, Kiseleva et al. 2022, U.S. Food and Drug Administration 2021].

Além disso, o SHAP pode apoiar o **monitoramento pós-implantação**. A análise periódica das contribuições permite identificar possíveis mudanças no comportamento do modelo ao longo do tempo, fenômeno conhecido como *drift*, que pode ocorrer devido a alterações na população atendida ou nas práticas clínicas. Esse tipo de monitoramento contribui para auditorias contínuas de desempenho e equidade (*fairness*), embora ainda seja mais comum como prática conceitual do que como exigência regulatória formal [Lundberg et al. 2020, Stogiannos et al. 2023, Singh et al. 2025].

Em conjunto, essas aplicações mostram que o SHAP não se limita à explicação de casos individuais. Ele também pode ser utilizado como ferramenta de validação, governança e monitoramento de modelos ao longo de todo o seu ciclo de vida.

2.6. Implementação Computacional do SHAP em R e Python

Esta seção apresenta, de forma paralela, a implementação do SHAP em R e em Python para um mesmo problema clínico: a classificação binária de risco cardiovascular. Em ambas as linguagens, utiliza-se o mesmo conjunto de variáveis e um modelo XGBoost equivalente. O objetivo é permitir que o leitor acompanhe as etapas nas duas abordagens, compare suas estruturas e compreenda boas práticas de reprodutibilidade.

Os experimentos são realizados com uma base de dados pública, anonimizada e amplamente utilizada em estudos metodológicos de aprendizado de máquina em saúde, composta por variáveis demográficas, clínicas e comportamentais associadas ao risco cardiovascular[1]. O fluxo de trabalho adotado é o mesmo em R e Python e inclui cinco etapas principais: (1) preparação dos dados e particionamento estratificado em treino e teste; (2) ajuste de um modelo XGBoost; (3) avaliação do desempenho por meio da AUC-ROC e da calibração; (4) cálculo de valores SHAP em nível local e global; e (5) geração de visualizações voltadas à interpretação clínica.

Os códigos apresentados a seguir foram organizados para serem autocontidos e executáveis no Google Colab[2], sem necessidade de configuração adicional do ambiente.

[1]Disponível em: `https://www.kaggle.com/datasets/sulianova/cardiovascular-disease-dataset`

[2]`https://colab.research.google.com`

 ©2026 SBC – Soc. Bras. de Computação

No caso do Python, basta utilizar o kernel padrão da plataforma. Para a implementação em R, é necessário selecionar a opção *Ambiente de execução* > *Alterar o tipo de ambiente de execução* > *R* antes de iniciar a sessão. Em ambos os casos, o arquivo `cardio_train.csv` deve ser carregado previamente, seja pelo painel *Files* do Colab, seja por código de upload executado no próprio notebook. A instalação das dependências é feita na primeira célula de cada implementação e, uma vez concluída, pode ser dispensada nas execuções seguintes dentro da mesma sessão.

Cabe destacar que o objetivo desta seção não é apresentar um tutorial completo de construção de modelos preditivos, mas sim ilustrar, de forma prática, como aplicar o SHAP em um conjunto de dados clínicos. O foco está na interpretação do modelo já ajustado, mostrando como extrair e analisar as contribuições das variáveis em nível local e global. Dessa forma, as etapas de modelagem são apresentadas de maneira objetiva, apenas na medida necessária para viabilizar o uso e a compreensão das explicações geradas pelo SHAP.

2.6.1. Estrutura dos Dados e Pré-processamento

Considerando o conjunto de dados descrito anteriormente, as variáveis preditoras abrangem diferentes dimensões clínicas relevantes para o risco cardiovascular. Entre as variáveis **demográficas**, incluem-se idade (convertida de dias para anos) e sexo. As variáveis **antropométricas** compreendem o IMC, calculado a partir de altura e peso. Os **sinais vitais** incluem as pressões arteriais sistólica e diastólica, medidas em mmHg. O conjunto também contempla **marcadores metabólicos categóricos**, como níveis de colesterol e glicemia codificados em categorias ordinais, além de **fatores comportamentais**, como tabagismo, consumo de álcool e prática de atividade física, representados como variáveis binárias.

A variável resposta é binária, $Y \in \{0, 1\}$, em que $Y = 1$ indica a presença de doença cardiovascular.

Antes do treinamento do modelo, algumas etapas de pré-processamento são particularmente importantes em dados clínicos. Em primeiro lugar, **valores ausentes** são frequentes em bases de prontuário eletrônico. Modelos como XGBoost e LightGBM lidam diretamente com dados faltantes, o que reduz a necessidade de intervenção. Ainda assim, pode-se optar por métodos de imputação desde que essa escolha seja explicitamente documentada.

Em relação às **variáveis categóricas**, é necessário convertê-las para formato numérico. Uma prática comum é utilizar *label encoding* para variáveis ordinais (como colesterol e glicemia) e *one-hot encoding* para variáveis nominais, o que é compatível com o uso posterior do TreeSHAP.

Quanto à **escala das variáveis**, modelos baseados em árvores são invariantes a transformações monotônicas, o que dispensa normalização. No entanto, manter as unidades clínicas originais facilita a interpretação dos valores SHAP.

Por fim, a presença de **outliers** deve ser avaliada com cuidado. Valores clinicamente implausíveis — como níveis extremos de pressão arterial — podem distorcer tanto o treinamento do modelo quanto as explicações geradas, devendo ser investigados e tratados

 ©2026 SBC – Soc. Bras. de Computação

previamente.

Cabe ressaltar que essas decisões de pré-processamento não afetam apenas o desempenho do modelo, mas também influenciam diretamente a interpretação dos valores SHAP, uma vez que alteram a forma como as variáveis são representadas e utilizadas na predição.

2.6.2. Implementação em Python com `shap` e `xgboost`

Em Python, a implementação do SHAP é facilitada pela biblioteca `shap`, que oferece suporte nativo ao algoritmo TreeSHAP para modelos baseados em árvores, como o XGBoost. Isso permite calcular explicações locais e globais de forma eficiente, sem necessidade de implementar manualmente os valores de Shapley [Lundberg and Lee 2017]. Nesta seção, apresentamos um exemplo completo, desde a preparação dos dados até a geração de gráficos interpretativos.

2.6.2.1. Preparação dos Dados, Limpeza e Treinamento do Modelo

Antes de calcular os valores SHAP, é necessário preparar a base, realizar uma limpeza mínima dos dados e treinar o modelo preditivo. No Google Colab, a instalação das bibliotecas pode ser feita com o comando abaixo:

Instalação (executar uma vez no Colab):

```
!pip install shap xgboost scikit-learn pandas numpy
matplotlib -quiet
```

O código a seguir carrega a base de dados, remove valores clinicamente implausíveis, cria variáveis derivadas e ajusta um modelo XGBoost para classificação binária.

```
import numpy as np
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.metrics import roc_auc_score, brier_score_loss

from xgboost import XGBClassifier
import shap

df = pd.read_csv("cardio_train.csv")

# Limpeza: pressão arterial
df = df[
(df["ap_hi"].between(70, 250)) &
(df["ap_lo"].between(40, 150)) &
(df["ap_hi"] > df["ap_lo"])
].copy()
```

 ©2026 SBC – Soc. Bras. de Computação

```
# Altura e peso
df = df[
(df["height"].between(120, 220)) &
(df["weight"].between(40, 200))
].copy()

df["age_years"] = df["age"] / 365.25
df["imc"] = df["weight"] / ((df["height"] / 100) ** 2)
df = df[df["imc"].between(15, 60)].copy()
print("N após limpeza:", df.shape[0])

features = [
"age_years", "gender", "imc",
"ap_hi", "ap_lo",
"cholesterol", "gluc",
"smoke", "alco", "active"
]
target = "cardio"
X = df[features]
y = df[target].astype(int)

# Particionamento estratificado (25% teste)
X_train, X_test, y_train, y_test = train_test_split(
X, y, test_size=0.25, random_state=42, stratify=y
)

model = XGBClassifier(
n_estimators=300,
max_depth=4,
learning_rate=0.05,
subsample=0.8,
colsample_bytree=0.8,
reg_lambda=1.0,
reg_alpha=0.1,
random_state=42,
eval_metric="logloss",
use_label_encoder=False,
n_jobs=-1
)

model.fit(X_train, y_train, eval_set=[(X_test, y_test)],
 verbose=False)

# Avaliação
y_prob = model.predict_proba(X_test)[:, 1]
```

 ©2026 SBC – Soc. Bras. de Computação

```
auc = roc_auc_score(y_test, y_prob)
brier = brier_score_loss(y_test, y_prob)

print(f"AUC-ROC: {auc:.3f}")
print(f"Brier Score: {brier:.4f}")
```

Esse bloco executa cinco tarefas principais. Primeiro, importa as bibliotecas necessárias para manipulação de dados, treinamento do modelo, avaliação e cálculo de valores SHAP. Em seguida, lê o arquivo `cardio_train.csv`. Depois, aplica uma limpeza básica para excluir valores incompatíveis com a prática clínica, como pressões arteriais inconsistentes, alturas ou pesos extremos, e valores de IMC fora de uma faixa plausível.

Na sequência, duas variáveis derivadas são criadas: idade em anos (`age_years`) e índice de massa corporal (`imc`). Essas transformações tornam os dados mais adequados tanto para o treinamento quanto para a interpretação clínica posterior. Depois disso, define-se o conjunto de variáveis preditoras e a variável resposta.

O particionamento entre treino e teste é feito de forma estratificada, preservando a proporção de casos com e sem doença cardiovascular em ambos os subconjuntos. Em seguida, ajusta-se um modelo `XGBClassifier` com hiperparâmetros moderados, suficientes para fins ilustrativos. Por fim, o modelo é avaliado por duas métricas: a AUC-ROC, que mede sua capacidade discriminativa, e o *Brier Score*, que avalia a calibração das probabilidades previstas.

2.6.2.2. Cálculo dos Valores SHAP com `TreeExplainer`

Depois do treinamento, o próximo passo é calcular os valores SHAP. Como o modelo utilizado é baseado em árvores, a forma mais eficiente de fazer isso é por meio do `TreeExplainer`, que implementa o TreeSHAP.

```
# TreeSHAP (log-odds)
explainer = shap.TreeExplainer(model)
shap_values = explainer.shap_values(X_test)
base_value = explainer.expected_value
print(f"Valor base (phi_0): {base_value:.4f}")

# Verificação da decomposição aditiva
i = 0
pred_logodds = base_value + shap_values[i].sum()
pred_prob_check = 1 / (1 + np.exp(-pred_logodds))

print(f"Predição reconstruída: {pred_prob_check:.4f}")
print(f"Predição do modelo: {y_prob[i]:.4f}")
```

Nesse trecho, o objeto `explainer` aprende como decompor as predições do modelo em contribuições individuais das variáveis. O resultado, armazenado em

 ©2026 SBC – Soc. Bras. de Computação

`shap_values`, contém um valor SHAP para cada variável e para cada paciente do conjunto de teste. Já `base_value` corresponde ao valor base da predição, isto é, à saída média do modelo antes de considerar as características específicas de cada indivíduo.

O código também mostra uma verificação importante: a soma dos valores SHAP de um paciente, adicionada ao valor base, deve reconstruir a predição do modelo na escala de *log-odds*. Em seguida, essa quantidade é transformada em probabilidade pela função logística. Essa etapa ajuda o leitor a visualizar, na prática, a propriedade de aditividade local do SHAP.

2.6.2.3. Visualizações para Interpretação

Uma das principais vantagens do SHAP é a possibilidade de gerar visualizações que ajudam a interpretar o modelo tanto em nível global quanto local. O código abaixo apresenta alguns dos gráficos mais utilizados.

```
import matplotlib.pyplot as plt

# Summary plots (beeswarm e bar)
shap.summary_plot(shap_values, X_test, plot_type="dot")
shap.summary_plot(shap_values, X_test, plot_type="bar")

# Dependence plot: ap_hi x colesterol
shap.dependence_plot(
"ap_hi",
shap_values,
X_test,
interaction_index="cholesterol"
)

# Waterfall plot (explicação local)
idx_paciente = 0
# Garantir escalar para base_value
if isinstance(base_value, (list, tuple, np.ndarray)):
bv = float(np.array(base_value).reshape(-1)[0])
else:
bv = float(base_value)

exp = shap.Explanation(
values=np.array(shap_values)[idx_paciente],
base_values=bv,
data=X_test.iloc[idx_paciente].to_numpy(),
feature_names=features
)

shap.plots.waterfall(exp)
```

 ©2026 SBC – Soc. Bras. de Computação

```
# Force plot
shap.force_plot(bv, shap_values[idx_paciente],
X_test.iloc[idx_paciente])
```

Os primeiros comandos geram gráficos de resumo, que permitem uma visão global do comportamento do modelo. O *beeswarm plot* mostra, para cada variável, a distribuição dos valores SHAP em toda a amostra, permitindo observar simultaneamente a importância, a direção do efeito (aumento ou redução do risco) e a variabilidade entre pacientes. Já o gráfico de barras apresenta a importância média global das variáveis, calculada a partir da magnitude média de suas contribuições.

Em seguida, o *dependence plot* explora como a contribuição de uma variável específica varia ao longo de seus valores. No exemplo apresentado, analisa-se a pressão arterial sistólica (`ap_hi`) e sua possível interação com os níveis de colesterol. Esse tipo de visualização é útil para identificar relações não lineares e efeitos condicionais entre variáveis.

Por fim, os gráficos *waterfall* e *force plot* apresentam explicações locais, isto é, detalham a predição de um paciente específico. Nesses gráficos, a predição individual é construída a partir do valor base e das contribuições positivas e negativas de cada variável, permitindo compreender de forma direta quais fatores levaram o modelo a classificar aquele paciente como de maior ou menor risco.

Cabe destacar que cada um desses gráficos será discutido em maior detalhe na seção seguinte, com foco em sua interpretação clínica e estatística. Para fins de padronização visual, as figuras apresentadas no texto serão geradas em R; no entanto, todas essas visualizações podem ser produzidas de forma equivalente em Python, como ilustrado neste exemplo.

De modo geral, esse fluxo em Python demonstra como treinar um modelo compatível com o TreeSHAP, calcular explicações locais e globais e gerar visualizações interpretáveis com poucas linhas de código. O objetivo não é otimizar exaustivamente o modelo, mas mostrar, de forma prática, como o SHAP pode ser incorporado à análise de um problema clínico real.

2.6.3. Implementação em R com `xgboost` e `shapviz`

No ecossistema R, a combinação entre `xgboost` para modelagem e `shapviz` para interpretação permite reproduzir, de forma equivalente, o fluxo apresentado em Python. A principal diferença está na sintaxe e nas convenções da linguagem, mas as etapas centrais permanecem as mesmas: preparação dos dados, ajuste do modelo, cálculo dos valores SHAP e geração de gráficos interpretativos.

2.6.3.1. Preparação dos Dados, Limpeza e Treinamento do Modelo

No Google Colab com kernel R, as bibliotecas necessárias podem ser instaladas com o comando abaixo:

 ©2026 SBC – Soc. Bras. de Computação

Instalação (executar uma vez no Colab com kernel R):

```
install.packages(c("data.table", "xgboost", "shapviz",
"pROC"))
```

O código a seguir realiza a leitura da base, aplica critérios básicos de limpeza, constrói variáveis derivadas, separa os dados em treino e teste e ajusta um modelo XGBoost para classificação binária.

```
library(data.table)
library(xgboost)
library(shapviz)
library(pROC)

df <- fread("cardio_train.csv")

# Limpeza: pressão arterial
df <- df[
ap_hi >= 70 & ap_hi <= 250 &
ap_lo >= 40 & ap_lo <= 150 &
ap_hi > ap_lo
]

# Altura e peso
df <- df[
height >= 120 & height <= 220 &
weight >= 40  & weight <= 200
]

df[, age_years := age / 365.25]
df[, imc := weight / (height / 100)^2]
df <- df[imc >= 15 & imc <= 60]
cat("N após limpeza:", nrow(df), "\n")

features <- c(
"age_years", "gender", "imc",
"ap_hi", "ap_lo",
"cholesterol", "gluc",
"smoke", "alco", "active"
)
target <- "cardio"
stopifnot(all(c(features, target) %in% names(df)))
X <- as.matrix(df[, ..features])
y <- as.numeric(df[[target]])

# Particionamento estratificado (25% teste)
```

 ©2026 SBC – Soc. Bras. de Computação

```
set.seed(42)
idx1 <- which(y == 1)
idx0 <- which(y == 0)

idx_test <- c(
sample(idx1, size = floor(0.25 * length(idx1))),
sample(idx0, size = floor(0.25 * length(idx0)))
)

X_train <- X[-idx_test, , drop = FALSE]
y_train <- y[-idx_test]
X_test  <- X[idx_test,  , drop = FALSE]
y_test  <- y[idx_test]

dtrain <- xgb.DMatrix(data = X_train, label = y_train)
dtest  <- xgb.DMatrix(data = X_test,  label = y_test)

params <- list(
objective = "binary:logistic",
eval_metric = "logloss",
max_depth = 4,
eta = 0.05,
subsample = 0.8,
colsample_bytree = 0.8,
lambda = 1.0,
alpha = 0.1
)

model <- xgb.train(
params = params,
data = dtrain,
nrounds = 300,
watchlist = list(test = dtest),
verbose = 0
)

# Avaliação
pred_prob <- predict(model, dtest)
auc <- pROC::auc(y_test, pred_prob)
cat("AUC-ROC:", round(as.numeric(auc), 3), "\n")
```

Esse bloco segue a mesma lógica da implementação em Python. Primeiro, carrega as bibliotecas responsáveis pela leitura dos dados, treinamento do modelo, cálculo das explicações e avaliação de desempenho. Em seguida, lê o arquivo `cardio_train.csv` e aplica uma limpeza básica para remover valores incompatíveis com a prática clínica, como pressões arteriais inconsistentes, alturas e pesos fora de faixas plausíveis e valores

 ©2026 SBC – Soc. Bras. de Computação

extremos de IMC.

Depois disso, são construídas duas variáveis derivadas: idade em anos (`age_years`) e índice de massa corporal (`imc`). Essas transformações tornam os dados mais adequados à interpretação clínica. Em seguida, selecionam-se as variáveis preditoras e a variável resposta, e os dados são convertidos para o formato matricial exigido pelo `xgboost`.

O particionamento em treino e teste é feito de forma estratificada, preservando a proporção entre indivíduos com e sem doença cardiovascular. Após isso, os objetos `xgb.DMatrix` são criados para armazenar os dados em um formato otimizado para o treinamento. O modelo é então ajustado com parâmetros moderados, suficientes para fins ilustrativos, e avaliado por meio da AUC-ROC, que mede sua capacidade de discriminar entre pacientes com e sem o desfecho.

2.6.3.2. Cálculo de Valores SHAP

Depois de treinado o modelo, o próximo passo é obter os valores SHAP. No `xgboost` em R, isso pode ser feito diretamente com o argumento `predcontrib = TRUE`, que retorna a contribuição de cada variável para cada predição individual.

```
# SHAP (log-odds; última coluna = BIAS)
shap_contrib <- predict(model, dtest, predcontrib = TRUE)
shap_matrix <- shap_contrib[, 1:(ncol(shap_contrib) - 1),
 drop = FALSE]
base_value  <- shap_contrib[1, ncol(shap_contrib)]
colnames(shap_matrix) <- features
sv <- shapviz(model, X_pred = dtest, X = X_test)
```

Nesse trecho, `shap_contrib` armazena uma matriz em que cada linha corresponde a um paciente e cada coluna representa a contribuição de uma variável. A última coluna contém o valor base da predição, também chamado de *BIAS*, que corresponde ao ponto de partida a partir do qual as contribuições individuais são somadas. A matriz `shap_matrix` contém apenas os valores SHAP das variáveis, já com os nomes das colunas ajustados para facilitar a leitura.

Em seguida, a função `shapviz()` organiza essas informações em um objeto próprio para visualização. Esse objeto reúne o modelo, os dados utilizados nas predições e os valores SHAP, permitindo gerar gráficos de forma direta e consistente.

2.6.3.3. Visualizações com `shapviz`

Uma vez calculados os valores SHAP, o pacote `shapviz` oferece funções específicas para gerar visualizações globais e locais de forma simples.

```
# Bar plot
```

 ©2026 SBC – Soc. Bras. de Computação

```
sv_importance(sv, kind = "bar")

# Summary plot
sv_importance(sv, kind = "beeswarm")

# Dependence
sv_dependence(sv, v = "ap_hi", color_var = "cholesterol")

# Waterfall local
sv_waterfall(sv, row_id = 1)

# Force plot
sv_force(sv, row_id = 1)
```

Os dois primeiros comandos geram gráficos de resumo do modelo. O gráfico de barras apresenta a importância global média das variáveis, enquanto o *beeswarm plot* mostra a distribuição completa das contribuições em toda a amostra, permitindo observar importância, direção do efeito e heterogeneidade entre pacientes.

O *dependence plot* examina como a contribuição de uma variável específica varia de acordo com seus valores. No exemplo acima, analisa-se a pressão arterial sistólica (`ap_hi`) e sua possível interação com o colesterol. Esse tipo de gráfico é útil para investigar relações não lineares e efeitos condicionais.

Já os gráficos *waterfall* e *force plot* apresentam explicações locais, isto é, mostram como a predição de um paciente específico é construída a partir do valor base e das contribuições positivas e negativas de cada variável. Essas visualizações ajudam a entender, de forma direta, por que o modelo classificou aquele indivíduo como de maior ou menor risco.

Cabe destacar que cada um desses gráficos será discutido em maior detalhe na seção seguinte, com foco em sua interpretação clínica e estatística. Para fins de padronização visual, as figuras apresentadas no texto serão geradas em R. Ainda assim, como mostrado anteriormente, todas essas visualizações também podem ser produzidas de forma equivalente em Python.

De modo geral, esse fluxo em R demonstra como treinar um modelo compatível com o TreeSHAP, calcular explicações locais e globais e gerar visualizações interpretáveis de forma relativamente simples. Assim como na implementação em Python, o objetivo não é ensinar exaustivamente a modelagem preditiva, mas mostrar, de maneira prática, como o SHAP pode ser aplicado à interpretação de um banco de dados clínico.

2.6.4. Considerações de Desempenho para Grandes Bases Clínicas

Em aplicações com grandes volumes de dados, como bases de prontuários eletrônicos com milhões de registros, o cálculo de valores SHAP para todos os pacientes pode se tornar computacionalmente custoso. Para viabilizar o uso do método em larga escala, é comum combinar estratégias de amostragem, otimização algorítmica e técnicas de engenharia.

Uma abordagem prática é a **amostragem estratificada**. Para análises globais, não

 ©2026 SBC – Soc. Bras. de Computação

é necessário utilizar toda a base de dados: amostras representativas, tipicamente entre 1.000 e 10.000 instâncias, costumam ser suficientes para estimar importâncias médias com boa precisão. Estudos mostram que, quando a amostra preserva a estrutura da população, especialmente a distribuição do desfecho e de subgrupos clínicos relevantes, as estimativas permanecem estáveis [Bachmann 2025].

Outra estratégia importante é o **cálculo sob demanda**. Em sistemas de produção, nem sempre é necessário gerar explicações para todos os pacientes. Em vez disso, pode-se calcular valores SHAP apenas para casos de interesse, como pacientes que acionam alertas clínicos ou situações em que o usuário solicita uma explicação. Essa abordagem reduz significativamente o custo computacional e está alinhada com o uso prático do modelo no contexto clínico [Bachmann 2025, Hu et al. 2024a].

Do ponto de vista computacional, a **paralelização** é uma das principais formas de ganho de desempenho. O algoritmo TreeSHAP permite calcular explicações de forma independente para cada instância, o que facilita sua execução em múltiplos núcleos de CPU ou em GPUs. Em Python, isso pode ser explorado por meio de parâmetros como `n_jobs` ou bibliotecas como `joblib`; em R, pacotes como `parallel`, `future` e implementações otimizadas oferecem funcionalidades semelhantes para acelerar a computação [Greenwell and Boehmke 2020].

Além disso, o uso de **cache de resultados** é particularmente relevante em ambientes clínicos e regulatórios. Armazenar os valores SHAP já calculados permite reutilizar explicações em auditorias futuras, garantir rastreabilidade das decisões e evitar recálculos desnecessários. Essa prática é especialmente importante quando se deseja reconstruir, de forma retrospectiva, o comportamento do modelo em situações específicas para fins de conformidade [Kiseleva et al. 2022, U.S. Food and Drug Administration 2021].

Por fim, a literatura recente tem proposto métodos que reduzem ainda mais o custo computacional do SHAP. Entre eles estão algoritmos acelerados, como o Fast TreeSHAP, e abordagens aproximadas, como o FastSHAP, que utilizam estratégias de aprendizado para estimar contribuições de forma mais eficiente. Há também métodos baseados em decomposição ou aproximações mais refinadas, que buscam manter a qualidade das explicações com menor custo [Olsen and Jullum 2025, Gevaert and Saeys 2022].

Em conjunto, essas estratégias tornam o uso do SHAP viável mesmo em cenários clínicos de grande escala, permitindo equilibrar custo computacional e qualidade interpretativa.

2.7. Visualização e Análise de Resultados SHAP

A visualização dos valores SHAP tem como objetivo transformar resultados matemáticos em representações que possam ser interpretadas por diferentes públicos, como clínicos, pesquisadores e gestores. Enquanto métricas tradicionais, como AUC-ROC, indicam o desempenho geral do modelo, as visualizações SHAP ajudam a entender *como* e *por que* o modelo chega a uma determinada predição. Em outras palavras, elas tornam explícita a contribuição de cada variável, tanto no nível individual quanto no nível populacional.

Nesta seção, são apresentadas as principais estratégias de visualização associadas ao SHAP, utilizando como referência o conjunto de dados de risco cardiovascular discutido

 ©2026 SBC – Soc. Bras. de Computação

ao longo do capítulo. O foco está na interpretação dos gráficos e em sua aplicação prática no contexto clínico.

2.7.1. Princípios de Visualização para Aplicações Clínicas

O uso de visualizações em saúde exige cuidado especial, pois os resultados precisam ser compreendidos rapidamente e, ao mesmo tempo, manter rigor técnico. Quatro princípios orientam o uso de gráficos SHAP nesse contexto.

O primeiro é a **clareza e acessibilidade**. As informações devem ser apresentadas em unidades que façam sentido para o clínico (por exemplo, mmHg ou probabilidade de risco), com uso consistente de elementos visuais como cores e posições. Além disso, é importante permitir diferentes níveis de detalhamento, de modo que usuários com mais ou menos experiência possam interpretar os resultados de forma adequada [Patel et al. 2024].

O segundo princípio é a **fidelidade ao modelo**. As visualizações devem refletir de forma fiel o comportamento do modelo, sem simplificações que distorçam sua interpretação. Isso inclui, sempre que possível, indicar incertezas, limitações dos dados e possíveis fontes de erro, aspectos fundamentais para a confiança no uso clínico dessas ferramentas [Harrigan et al. 2022, Markus et al. 2021].

O terceiro princípio é a **relevância clínica**. Os padrões observados nos gráficos precisam ser interpretados à luz do conhecimento médico existente e conectados a tarefas reais, como diagnóstico, triagem ou acompanhamento de pacientes. Sem essa contextualização, a visualização pode ser tecnicamente correta, mas pouco útil na prática [Patel et al. 2024, Tonekaboni et al. 2019].

Por fim, destaca-se a **acionabilidade**. Sempre que possível, as visualizações devem apontar fatores que podem ser modificados, indicar limiares clínicos importantes e destacar situações de risco que demandem intervenção. O objetivo não é apenas explicar o modelo, mas apoiar decisões concretas no cuidado ao paciente [Harrigan et al. 2022, Tonekaboni et al. 2019].

2.7.2. Summary Plot: Visão Global do Comportamento do Modelo

O *summary plot*, também conhecido como *beeswarm plot*, é uma das visualizações mais completas para análise global com SHAP. Em um único gráfico, ele combina três informações importantes: a importância das variáveis no modelo, a distribuição de seus efeitos e a relação entre o valor da variável e sua contribuição para a predição [Lundberg et al. 2019, Lundberg and Lee 2017, Ponce-Bobadilla et al. 2024].

Nesse gráfico, as variáveis são organizadas no eixo vertical de acordo com sua importância global, medida pela média do valor absoluto das contribuições ($\mathbb{E}[|\phi_i|]$). Cada ponto representa um paciente da amostra. A posição horizontal do ponto indica o valor SHAP, ou seja, o quanto aquela variável aumentou ou reduziu a predição para aquele indivíduo. A cor do ponto representa o valor original da variável (por exemplo, valores mais altos ou mais baixos), o que permite visualizar, ao mesmo tempo, a direção do efeito, possíveis relações monotônicas, padrões não lineares e diferenças entre pacientes [Lundberg et al. 2020, Liu et al. 2022, Ponce-Bobadilla et al. 2024].

Na Figura 2.1, observa-se que a pressão arterial sistólica (`ap_hi`) e a idade

 ©2026 SBC – Soc. Bras. de Computação

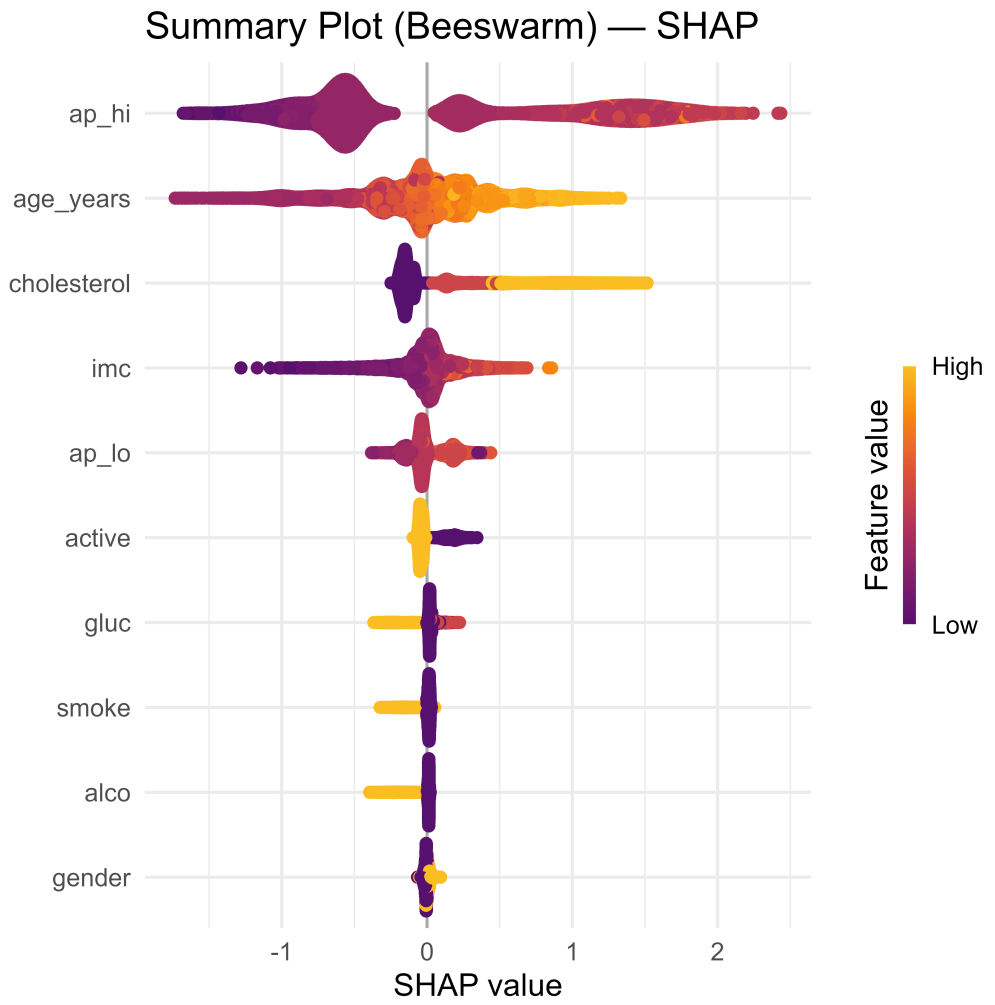


Figura 2.1. Summary plot (beeswarm) dos valores SHAP para o modelo XGBoost treinado no conjunto de dados cardiovascular. Variáveis ordenadas por importância global; coloração indica o valor original do preditor.

©2026 SBC – Soc. Bras. de Computação

(`age_years`) são as variáveis com maior influência no modelo. Valores mais altos dessas variáveis (indicados por cores mais claras) estão consistentemente associados a maior risco predito, ao passo que valores baixos contribuem negativamente para a predição. O colesterol, representado como variável ordinal, apresenta um padrão monotônico claro, com valores elevados associados a SHAP positivos. O IMC (`imc`) exibe comportamento assimétrico: valores baixos concentram-se em SHAP negativos, enquanto valores altos produzem contribuições ligeiramente positivas.

A pressão diastólica (`ap_lo`) também contribui para o risco, porém com menor magnitude e dispersão. A variável atividade física (`active`) apresenta SHAP predominantemente próximos de zero, com leve tendência protetora para indivíduos ativos. Por outro lado, variáveis como glicemia (`gluc`), tabagismo (`smoke`), consumo de álcool (`alco`) e sexo (`gender`) têm impacto relativamente menor quando consideradas em conjunto com as demais variáveis do modelo.

Uma forma mais simples de visualizar a importância das variáveis é o *bar plot*, que representa cada variável pela média do valor absoluto de suas contribuições ($\mathbb{E}[|\phi_i|]$) [Lundberg et al. 2020, Ponce-Bobadilla et al. 2024]. Esse gráfico oferece uma leitura direta do peso relativo de cada preditor no modelo.

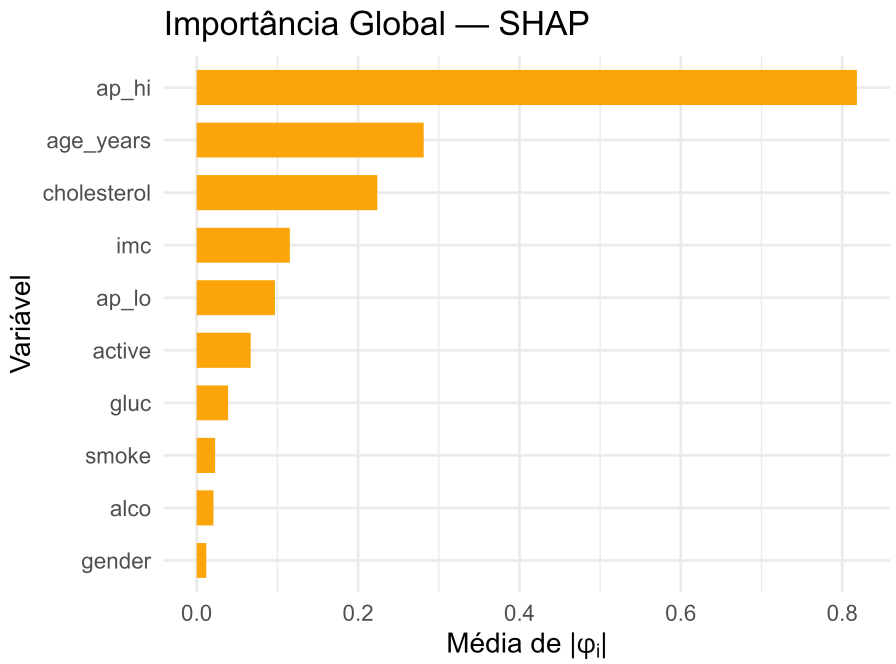


Figura 2.2. Bar plot de importância global baseado em $\mathbb{E}[|\phi_i|]$. As barras indicam a contribuição relativa de cada variável no modelo XGBoost treinado com os dados cardiovasculares.

Na Figura 2.2, observa-se novamente a predominância da pressão arterial sistólica, seguida pela idade e pelo colesterol. Variáveis comportamentais e sexo aparecem com menor contribuição relativa. Esse tipo de gráfico é útil para uma visão rápida e resumida do modelo, especialmente em contextos de comunicação. No entanto, ele não mostra a direção dos efeitos nem a variabilidade entre pacientes — informações que estão presentes no *beeswarm plot* [Lundberg et al. 2020, Ponce-Bobadilla et al. 2024].

 ©2026 SBC – Soc. Bras. de Computação

2.7.3. Dependence Plot: Relações Não Lineares e Limiares Clínicos

O *dependence plot* é uma visualização que permite analisar como o efeito de uma variável muda ao longo de seus valores. Nesse gráfico, o eixo x representa o valor observado da variável, enquanto o eixo y mostra sua contribuição SHAP para a predição. Cada ponto corresponde a um paciente da base, o que permite observar não apenas tendências médias, mas também a variabilidade entre indivíduos [Lundberg et al. 2020, Ponce-Bobadilla et al. 2024].

Diferentemente de métodos tradicionais, como gráficos de dependência parcial, o *dependence plot* preserva a heterogeneidade individual. Ou seja, ele não mostra apenas o comportamento médio do modelo, mas revela como o efeito da variável varia entre pacientes com perfis diferentes [Lundberg et al. 2020, Cha et al. 2021].

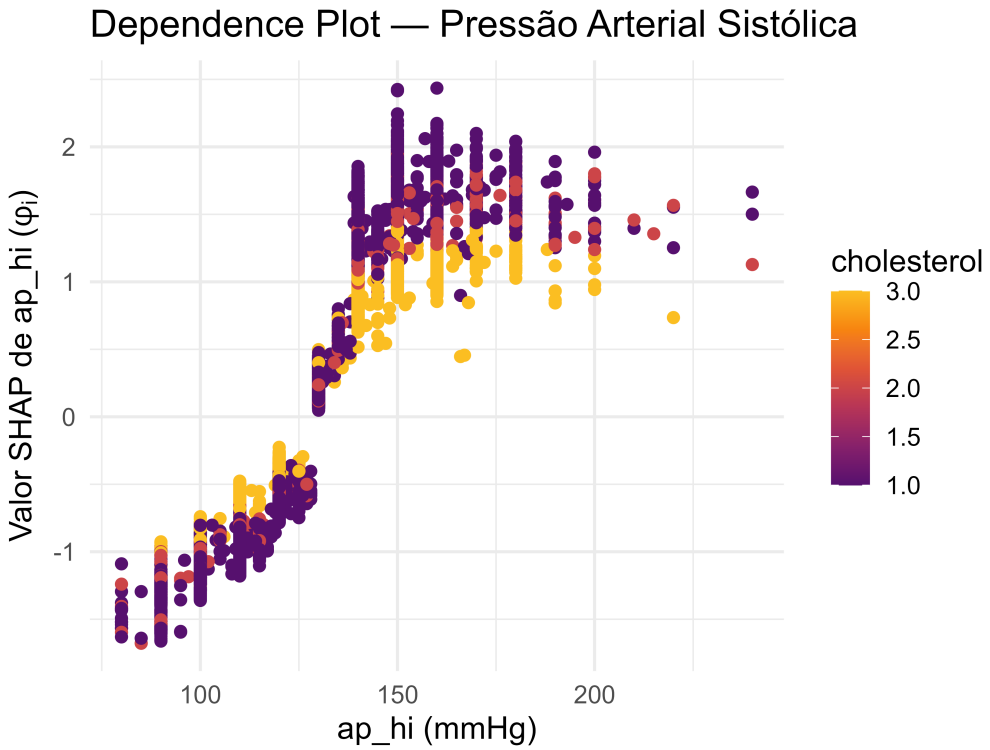


Figura 2.3. Dependence plot dos valores SHAP para `ap_hi`, com coloração pela variável ordinal de colesterol. Cada ponto representa uma observação individual.

Na Figura 2.3, observa-se uma relação não linear entre a pressão arterial sistólica (`ap_hi`) e o risco predito pelo modelo. Para valores abaixo de aproximadamente 130–135 mmHg, as contribuições SHAP são predominantemente negativas, indicando que pressões mais baixas reduzem o risco predito. A partir de cerca de 135–140 mmHg, há uma transição abrupta para contribuições positivas e crescentes. Esse ponto de inflexão é consistente com os limiares utilizados nas diretrizes clínicas para definição de hipertensão. Acima desse intervalo, o efeito permanece positivo e crescente, com dispersão vertical considerável dos pontos, refletindo a variação interindividual no impacto dessa variável.

A cor dos pontos, representando o nível de colesterol (`cholesterol`), permite identificar uma possível interação entre as duas variáveis. Na faixa de pressão baixa a

 ©2026 SBC – Soc. Bras. de Computação

moderada (aproximadamente 80–130 mmHg), pacientes com colesterol mais elevado tendem a apresentar contribuições SHAP ligeiramente maiores do que aqueles com colesterol baixo para um mesmo valor de pressão. De modo geral, a dispersão vertical dos pontos indica que o efeito da pressão arterial sistólica não é isolado, mas modulado pelo conjunto das demais características clínicas do paciente.

2.7.4. Visualizações Locais: Explicando Decisões Individuais

Para compreender como o modelo toma decisões em nível individual, duas visualizações são amplamente utilizadas: o *waterfall plot* e o *force plot*. Ambas mostram como a predição de um paciente é construída a partir das contribuições das variáveis, mas o fazem de maneiras complementares [Bifarin 2022, Lin et al. 2025].

Waterfall plot. O *waterfall plot* apresenta a decomposição da predição de forma sequencial. O gráfico começa no valor base ϕ_0, que corresponde à predição média do modelo sobre o conjunto de referência utilizado pelo SHAP, e adiciona (ou subtrai) as contribuições SHAP de cada variável até alcançar a predição final [Ponce-Bobadilla et al. 2024, Lin et al. 2025, Bifarin 2022]. Contribuições positivas, que aumentam o risco, são representadas à direita; contribuições negativas, que reduzem o risco, aparecem à esquerda. As contribuições são expressas na escala de *log-odds*, e não em probabilidade. A conversão para probabilidade deve ser realizada apenas após a soma de todas as contribuições, por meio da função logística [Lundberg and Lee 2017, Ponce-Bobadilla et al. 2024].

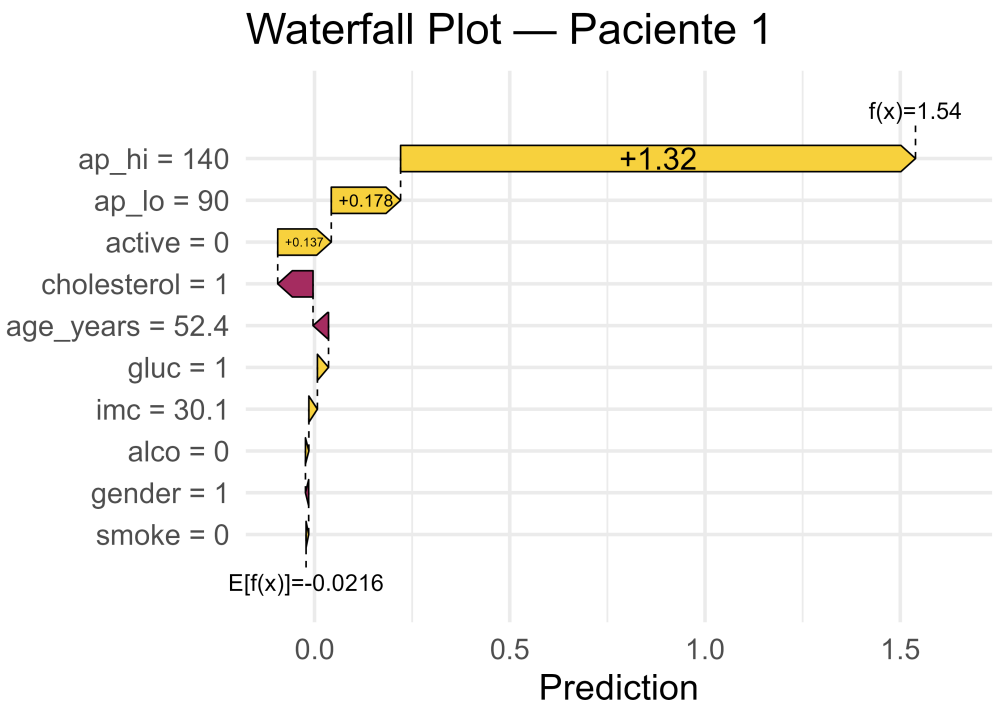


Figura 2.4. Waterfall plot dos valores SHAP para o paciente exemplo, ilustrando a decomposição da predição a partir de ϕ_0 até a predição final $f(x) \approx 1{,}54$.

Na Figura 2.4, observa-se que a pressão arterial sistólica (`ap_hi` = 140 mmHg)

 ©2026 SBC – Soc. Bras. de Computação

é, de longe, o principal fator que eleva o risco predito ($\varphi = +1{,}32$), seguida pela pressão diastólica (`ap_lo` = 90 mmHg; $\varphi = +0{,}18$) e pelo sedentarismo (`active` = 0; $\varphi = +0{,}14$). Por outro lado, o colesterol normal (`cholesterol` = 1) e a idade de 52,4 anos contribuem modestamente para reduzir o risco predito, enquanto as demais variáveis — glicemia, IMC, álcool, sexo e tabagismo — apresentam contribuições próximas de zero. A predição final é $f(x) = 1{,}54$ em escala de *log-odds*, partindo de um valor base de $E[f(x)] = -0{,}02$.

Force plot. O *force plot* apresenta as mesmas informações, mas em um formato horizontal mais compacto. Nesse gráfico, as contribuições positivas (fatores de risco) são representadas por tons quentes, enquanto as contribuições negativas (fatores protetores) aparecem em tons frios. Diferentemente do *waterfall plot*, que enfatiza a sequência de acumulação, o *force plot* destaca o balanço entre forças que aumentam e reduzem o risco [Stenwig et al. 2022, Hu et al. 2024a].

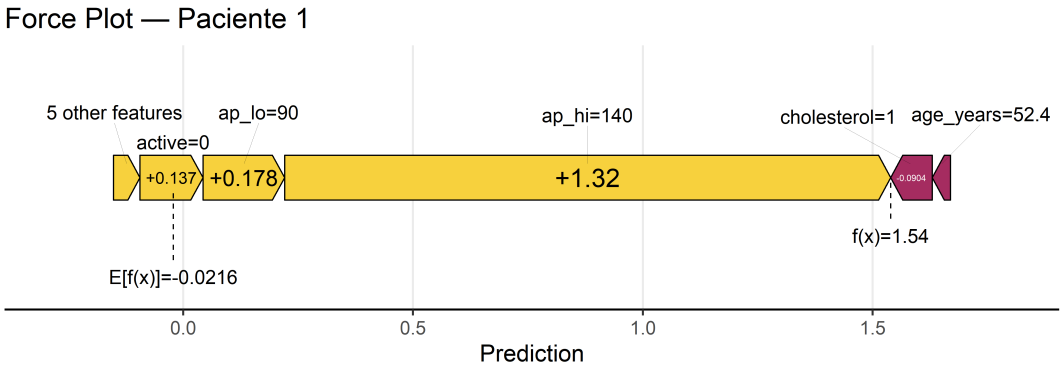


Figura 2.5. Force plot dos valores SHAP para o paciente exemplo. Contribuições positivas (tons quentes) e negativas (tons frios) são apresentadas simultaneamente, ilustrando o deslocamento da predição de ϕ_0 até $f(x)$.

Esse formato é particularmente útil em interfaces clínicas, onde é necessário visualizar rapidamente os fatores mais relevantes para a decisão. No entanto, quando o número de variáveis é elevado, o gráfico pode se tornar mais difícil de interpretar, pois várias contribuições são exibidas simultaneamente [Ponce-Bobadilla et al. 2024, Contreras et al. 2024].

2.7.5. Armadilhas e Limitações na Interpretação Visual

Apesar de sua utilidade, as visualizações SHAP devem ser interpretadas com cautela. Elas facilitam a compreensão do comportamento do modelo, mas também podem induzir conclusões equivocadas se analisadas de forma acrítica.

Correlação não implica causalidade. Uma contribuição elevada de uma variável não significa que ela seja causa direta do desfecho. Por exemplo, o colesterol pode aparecer como um fator importante não apenas por seu papel fisiológico, mas também por refletir aspectos indiretos, como hábitos alimentares ou acesso ao sistema de saúde. Nesse sentido, intervenções focadas apenas na variável observada podem não produzir o efeito esperado

 ©2026 SBC – Soc. Bras. de Computação

se os determinantes subjacentes não forem considerados [Ponce-Bobadilla et al. 2024, Salih et al. 2023, Nohara et al. 2021, Bifarin 2022, Hettikankanamage et al. 2025].

Instabilidade com variáveis correlacionadas. Quando há forte correlação entre variáveis, como entre creatinina e taxa de filtração glomerular, as contribuições SHAP podem ser distribuídas de forma instável entre elas. Essa instabilidade pode variar entre execuções, especialmente em métodos que assumem independência entre variáveis ou utilizam estratégias de amostragem [Ponce-Bobadilla et al. 2024, Salih et al. 2023, Hu et al. 2024a].

Dependência da amostra de referência. Os valores SHAP dependem da distribuição dos dados utilizada como referência. O valor base ϕ_0 e as contribuições individuais são calculados com base nessa distribuição. Assim, se a amostra não for representativa da população de interesse, as conclusões sobre a importância das variáveis podem ser distorcidas [Liu et al. 2022, Ponce-Bobadilla et al. 2024].

Extrapolação visual arriscada. Os padrões observados em gráficos, como os *dependence plots*, são válidos apenas dentro da faixa de valores efetivamente presente nos dados. Inferências feitas fora dessas regiões, especialmente em áreas com pouca ou nenhuma observação, podem ser enganosas, sobretudo quando o modelo apresenta comportamento não linear [Hooker et al. 2019, Ponce-Bobadilla et al. 2024].

Ilusão de compreensão completa. Visualizações bem construídas podem transmitir a impressão de que o modelo foi totalmente compreendido. No entanto, os valores SHAP descrevem apenas como o modelo se comporta diante dos dados disponíveis. Eles não revelam, por si só, os mecanismos causais do fenômeno clínico. Trabalhos recentes destacam que interpretações sem validação externa podem ser equivocadas quando tratadas como evidência objetiva [Huang and ao Marques-Silva 2024, Hooshyar and Yang 2024, Singh et al. 2025].

Em conjunto, essas limitações mostram que as visualizações SHAP devem ser entendidas como ferramentas de apoio à interpretação, e não como evidência definitiva. Sua utilização deve ser sempre complementada por conhecimento clínico, análise crítica e validação empírica. A seção seguinte aprofunda essas questões ao discutir limitações metodológicas e boas práticas para o uso responsável do SHAP em ambientes clínicos.

2.8. Limitações, Desafios e Boas Práticas

O uso do SHAP em aplicações clínicas traz benefícios importantes em termos de interpretabilidade, mas também envolve limitações conceituais, desafios técnicos e riscos de interpretação inadequada. Esses aspectos precisam ser reconhecidos e gerenciados de forma ativa. Esta seção discute essas questões e apresenta boas práticas para o uso responsável do método em saúde.

 ©2026 SBC – Soc. Bras. de Computação

2.8.1. Limitações Conceituais Fundamentais

2.8.1.1. SHAP Explica o Modelo, Não o Fenômeno Clínico

A limitação mais importante do SHAP é que ele explica o comportamento do modelo, e não o fenômeno clínico em si. Os valores SHAP refletem as relações que o modelo aprendeu a partir dos dados, incluindo sua estrutura, suas hipóteses implícitas e a distribuição da amostra. No entanto, eles não estabelecem relações causais nem explicam mecanismos fisiopatológicos [Lundberg and Lee 2017, Ponce-Bobadilla et al. 2024, Salih et al. 2023, Bifarin 2022].

Na prática, isso significa que uma variável pode aparecer como altamente relevante no modelo sem ser, de fato, a causa do desfecho. Por exemplo, o número de exames realizados pode estar associado a maior mortalidade hospitalar não porque cause o desfecho, mas porque pacientes mais graves tendem a receber mais exames. Interpretar esse resultado de forma causal levaria a decisões equivocadas ditadas por vieses de presença no sistema de saúde [Momenzadeh et al. 2022].

As explicações SHAP devem ser interpretadas como indicações de como o modelo utiliza as variáveis, e não como evidência causal. Para responder a perguntas causais, são necessários métodos específicos, como modelos estruturais, variáveis instrumentais ou ensaios clínicos [Chen et al. 2021]. Nesse sentido, o SHAP é mais adequado como ferramenta de geração de hipóteses, que deve ser integrada ao conhecimento clínico e a outras fontes de evidência empírica [Hettikankanamage et al. 2025, Viswan et al. 2023].

2.8.1.2. Dependência da Qualidade e Representatividade dos Dados

A qualidade das explicações depende diretamente da qualidade dos dados. Se o modelo é treinado com dados enviesados ou incompletos, as explicações refletirão essas limitações.

Vieses de seleção podem comprometer a generalização do modelo. Erros de mensuração podem distorcer as associações aprendidas. Vieses históricos, como subtratamento de determinados grupos, podem ser reproduzidos e até evidenciados pelas explicações SHAP [Obermeyer et al. 2019, Chen et al. 2021]. Além disso, a presença de confundidores não controlados pode levar o modelo a aprender associações que não representam relações reais [Ponce-Bobadilla et al. 2024, Chen et al. 2021].

Outro ponto importante é a escolha do conjunto de referência (*background*) utilizado para calcular os valores SHAP. Amostras pequenas, desbalanceadas ou não representativas podem introduzir variabilidade excessiva na estimativa de importância das variáveis [Liu et al. 2022] e até permitir manipulação das explicações por meio de seleção enviesada de dados (ataques adversariais), ocultando comportamentos discriminatórios do modelo [Wood et al. 2023, Xin et al. 2025].

 ©2026 SBC – Soc. Bras. de Computação

2.8.2. Desafios Técnicos e Metodológicos

2.8.2.1. Multicolinearidade e Ambiguidade de Atribuição

Em dados clínicos, é comum que variáveis estejam correlacionadas. Nesses casos, o SHAP distribui a contribuição entre elas de acordo com os axiomas de Shapley. Embora essa decomposição seja matematicamente consistente, ela pode ser difícil de interpretar do ponto de vista clínico.

Por exemplo, variáveis como pressão arterial sistólica e diastólica ou diferentes marcadores renais podem compartilhar informações semelhantes. O resultado é que a importância pode ser diluída entre essas variáveis, dificultando a identificação de qual delas é mais relevante na prática [Salih et al. 2023, Hu et al. 2024b]. Em métodos aproximados, esse problema pode ser agravado pela geração de combinações de dados pouco realistas durante o cálculo das contribuições [Hu et al. 2024b].

Para lidar com esse problema, é recomendável analisar variáveis correlacionadas em conjunto, explorar valores de interação SHAP e documentar explicitamente as relações conhecidas entre variáveis [Contreras et al. 2024, Lundberg et al. 2020].

2.8.2.2. Complexidade Computacional em Escala

O cálculo exato dos valores de Shapley é computacionalmente complexo e, em muitos casos, inviável. Embora o TreeSHAP torne esse cálculo eficiente para modelos baseados em árvores [Lundberg et al. 2020], o custo ainda pode ser elevado em bases muito grandes.

Na prática, estratégias como amostragem estratificada, cálculo sob demanda, paralelização e armazenamento de resultados são utilizadas para tornar o método viável em larga escala. Abordagens adicionais, como o uso de agrupamentos (clustering) combinado com SHAP, também podem reduzir o custo mantendo boa fidelidade das explicações [Bachmann 2025]. Sempre que métodos aproximados forem utilizados, seu impacto deve ser avaliado e documentado.

2.8.2.3. Estabilidade e Reprodutibilidade

Os valores SHAP podem variar entre diferentes execuções do modelo. Mudanças na divisão dos dados, na inicialização do algoritmo ou nos hiperparâmetros podem alterar o ranking de importância das variáveis. Esse efeito é particularmente relevante em métodos baseados em amostragem, como o KernelSHAP [Kelodjou et al. 2024, Bachmann 2025].

Para garantir maior robustez, recomenda-se avaliar a estabilidade das explicações. Isso pode ser feito treinando múltiplos modelos, utilizando reamostragem dos dados e verificando se os padrões observados se mantêm consistentes. Em aplicações críticas, deve-se dar maior peso a variáveis cujas contribuições são estáveis em diferentes configurações.

 ©2026 SBC – Soc. Bras. de Computação

2.8.3. Riscos de Uso Indevido em Contextos Clínicos

As visualizações SHAP, por serem intuitivas e visualmente claras, podem gerar uma falsa sensação de compreensão completa do modelo. Isso pode levar a excesso de confiança nas predições ou a interpretações equivocadas [Singh et al. 2025, Salih et al. 2023].

Em ambientes clínicos com alta carga cognitiva, como unidades de terapia intensiva, explicações muito detalhadas podem contribuir para fadiga de alertas. Nesses casos, é preferível apresentar informações de forma concisa, destacando apenas os fatores mais relevantes [Ponce-Bobadilla et al. 2024, Rasheed et al. 2021, Patel et al. 2024].

Além disso, explicações locais não garantem que a predição esteja correta. Elas apenas mostram como o modelo chegou àquele resultado. Portanto, não devem ser utilizadas como única base para decisões clínicas de alto risco [Singh et al. 2025]. A responsabilidade final permanece com o profissional de saúde [Kheder et al. 2025].

2.8.4. Boas Práticas para Aplicações em Saúde

O uso responsável do SHAP requer atenção ao longo de todo o ciclo de vida do modelo [Ponce-Bobadilla et al. 2024, Hu et al. 2022].

Antes da análise, é fundamental garantir que o modelo tenha desempenho adequado, incluindo boa calibração, validação externa e avaliação por subgrupos. Também é importante documentar todo o pipeline e definir claramente os objetivos da interpretação [Kheder et al. 2025, Chen et al. 2021].

Durante a análise, recomenda-se combinar explicações globais e locais, avaliar a plausibilidade clínica dos resultados, investigar discrepâncias com especialistas e considerar diferentes subgrupos populacionais [Hakkoum et al. 2024, Prendin et al. 2023]. A escala utilizada (probabilidade ou *log-odds*) deve ser explicitada [Ponce-Bobadilla et al. 2024].

Na comunicação dos resultados, a linguagem deve ser adaptada ao público, evitando interpretações causais indevidas [Arrieta et al. 2020, Bifarin 2022]. As limitações do método devem ser explicitadas, e sempre que possível as interpretações devem ser validadas com profissionais de saúde [Ghasemi et al. 2024].

Em produção, é importante monitorar continuamente o comportamento do modelo e de suas explicações, armazenar resultados para auditoria e revisar o sistema após atualizações ou mudanças na população atendida [Ponce-Bobadilla et al. 2024, Stogiannos et al. 2023, Singh et al. 2025].

Em síntese, o SHAP é uma ferramenta poderosa para interpretação de modelos, mas seu uso seguro exige análise crítica, integração com conhecimento clínico e atenção constante às suas limitações.

A seção seguinte discute as implicações éticas, regulatórias e de governança associadas ao uso de modelos explicáveis em saúde.

 ©2026 SBC – Soc. Bras. de Computação

2.9. Considerações Éticas, Regulatórias e Governança de Modelos Explicáveis em Saúde

O uso de modelos explicáveis em saúde não envolve apenas questões técnicas, mas também implicações éticas, legais e organizacionais. A adoção responsável dessas ferramentas exige atenção a regulamentações, definição clara de responsabilidades, avaliação de equidade e implementação de mecanismos de governança ao longo de todo o ciclo de vida do modelo.

Leis e diretrizes recentes, como a LGPD no Brasil, o GDPR na União Europeia e o *EU AI Act*, estabelecem requisitos explícitos relacionados à transparência e à prestação de contas em decisões automatizadas [Goodman and Flaxman 2017, Rasheed et al. 2021, Khan et al. 2024, Aldhafeeri 2025]. Nesse contexto, métodos como o SHAP podem contribuir para tornar essas exigências operacionais, ao fornecer explicações sobre o funcionamento dos modelos.

No entanto, é importante reconhecer que a simples disponibilização de explicações técnicas não é suficiente. Essas explicações precisam ser traduzidas para formatos compreensíveis e relevantes para o contexto clínico. Além disso, a explicabilidade não substitui outras obrigações legais, como o consentimento informado, a proteção de dados pessoais e a minimização do uso de informações sensíveis [Amann et al. 2020].

Embora o SHAP contribua para tornar decisões mais transparentes e rastreáveis, ele não transfere a responsabilidade para o sistema automatizado. A decisão final continua sendo do profissional de saúde, que deve interpretar as informações fornecidas pelo modelo à luz do contexto clínico [Kiseleva et al. 2022, Kheder et al. 2025]. Assim, a *accountability* permanece inerente ao agente humano, mesmo quando há apoio de sistemas baseados em inteligência artificial.

As explicações SHAP podem ser utilizadas para identificar possíveis vieses no modelo, especialmente quando as contribuições são analisadas em diferentes subgrupos populacionais. No entanto, essa análise tem limitações. Variáveis que funcionam como *proxies* de atributos protegidos, como raça ou condição socioeconômica, podem ocultar mecanismos de discriminação estrutural [Chen et al. 2021, Obermeyer et al. 2019]. Por isso, a avaliação de equidade não deve se basear apenas em explicações, mas também em métricas específicas de *fairness* e análises complementares [Amann et al. 2020, Tung et al. 2025, Chen et al. 2021].

Em sistemas classificados como *Software as a Medical Device* (SaMD), órgãos regulatórios como FDA, ANVISA e entidades europeias exigem documentação detalhada sobre o propósito clínico, o desempenho do modelo e seus mecanismos de explicação [Rasheed et al. 2021, Aldhafeeri 2025, U.S. Food and Drug Administration 2021, Kiseleva et al. 2022].

Para atender a essas exigências, é necessário estabelecer uma estrutura de governança que acompanhe todo o ciclo de vida do modelo. Isso inclui a participação de equipes multidisciplinares, validação antes da implantação, monitoramento contínuo após o uso em produção e manutenção de documentação versionada e auditável [Reddy et al. 2020, Upadhyay et al. 2023, Stogiannos et al. 2023, Singh et al. 2025].

 ©2026 SBC – Soc. Bras. de Computação

Em síntese, a explicabilidade é um componente importante, mas não suficiente para o uso responsável de modelos em saúde. Ela deve ser integrada a um conjunto mais amplo de práticas que envolvem governança, responsabilidade e avaliação contínua. As considerações finais retomam esses pontos à luz do estado atual e das perspectivas futuras do campo.

2.10. Considerações Finais e Perspectivas

O SHAP consolidou-se, na última década, como uma das principais abordagens para explicabilidade de modelos de aprendizado de máquina em saúde. Esse destaque não é acidental. Sua base nos valores de Shapley fornece garantias matemáticas que muitos métodos heurísticos não possuem: a decomposição da predição é exata, as contribuições individuais somam ao valor final do modelo e variáveis irrelevantes não recebem importância [Lundberg and Lee 2017, Bifarin 2022]. Além disso, a existência de implementações eficientes, como o TreeSHAP, e a possibilidade de análise tanto local quanto global tornam o método particularmente adequado ao contexto clínico, que exige compreender decisões individuais e, ao mesmo tempo, avaliar padrões populacionais.

Ao longo deste capítulo, foi possível observar, no entanto, que essas propriedades não eliminam limitações importantes. Os valores SHAP descrevem o comportamento do modelo, não o fenômeno clínico em si. Eles capturam associações presentes nos dados, mas não estabelecem relações causais [Bifarin 2022, Hettikankanamage et al. 2025]. Além disso, suas interpretações dependem da qualidade dos dados, da presença de variáveis correlacionadas e das escolhas feitas na implementação [Hu et al. 2024a, Liu et al. 2022].

Esses limites não invalidam o uso do SHAP, mas indicam a necessidade de uma aplicação cuidadosa. As explicações devem ser interpretadas em conjunto com o conhecimento clínico, avaliadas quanto à plausibilidade e analisadas em diferentes subgrupos [Prendin et al. 2023, Hakkoum et al. 2024]. Também é fundamental explicitar as limitações do método, especialmente em relação ao que ele não permite afirmar.

Um ponto central é que explicabilidade não deve ser tratada como um objetivo isolado. Uma explicação pode ser matematicamente correta e ainda assim pouco útil na prática clínica, se não for compreensível, relevante ou acionável [Tonekaboni et al. 2019]. O valor do SHAP depende de sua integração em um contexto mais amplo, que inclui validação clínica rigorosa, análise de equidade, consideração de aspectos causais e estruturas de governança bem definidas [Stogiannos et al. 2023].

Diversas questões permanecem em aberto e representam oportunidades de avanço. A integração entre métodos de explicabilidade e abordagens causais formais pode ampliar o valor interpretativo das análises, aproximando-as de decisões clínicas interventivas. A adaptação do SHAP a dados longitudinais e multimodais, cada vez mais comuns em sistemas de saúde, ainda demanda desenvolvimento metodológico [Hettikankanamage et al. 2025, Aldhafeeri 2025]. Além disso, há uma lacuna importante na avaliação do impacto das explicações sobre desfechos clínicos reais, indo além de medidas de usabilidade ou aceitação por parte dos usuários [Hettikankanamage et al. 2025, Singh et al. 2025]. Por fim, a consolidação de padrões regulatórios específicos para explicabilidade em sistemas classificados como *Software as a Medical Device* permanece um desafio relevante [U.S. Food and Drug Administration 2021, Singh et al. 2025].

 ©2026 SBC – Soc. Bras. de Computação

Dessa forma, o SHAP é uma ferramenta poderosa para tornar modelos mais transparentes e auditáveis, mas seu valor depende de como é utilizado. Quando aplicado de forma criteriosa e integrado ao conhecimento clínico e às exigências regulatórias, ele contribui para o desenvolvimento de sistemas de inteligência artificial mais confiáveis e responsáveis [Amann et al. 2020]. Esse é o objetivo final da explicabilidade: não apenas tornar modelos compreensíveis, mas garantir que possam ser utilizados de forma segura, crítica e justificada na prática clínica.

Referências

[Alabi et al. 2023] Alabi, R. O., Elmusrati, M., Leivo, I., Almangush, A., and Mäkitie, A. (2023). Machine learning explainability in nasopharyngeal cancer survival using lime and shap. *Scientific Reports*, 13.

[Aldhafeeri 2025] Aldhafeeri, F. (2025). Governing artificial intelligence in radiology: A systematic review of ethical, legal, and regulatory frameworks. *Diagnostics*, 15.

[Amann et al. 2020] Amann, J., Blasimme, A., Vayena, E., Frey, D., Madai, V. I., and Consortium, P. (2020). Explainability for artificial intelligence in healthcare: a multidisciplinary perspective. *BMC medical informatics and decision making*, 20(1):310.

[Arrieta et al. 2020] Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115.

[Auzine et al. 2024] Auzine, M. M., Khan, M. H.-M., Baichoo, S., Sahib, N. G., Bissoonauth-Daiboo, P., Gao, X., and Heetun, Z. (2024). Development of an ensemble cnn model with explainable ai for the classification of gastrointestinal cancer. *PLOS ONE*, 19.

[Bachmann 2025] Bachmann, S. (2025). Efficient xai: A low-cost data reduction approach to shap interpretability. *J. Artif. Intell. Res.*, 83.

[Bajwa et al. 2023] Bajwa, A., Nosheen, N., Talpur, K., and Akram, S. (2023). A prospective study on diabetic retinopathy detection based on modify convolutional neural network using fundus images at sindh institute of ophthalmology and visual sciences. *Diagnostics*, 13.

[Belle and Papantonis 2020] Belle, V. and Papantonis, I. (2020). Principles and practice of explainable machine learning. *Frontiers in Big Data*, 4.

[Bifarin 2022] Bifarin, O. O. (2022). Interpretable machine learning with tree-based shapley additive explanations: Application to metabolomics datasets for binary classification. *PLOS ONE*, 18.

[Binzagr 2024] Binzagr, F. (2024). Explainable ai-driven model for gastrointestinal cancer classification. *Frontiers in medicine*, 11:1349373.

 ©2026 SBC – Soc. Bras. de Computação

[Bodria et al. 2021] Bodria, F., Giannotti, F., Guidotti, R., Naretto, F., Pedreschi, D., and Rinzivillo, S. (2021). Benchmarking and survey of explanation methods for black box models. *Data Mining and Knowledge Discovery*, 37:1719–1778.

[Brenning 2021] Brenning, A. (2021). Interpreting machine-learning models in transformed feature space with an application to remote-sensing classification. *Machine Learning*, 112:3455–3471.

[Burrell 2016] Burrell, J. (2016). How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data and Society*, 3.

[Cha et al. 2021] Cha, Y., Shin, J., Go, B., Lee, D.-S., Kim, Y., Kim, T., and Park, Y.-S. (2021). An interpretable machine learning method for supporting ecosystem management: Application to species distribution models of freshwater macroinvertebrates. *Journal of environmental management*, 291:112719.

[Chen et al. 2024] Chen, C., Isa, N. A. M., and Liu, X. (2024). A review of convolutional neural network based methods for medical image classification. *Computers in biology and medicine*, 185:109507.

[Chen et al. 2021] Chen, I. Y., Pierson, E., Rose, S., Joshi, S., Ferryman, K., and Ghassemi, M. (2021). Ethical machine learning in healthcare. *Annual review of biomedical data science*, 4(1):123–144.

[Contreras et al. 2024] Contreras, J., Winterfeld, A., Popp, J., and Bocklitz, T. (2024). Spectral zones-based shap/lime: Enhancing interpretability in spectral deep learning models through grouped feature analysis. *Analytical Chemistry*, 96:15588 – 15597.

[Covert et al. 2020] Covert, I., Lundberg, S. M., and Lee, S.-I. (2020). Understanding global feature contributions with additive importance measures. *Advances in neural information processing systems*, 33:17212–17223.

[Daoud and Bayoumi 2019] Daoud, H. G. and Bayoumi, M. (2019). Efficient epileptic seizure prediction based on deep learning. *IEEE Transactions on Biomedical Circuits and Systems*, 13:804–813.

[Elshawi et al. 2019] Elshawi, R., Al-Mallah, M., and Sakr, S. (2019). On the interpretability of machine learning-based model for predicting hypertension. *BMC Medical Informatics and Decision Making*, 19.

[Esteva et al. 2017] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., and Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118.

[Gevaert and Saeys 2022] Gevaert, A. and Saeys, Y. (2022). Pdd-shap: Fast approximations for shapley values using functional decomposition.

[Ghasemi et al. 2024] Ghasemi, A., Hashtarkhani, S., Schwartz, D. L., and Shaban-Nejad, A. (2024). Explainable artificial intelligence in breast cancer detection and risk prediction: A systematic scoping review. *Cancer Innovation*, 3.

 ©2026 SBC – Soc. Bras. de Computação

[Ghosh and Khandoker 2024] Ghosh, S. K. and Khandoker, A. (2024). Investigation on explainable machine learning models to predict chronic kidney diseases. *Scientific Reports*, 14.

[Goldstein et al. 2017] Goldstein, B. A., Navar, A. M., Pencina, M. J., and Ioannidis, J. P. (2017). Opportunities and challenges in developing risk prediction models with electronic health records data: a systematic review. *Journal of the American Medical Informatics Association*, 24(1):198–208.

[Goodman and Flaxman 2017] Goodman, B. and Flaxman, S. (2017). European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57.

[Greenwell 2017] Greenwell, B. M. (2017). pdp: An r package for constructing partial dependence plots. *R J.*, 9:421.

[Greenwell and Boehmke 2020] Greenwell, B. M. and Boehmke, B. C. (2020). Variable importance plots - an introduction to the vip package. *R J.*, 12:343.

[Hakkoum et al. 2024] Hakkoum, H., Idri, A., and Abnane, I. (2024). Global and local interpretability techniques of supervised machine learning black box models for numerical medical data. *Eng. Appl. Artif. Intell.*, 131:107829.

[Hannun et al. 2019] Hannun, A. Y., Rajpurkar, P., Haghpanahi, M., Tison, G. H., Bourn, C., Turakhia, M. P., and Ng, A. Y. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature Medicine*, 25(1):65–69.

[Harrigan et al. 2022] Harrigan, C. F., Morgenshtern, G., Goldenberg, A., and Chevalier, F. (2022). Considerations for visualizing uncertainty in clinical machine learning models.

[Hassija et al. 2023] Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., Scardapane, S., Spinelli, I., Mahmud, M., and Hussain, A. (2023). Interpreting black-box models: A review on explainable artificial intelligence. *Cognitive Computation*, 16:45–74.

[Hettikankanamage et al. 2025] Hettikankanamage, N. D., Shafiabady, N., Chatteur, F., Wu, R. M. X., Din, F. U., and Zhou, J. (2025). explainable artificial intelligence (xai): A systematic review for unveiling the black box models and their relevance to biomedical imaging and sensing. *Sensors (Basel, Switzerland)*, 25.

[Hooker et al. 2019] Hooker, G., Mentch, L., and Zhou, S. (2019). Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, 31.

[Hooshyar and Yang 2024] Hooshyar, D. and Yang, Y. (2024). Problems with shap and lime in interpretable ai for education: A comparative study of post-hoc explanations and neural-symbolic rule extraction. *IEEE Access*, 12:137472–137490.

 ©2026 SBC – Soc. Bras. de Computação

[Hu et al. 2024a] Hu, C., Gao, C., Li, T., Liu, C., and Peng, Z. (2024a). Explainable artificial intelligence model for mortality risk prediction in the intensive care unit: a derivation and validation study. *Postgraduate medical journal*, 100(1182):219–227.

[Hu et al. 2022] Hu, C., Li, L., ping Huang, W., Wu, T., Xu, Q., Liu, J., and Hu, B. (2022). Interpretable machine learning for early prediction of prognosis in sepsis: A discovery and validation study. *Infectious Diseases and Therapy*, 11:1117 – 1132.

[Hu et al. 2024b] Hu, X., Zhu, M., Feng, Z., and Stanković, L. (2024b). Manifold-based shapley explanations for high dimensional correlated features. *Neural networks : the official journal of the International Neural Network Society*, 180:106634.

[Huang and ao Marques-Silva 2024] Huang, X. and ao Marques-Silva, J. (2024). On the failings of shapley values for explainability. *Int. J. Approx. Reason.*, 171:109112.

[Joachim et al. 2026] Joachim, K., Sparks, O., Perrotta, A., Lin, A., Gettleman, B., Hamad, C., Jeong, S., Dingle, E., Stavrakis, A., and Christ, A. B. (2026). Evaluating the methodological suitability of partial dependence plots and shapley additive explanations for population-level interpretation of machine learning models in total joint arthroplasty. *Arthroplasty*, 8.

[Kelodjou et al. 2024] Kelodjou, G., Rozé, L., Masson, V., Galárraga, L., Gaudel, R., Tchuente, M., and Termier, A. (2024). Shaping up shap: enhancing stability through layer-wise neighbor selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 13094–13103.

[Khan et al. 2024] Khan, M. M., Shah, N., Shaikh, N., Thabet, A., Alrabayah, T., and belkhair, S. (2024). Towards secure and trusted ai in healthcare: A systematic review of emerging innovations and ethical challenges. *International journal of medical informatics*, 195:105780.

[Kheder et al. 2025] Kheder, W., Leblouba, M., Rego, R., and Hamdoon, Z. (2025). Multicentre validation and clinical interpretation of an explainable gradient-boosting model for dental-implant survival/failure prediction. *Journal of dentistry*, page 106166.

[Kiseleva et al. 2022] Kiseleva, A., Kotzinos, D., and Hert, P. (2022). Transparency of ai in healthcare as a multilayered system of accountabilities: Between legal requirements and technical limitations. *Frontiers in Artificial Intelligence*, 5.

[Lin et al. 2025] Lin, Q., Zhao, W., Zhang, H., Chen, W., Lian, S., Ruan, Q., Qu, Z., Lin, Y., Chai, D., and Lin, X. (2025). Predicting the risk of heart failure after acute myocardial infarction using an interpretable machine learning model. *Frontiers in Cardiovascular Medicine*, 12.

[Lip et al. 2010] Lip, G. Y., Nieuwlaat, R., Pisters, R., Lane, D. A., and Crijns, H. J. (2010). Refining clinical risk stratification for predicting stroke and thromboembolism in atrial fibrillation using a novel risk factor-based approach: the euro heart survey on atrial fibrillation. *Chest*, 137(2):263–272.

 ©2026 SBC – Soc. Bras. de Computação

[Lipton 2018] Lipton, Z. C. (2018). The mythos of model interpretability. *Queue*, 16(3):31–57.

[Liu et al. 2022] Liu, M., Ning, Y., Yuan, H., Ong, M. E. H., and Liu, N. (2022). Balanced background and explanation data are needed in explaining deep learning models with shap: An empirical study on clinical decision making.

[Lundberg and Lee 2017] Lundberg, S. and Lee, S.-I. (2017). A unified approach to interpreting model predictions.

[Lundberg et al. 2020] Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67.

[Lundberg et al. 2019] Lundberg, S. M., Erion, G. G., and Lee, S.-I. (2019). Consistent individualized feature attribution for tree ensembles.

[Luo et al. 2024] Luo, H., Xiang, C., Zeng, L., Li, S., Mei, X., Xiong, L., Liu, Y., Wen, C., Cui, Y., Du, L., Zhou, Y., Wang, K., Li, L., Liu, Z., Wu, Q., Pu, J., and Yue, R. (2024). Shap based predictive modeling for 1 year all-cause readmission risk in elderly heart failure patients: feature selection and model interpretation. *Scientific Reports*, 14.

[Mahmoudi et al. 2020] Mahmoudi, E., Kamdar, N., Kim, N., Gonzales, G., Singh, K., and Waljee, A. (2020). Use of electronic medical records in development and validation of risk prediction models of hospital readmission: systematic review. *The BMJ*, 369.

[Markus et al. 2021] Markus, A. F., Kors, J. A., and Rijnbeek, P. R. (2021). The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of biomedical informatics*, 113:103655.

[Mienye et al. 2025] Mienye, I. D., Swart, T. G., Obaido, G., Jordan, M., and Ilono, P. (2025). Deep convolutional neural networks in medical image analysis: A review. *Inf.*, 16:195.

[Miotto et al. 2018] Miotto, R., Wang, F., Wang, S., Jiang, X., and Dudley, J. T. (2018). Deep learning for healthcare: review, opportunities and challenges. *Briefings in Bioinformatics*, 19(6):1236–1246.

[Molnar 2020] Molnar, C. (2020). *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. 3 edition.

[Momenzadeh et al. 2022] Momenzadeh, A., Shamsa, A., and Meyer, J. G. (2022). Bias or biology? importance of model interpretation in machine learning studies from electronic health records. *JAMIA Open*, 5.

[Nohara et al. 2021] Nohara, Y., Matsumoto, K., Soejima, H., and Nakashima, N. (2021). Explanation of machine learning models using shapley additive explanation and application for real data in hospital. *Computer methods and programs in biomedicine*, 214:106584.

 ©2026 SBC – Soc. Bras. de Computação

[Obermeyer et al. 2019] Obermeyer, Z., Powers, B., Vogeli, C., and Mullainathan, S. (2019). Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453.

[Olsen and Jullum 2025] Olsen, L. H. B. and Jullum, M. (2025). Improving the weighting strategy in kernelshap. In *World Conference on Explainable Artificial Intelligence*, pages 194–218. Springer.

[Patel et al. 2024] Patel, A. M., Baxter, W., and Porat, T. (2024). Toward guidelines for designing holistic integrated information visualizations for time-critical contexts: Systematic review. *Journal of Medical Internet Research*, 26.

[Patharkar et al. 2024] Patharkar, A., Cai, F., Al-Hindawi, F., and Wu, T. (2024). Predictive modeling of biomedical temporal data in healthcare applications: review and future directions. *Frontiers in Physiology*, 15.

[Ponce-Bobadilla et al. 2024] Ponce-Bobadilla, A. V., Schmitt, V., Maier, C., Mensing, S., and Stodtmann, S. (2024). Practical guide to shap analysis: Explaining supervised machine learning model predictions in drug development. *Clinical and Translational Science*, 17.

[Prendin et al. 2023] Prendin, F., Pavan, J., Cappon, G., Favero, S. D., Sparacino, G., and Facchinetti, A. (2023). The importance of interpreting machine learning models for blood glucose prediction in diabetes: an analysis using shap. *Scientific Reports*, 13.

[Rajkomar et al. 2018] Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., et al. (2018). Scalable and accurate deep learning with electronic health records. *NPJ Digital Medicine*, 1(1):18.

[Rasheed et al. 2021] Rasheed, K., Qayyum, A., Ghaly, M., Al-Fuqaha, A. I., Razi, A., and Qadir, J. (2021). Explainable, trustworthy, and ethical machine learning for healthcare: A survey. *Computers in biology and medicine*, 149:106043.

[Reddy et al. 2020] Reddy, S., Allan, S., Coghlan, S., and Cooper, P. (2020). A governance model for the application of ai in health care. *Journal of the American Medical Informatics Association*, 27(3):491–497.

[Ribeiro et al. 2016] Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144.

[Roshinta and Gábor 2024] Roshinta, T. A. and Gábor, S. (2024). A comparative study of lime and shap for enhancing trustworthiness and efficiency in explainable ai systems. *2024 IEEE International Conference on Computing (ICOCO)*, pages 134–139.

[Rudin 2019] Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215.

 ©2026 SBC – Soc. Bras. de Computação

[Salih et al. 2023] Salih, A. M. A., Raisi-Estabragh, Z., Galazzo, I., Radeva, P., Petersen, S. E., Lekadir, K., and Menegaz, G. (2023). A perspective on explainable artificial intelligence methods: Shap and lime. *Advanced Intelligent Systems*, 7.

[Selvaraju et al. 2020] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. (2020). Grad-cam: visual explanations from deep networks via gradient-based localization. *International journal of computer vision*, 128(2):336–359.

[Shapley 1953] Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317. Reprinted in: Kuhn, H.W. (ed.) Classics in Game Theory, Princeton University Press, 1997.

[Singh et al. 2025] Singh, Y., Hathaway, Q. A., Keishing, V., Salehi, S., Wei, Y., Horvat, N., Vera-Garcia, D. V., Choudhary, A., Kh, A. M., Quaia, E., and Andersen, J. (2025). Beyond post hoc explanations: A comprehensive framework for accountable ai in medical imaging through transparency, interpretability, and explainability. *Bioengineering*, 12.

[Stenwig et al. 2022] Stenwig, E., Salvi, G., Rossi, P. S., and Skjærvold, N.-K. (2022). Comparative analysis of explainable machine learning prediction models for hospital mortality. *BMC Medical Research Methodology*, 22.

[Stiglic et al. 2020] Stiglic, G., Kocbek, P., Fijacko, N., Zitnik, M., Verbert, K., and Cilar, L. (2020). Interpretability of machine learning-based prediction models in healthcare. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 10.

[Stogiannos et al. 2023] Stogiannos, N., Malik, R., Kumar, A., Barnes, A., Pogose, M., Harvey, H., McEntee, M., and Malamateniou, C. (2023). Black box no more: a scoping review of ai governance frameworks to guide procurement and adoption of ai in medical imaging and radiotherapy in the uk. *The British Journal of Radiology*, 96.

[Tan et al. 2023] Tan, Z., Tian, Y., and Li, J. (2023). Glime: General, stable and local lime explanation.

[Tonekaboni et al. 2019] Tonekaboni, S., Joshi, S., McCradden, M. D., and Goldenberg, A. (2019). What clinicians want: contextualizing explainable machine learning for clinical end use. *Proceedings of Machine Learning Research*, 106:359–380. Machine Learning for Healthcare Conference.

[Topol 2019] Topol, E. J. (2019). High-performance medicine: the convergence of human and artificial intelligence. *Nature Medicine*, 25(1):44–56.

[Tun et al. 2024] Tun, H., Rahman, H., Naing, L., and Malik, O. A. (2024). Trust in artificial intelligence–based clinical decision support systems among health care workers: Systematic review. *Journal of Medical Internet Research*, 27.

[Tung et al. 2025] Tung, T., Hasnaeen, S. M. N., and Zhao, X. (2025). Ethical and practical challenges of generative ai in healthcare and proposed solutions: a survey. *Frontiers in Digital Health*, 7.

 ©2026 SBC – Soc. Bras. de Computação

[Upadhyay et al. 2023] Upadhyay, U., Gradisek, A., Iqbal, U., Dhar, E., Li, Y., and Syed-Abdul, S. (2023). Call for the responsible artificial intelligence in the healthcare. *BMJ Health & Care Informatics*, 30.

[U.S. Food and Drug Administration 2021] U.S. Food and Drug Administration (2021). Artificial intelligence/machine learning (AI/ML)-based software as a medical device (SaMD) action plan. Accessed: 2026.

[Vimbi et al. 2024] Vimbi, V., Shaffi, N., and Mahmud, M. (2024). Interpreting artificial intelligence models: a systematic review on the application of lime and shap in alzheimer's disease detection. *Brain Informatics*, 11.

[Vincent et al. 1996] Vincent, J.-L., Moreno, R., Takala, J., Willatts, S., De Mendonça, A., Bruining, H., Reinhart, C. K., Suter, P., and Thijs, L. G. (1996). The sofa (sepsis-related organ failure assessment) score to describe organ dysfunction/failure: On behalf of the working group on sepsis-related problems of the european society of intensive care medicine (see contributors to the project in the appendix). *Intensive care medicine*, 22(7):707–710.

[Viswan et al. 2023] Viswan, V., Shaffi, N., Mahmud, M., Subramanian, K., and Hajamohideen, F. (2023). Explainable artificial intelligence in alzheimer's disease classification: A systematic review. *Cognitive Computation*, 16:1–44.

[Wang et al. 2024] Wang, H., Liang, Q., Hancock, J. T., and Khoshgoftaar, T. (2024). Feature selection strategies: a comparative analysis of shap-value and importance-based methods. *Journal of Big Data*, 11:1–16.

[Wood et al. 2023] Wood, D., Papamarkou, T., Benatan, M., and Allmendinger, R. (2023). Model-agnostic variable importance for predictive uncertainty: an entropy-based approach. *Data Mining and Knowledge Discovery*, 38:4184 – 4216.

[working group and risk collaboration 2021] working group, S. and risk collaboration, E. C. (2021). Score2 risk prediction algorithms: new models to estimate 10-year risk of cardiovascular disease in europe. *European Heart Journal*, 42(25):2439–2454.

[Xin et al. 2025] Xin, X., Hooker, G., and Huang, F. (2025). Pitfalls in machine learning interpretability: Manipulating partial dependence plots to hide discrimination. *Insurance: Mathematics and Economics*, page 103135.

[Yang et al. 2022] Yang, X., Chen, A., Pournejatian, N. M., Shin, H.-C., Smith, K. E., Parisien, C., Compas, C. B., Martin, C., Costa, A. B., Flores, M. G., Zhang, Y., Magoc, T., Harle, C., Lipori, G. P., Mitchell, D. A., Hogan, W., Shenkman, E., Bian, J., and Wu, Y. (2022). A large language model for electronic health records. *NPJ Digital Medicine*, 5.

[Yuan et al. 2025] Yuan, K., Yoon, C. H., Gu, Q., Munby, H., Walker, A., Zhu, T., and Eyre, D. W. (2025). Transformers and large language models are efficient feature extractors for electronic health record studies. *Communications Medicine*, 5.

 ©2026 SBC – Soc. Bras. de Computação

[Zafar and Khan 2021] Zafar, M. R. and Khan, N. (2021). Deterministic local interpretable model-agnostic explanations for stable explainability. *Mach. Learn. Knowl. Extr.*, 3:525–541.

[Zeng et al. 2025] Zeng, X., Chen, J., Zeng, X., Tang, X., and Peng, J. (2025). Integrating multiparametric mri radiomics and clinical models to assess sensitivity to neoadjuvant chemotherapy in breast cancer: A multicenter study. *Journal of Applied Clinical Medical Physics*, 26.

 ©2026 SBC – Soc. Bras. de Computação

Capítulo 3

Construção de Agente de Conversação com Aplicação na Saúde: Do Problema Real à Solução Inteligente

Amanda Morais Almeida, Diêgo Farias de Freitas, Paulo Eduardo Ambrósio

Abstract

This chapter addresses the development of intelligent conversational agents as a solution to structural challenges and opportunities in the healthcare sector. The proposal is based on the need for Artificial Intelligence (AI) adoption to automate processes, minimize errors, and reduce the cognitive load of professionals, utilizing an architecture grounded in Large Language Models (LLMs) integrated with Retrieval-Augmented Generation (RAG). This study describes essential technical foundations, including the role of vector embeddings, context management via chunking, and the orchestration of clinical workflows through the LangChain framework. To ensure patient safety and mitigate hallucinations, it discusses the implementation of technical supervision layers called Guardrails and the use of multi-agent systems for decision auditing. The text concludes by analyzing implementation challenges in heterogeneous infrastructures, the necessity of compliance with regulatory frameworks, and perspectives for digital health that keeps the human in the loop

 ©2026 SBC – Soc. Bras. de Computação

Resumo

Este capítulo aborda o desenvolvimento de agentes conversacionais inteligentes como solução para desafios e oportunidades estruturais no setor de saúde. A proposta parte da necessidade de adesão à Inteligência Artificial (IA) para a automatização de processos, minimização de erros e redução da carga cognitiva de profissionais, utilizando uma arquitetura baseada em Modelos de Linguagem de Grande Escala (LLMs) integrados à Geração Aumentada por Recuperação (RAG). A presente investigação descreve os fundamentos técnicos essenciais, incluindo o papel dos embeddings vetoriais, a gestão de contexto via chunking e a orquestração de fluxos clínicos através do framework LangChain. Para garantir a segurança do paciente e mitigar alucinações, é discutida a implementação de camadas de supervisão técnica denominadas Guardrails e o uso de sistemas multiagentes para a auditoria de decisões. O texto conclui analisando os desafios de implementação em infraestruturas heterogêneas, a necessidade de conformidade com marcos regulatórios e as perspectivas para uma saúde digital que mantenha o ser humano no centro do processo (human-in-the-loop).

3.1. Agentes de IA na Transformação Digital da Saúde

A Inteligência Artificial (IA) tem desempenhado um papel transformador no setor de saúde, embora grande parte de suas aplicações atuais ainda opere de forma restrita a tarefas isoladas, enfrentando barreiras como a complexidade dos dados clínicos e a presença de vieses algorítmicos. Essa limitação tecnológica ocorre em um momento de crise estrutural no setor que, de acordo com Karunanayake (2025), projeta-se que haverá um déficit global de 18 milhões de profissionais de saúde até 2030. Esse dado evidencia a possibilidade do significativo impacto no acesso ao cuidado, especialmente em regiões de baixa renda. Diante desse cenário, a integração de tecnologias digitais deixa de ser apenas uma inovação para se tornar uma estratégia fundamental dos sistemas médicos modernos, visando a otimização de fluxos de trabalho e a democratização de soluções inteligentes.

No âmbito prático dessa transformação, a eficiência da triagem em departamentos de emergência é frequentemente comprometida pela natureza dinâmica do ambiente clínico. Segundo Suamchaiyaphum et al. (2024), a acurácia dos enfermeiros é diretamente influenciada por fatores como o volume de pacientes e a carga horária de trabalho, o que pode gerar um desalinhamento entre a necessidade clínica e a decisão tomada. Essa lacuna de conhecimento e a variabilidade nos resultados de triagem evidenciam a urgência de sistemas de apoio que reduzam a carga cognitiva e garantam a aderência aos protocolos, independentemente das pressões ambientais.

Adicionalmente, a aplicação da triagem na Atenção Primária à Saúde (APS) revela desafios distintos. Conforme apontado por Park et al. (2025), embora a triagem conduzida por enfermeiros apresente níveis aceitáveis de concordância clínica, há uma tendência persistente ao over-referral que é descrito como encaminhamento excessivo para serviços de emergência como estratégia de segurança. Nesse cenário, a transição para

 ©2026 SBC – Soc. Bras. de Computação

sistemas baseados em IA e *machine learning* surge não apenas como uma inovação, mas como uma necessidade para aprimorar a priorização clínica. O suporte de ferramentas digitais e algoritmos validados permite que o profissional de saúde atue de forma mais precisa na seleção do local de cuidado, otimizando os recursos do sistema e garantindo que o acesso do paciente seja pautado por evidências estruturadas, reduzindo a subjetividade inerente ao atendimento remoto ou síncrono.

Para mitigar tais limitações, a transição para sistemas de maior confiabilidade e com arquitetura multicamadas marca o início da chamada era agêntica. Conforme detalha Karunanayake (2025), este momento é caracterizado por sistemas de agentes de IA com funcionalidade autônoma, raciocínio avançado e interações dinâmicas. Diferente das aplicações tradicionais, essa tecnologia incorpora arquiteturas adaptáveis que aumentam a autonomia na tomada de decisão. No contexto da triagem, isso significa que o sistema utiliza um raciocínio probabilístico para atualizar previsões continuamente com base em novos dados, funcionando como uma solução capaz de mitigar a escassez de profissionais ao automatizar tarefas administrativas e aprimorar a eficiência do fluxo de trabalho na porta de entrada dos serviços de saúde.

Contudo, essa transformação digital não se resume à automação de processos, mas à criação de um ecossistema onde a inteligência artificial atua como um extensor das capacidades humanas. Sob a perspectiva da Modelagem Computacional, o desafio reside na representação fiel dos protocolos clínicos em modelos de decisão que sejam, simultaneamente, flexíveis para a linguagem natural e rígidos para a segurança do paciente. Assim, a era agêntica na saúde pressupõe uma infraestrutura que suporte o aprendizado contínuo e a explicabilidade, transformando o vasto volume de dados clínicos em estratégia para otimização de fluxos operacionais, maior segurança na prestação de serviços e no desenvolvimento de produtos para os pacientes.

3.2. Fundamentos Técnicos para Construção de Agentes

Este tópico dedica-se à construção da base teórica necessária para viabilizar o desenvolvimento do agente, permitindo uma compreensão aprofundada de como cada componente técnico influencia diretamente no desempenho e na confiabilidade do resultado final. A investigação percorre as camadas fundamentais do sistema com o objetivo de demonstrar a atuação de cada tecnologia e os estudos existentes que validam sua eficácia em cenários de alta criticidade. Ao detalhar as definições conceituais e suas finalidades específicas, o texto estabelece as metodologias adotadas e as correlaciona com as alternativas e paradigmas de automação vigentes na literatura.

A fundamentação parte de uma análise dos componentes arquiteturais que permitem a transformação de uma necessidade real em uma solução inteligente e tecnicamente segura. Para isso, o capítulo aprofunda-se no Processamento de Linguagem Natural (PLN) e no papel dos *Large Language Models* (LLM) como núcleos de processamento cognitivo. Em seguida, são explorados os *embeddings* como técnica de representação vetorial e as estratégias de *chunking* e janelas de memória, elementos

 ©2026 SBC – Soc. Bras. de Computação

essenciais para a gestão de contexto que compõe a arquitetura de *Retrieval-Augmented Generation* (RAG), traduzida como Geração Aumentada por Recuperação. Esta estrutura é refinada pelas técnicas de engenharia de *prompt*, que otimizam a interação. Por fim, abordam-se os *guardrails*, voltados à garantia da segurança sistêmica, e a orquestração via LangChain, que viabiliza a integração eficiente de todas as camadas do agente.

A compreensão dos agentes inteligentes no setor saúde exige a análise das tecnologias que permitem a transição de sistemas estáticos para modelos dinâmicos de interação. A arquitetura desses sistemas baseia-se em avanços significativos no Processamento de Linguagem Natural (PLN) e na Ciência de Dados, permitindo que diferentes classes de agentes desempenhem funções específicas no fluxo hospitalar. A Tabela 1 sintetiza essa diversidade tecnológica, categorizando os agentes de acordo com suas aplicações, categorias de saúde e bases computacionais [Karunanayake 2025].

Tabela 3.1. Categorização de tipos de agentes de IA na saúde [Karunanayake 2025].

Agentes de IA	Principais Aplicações	Categorias de Saúde	Principais Usuários	Principais Tecnologias de IA
Agentes baseados em imagem	Diagnóstico de doenças, detecção precoce, geração de relatórios.	Diagnóstico, suporte à decisão clínica.	Radiologistas, Médicos.	Visão computacional (CNNs, ViTs), MLLMs para integração imagem-texto.
Agentes de análise preditiva	Predição de risco, previsão de progressão de doenças, resultados dos pacientes.	Suporte à Decisão Clínica, Tratamento e Cuidado, Descoberta de Fármacos.	Médicos, Equipes de Cuidado.	Modelagem Preditiva (ML supervisionado, ensemble, séries temporais).
Agentes conversacio-Nais	Verificação de sintomas, triagem de pacientes, consultas virtuais.	Engajamento e Monitoramento do Paciente.	Pacientes, Clínicos Gerais.	PLN, Sistemas de Diálogo, LLMs Pré-treinados.
Agentes de PLN (NLP)	Processamento de notas clínicas, sumarização de EHRs, extração de insights.	Operações e Administração, Suporte à Decisão Clínica.	Codificadores Médicos, Analistas.	PLN, LLMs Pré-treinados.
Agentes baseados em regras	Seguimento de diretrizes clínicas, alerta para interações medicamentosas.	Suporte à Decisão Clínica.	Médicos, Farmacêuticos.	Raciocínio Baseado em Regras, lógica, grafos de conhecimento.

©2026 SBC – Soc. Bras. de Computação

Agentes híbridos	Combinação de imagem, texto, vídeo e análise preditiva para decisões.	Suporte à Decisão Clínica, Diagnóstico, Cirurgia Robótica.	Médicos, Radiologistas , Cirurgiões.	Aprendizado Multimodal.
Agentes de ML	Classificação de doenças, detecção de anomalias, planejamento de tratamento.	Diagnóstico, Tratamento, Descoberta de Fármacos.	Cientistas de Dados, Médicos.	Algoritmos de ML/DL, Aprendizado por Reforço (RL).
Agentes de sistemas especialistas	Emulação de expertise clínica para diagnóstico e planejamento.	Suporte à Decisão Clínica, Cirurgia Robótica.	Especialistas, Pesquisadores , Cirurgiões.	Sistemas baseados em conhecimento e regras.
Agentes recomenda-Dores	Sugestão de testes diagnósticos, tratamentos personalizados.	Tratamento e Cuidado, Suporte à Decisão Clínica.	Médicos, Equipes de Cuidado.	Filtragem colaborativa, sistemas de recomendação, RL.

A análise da Tabela 3.1 revela que o desenvolvimento de agentes conversacionais e sistemas híbridos para triagem clínica depende fundamentalmente de LLMs pré-treinados e de arquiteturas multimodais. A orquestração desses componentes visa transformar a entrada de dados não estruturados em decisões clínicas seguras e rastreáveis. Para compreender como essa conversão de linguagem bruta em decisão estruturada ocorre, é imperativo analisar o Processamento de Linguagem Natural como a camada primária de interface e inteligência do sistema.

3.2.1. Processamento de Linguagem Natural (PLN)

O Processamento de Linguagem Natural (PLN) é a disciplina da Inteligência Artificial que permite que máquinas realizem os processos de leitura e interpretação da comunicação humana. Segundo Jurafsky e Martin (2026), o principal desafio do PLN não é apenas identificar palavras, mas construir representações contextuais de significado. Para um sistema de saúde, isso significa entender que a palavra “aguda” em uma triagem possui um peso semântico de urgência que termos administrativos não possuem.

Para compreender a fundamentação dos agentes inteligentes atuais, é necessário observar a trajetória tecnológica que moldou o campo. Esta evolução, conforme discutido por Zaniboni (2025) e sintetizada por Jeevan (2023), reflete a transição de métodos estatísticos para modelos de entendimento profundo. Essa progressão histórica e as principais características de cada período estão detalhadas na Tabela 3.2.

 ©2026 SBC – Soc. Bras. de Computação

Tabela 3.2 – Eras da evolução do Processamento de Linguagem Natural [Jeevan 2023].

Era	Período	Abordagem Principal	Características e Limitações
Baseada em Regras	1960 — 1980	Lógica simbólica (*se-então*)	Operava com caminhos rígidos criados por especialistas. Falhava em entender nuances, gírias ou contextos informais.
Estatística	1980 — 2000	Frequência e Probabilidade (ex: TF-IDF)	Identificava termos importantes pela frequência, mas ignorava a ordem das palavras e o significado real das frases.
Redes Neurais e *Embeddings*	2000 — 2017	Vetores Densos e RNNs	Palavras similares passaram a ser vizinhas em um espaço matemático. Contudo, a leitura sequencial dificultava o contexto em frases longas.
Era Agêntica	2017 — Presente	Transformers e Atenção	Processamento paralelo e global. O modelo consegue correlacionar informações distantes em um mesmo texto.

A inovação que sustenta os agentes modernos é a arquitetura Transformer, que introduziu o conceito de autoatenção para resolver o problema da ambiguidade [Vaswani et al. 2017]. Na prática, a atenção funciona como um filtro seletivo que permite ao modelo identificar quais palavras em uma frase são mais importantes para definir o sentido de um termo específico. À medida que um *token*, uma palavra ou parte dela, percorre esse fluxo, cada camada do Transformer adiciona informações contextuais a ele, sem apagar a informação original.

Jurafsky e Martin (2026) explicam que essa arquitetura opera através de um fluxo residual, onde cada camada do sistema adiciona novas camadas de entendimento sobre a palavra original sem descartar o que foi processado anteriormente. Ou seja, o modelo consegue realizar verificações para trás e para frente simultaneamente, permitindo que o significado de um termo seja enriquecido por palavras que apareceram muito antes no texto. Isso permite que um agente identifique, por exemplo, que o termo "pressão" em uma conversa clínica refere-se à medida arterial do paciente e não a uma força física ou emocional, baseando-se nas palavras vizinhas como "hipertensão" ou "aparelho".

O componente que permite essa possibilidade é a autoatenção ou *self-attention*. Didaticamente, a atenção funciona como um filtro de relevância. Quando o agente

 ©2026 SBC – Soc. Bras. de Computação

processa a frase "O paciente apresenta dor intensa no peito", o mecanismo de atenção faz com que a palavra "intensa" indique com maior relevância para o modelo quando ele analisa "dor", enquanto palavras como "no" ou "o" são ignoradas. Matematicamente, isso é feito através de três vetores: *Query* (Consulta), *Key* (Chave) e *Value* (Valor). No exemplo citado, o vetor *Query* pergunta "O que eu estou procurando?", o *Key* responde "O que eu tenho de informação?" e o *Value* entrega o conteúdo relevante. Essa interação resulta em Mapas de Atenção que são visualizações que mostram as conexões que o modelo faz entre as palavras.

Para ilustrar esse funcionamento, a Figura 3.1 apresenta o comportamento das *attention heads* ou cabeças de atenção, ao processarem a palavra *it*, traduzida como ele ou ela, em uma sentença. Nota-se que, enquanto uma cabeça foca no sujeito *the animal*, outra foca no adjetivo *tired*. Em termos de modelagem, isso significa que a representação final da palavra *it* funde ou absorve parte das características de ambos os termos, permitindo que o modelo compreenda o contexto completo da frase.

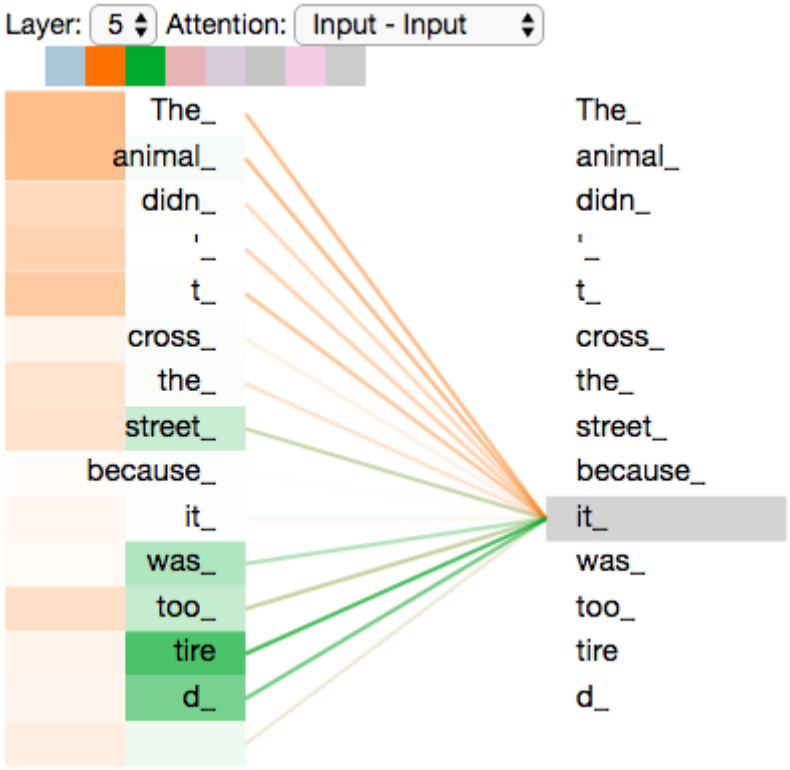


Figura 3.1 – Visualização das *attention heads* focando em diferentes dependências semânticas [Alammar 2018].

O desenvolvimento de um agente inteligente depende dessa capacidade de capturar o conhecimento de mundo através do pré-treinamento generativo [Radford et al. 2018]. Esse processo permite que o modelo aprenda a estrutura da linguagem em grandes acervos de livros e documentos antes de ser ajustado para a tarefa específica. Para a

©2026 SBC – Soc. Bras. de Computação

construção do agente, essa base teórica garante que o sistema não apenas repita frases programadas, mas que consiga realizar transformações de entrada para lidar com diferentes tipos de perguntas e interações, mantendo a coerência e a lógica durante todo o evento.

Na área da saúde, PLN exerce um papel estratégico, uma vez que viabiliza o tratamento da desestruturação típica de prontuários e relatos médicos [Nascimento 2024]. O sistema consegue realizar a extração de entidades nomeadas, identificando automaticamente sintomas e dosagens de medicamentos em meio a textos longos, o que agiliza o fluxo de trabalho hospitalar e reduz a carga cognitiva da equipe assistencial. Além disso, a análise de sentimento permite ao agente capturar o estado emocional ou o nível de desconforto relatado pelo paciente, oferecendo uma camada de humanização e precisão diagnóstica que sistemas puramente estatísticos não conseguiriam atingir.

A integração dessas camadas tecnológicas resulta na criação de sistemas que ao processar palavras, interpretam a gravidade e a urgência contidas na linguagem natural. Esta base teórica é o que permite que os *Large Language Models* (LLMs) compreendam a fundo as necessidades dos pacientes, elevando a capacidade de resposta do agente a níveis de precisão comparáveis ao entendimento humano em tarefas específicas de suporte à decisão.

3.2.2. *Large Language Models* (LLM)

Os Modelos de Linguagem de Grande Escala, amplamente conhecidos pela sigla LLM, constituem o estágio mais avançado na evolução do Processamento de Linguagem Natural. Em termos estruturais, um LLM é um modelo baseado na arquitetura Transformer que foi treinado em uma escala sem precedentes de dados e parâmetros. Segundo Brown et al. (2020), a principal característica que define esses sistemas é o surgimento de habilidades cognitivas complexas conforme o modelo é escalado para centenas de bilhões de parâmetros, como observado no GPT-3. Para a literatura, esse fenômeno de escala é quase comparável a uma lei da física, onde capacidades imprevistas de raciocínio e síntese emergem de forma previsível à medida que o volume de dados e o poder computacional aumentam [Tamkin et al. 2021].

O diferencial fundamental entre um LLM e as ferramentas de PLN tradicionais reside na sua natureza generalista. Enquanto modelos antigos eram treinados para tarefas específicas, como classificar sentimentos ou traduzir frases, os LLMs são definidos como "aprendizes de poucos exemplos" ou *few-shot learners*. Isso significa que eles possuem a capacidade intrínseca de compreender e executar novas tarefas recebendo apenas algumas demonstrações textuais dentro do contexto da conversa, sem a necessidade de atualizar seus pesos matemáticos ou passar por um novo treinamento [Brown et al. 2020].

 ©2026 SBC – Soc. Bras. de Computação

Para ilustrar essa versatilidade, a Figura 3.2 apresenta as três configurações principais de uso desses modelos. No modo *zero-shot*, o modelo recebe apenas uma descrição da tarefa, no *one-shot*, é fornecido um único exemplo de demonstração e no *few-shot*, o modelo é condicionado com alguns exemplos antes da execução final. Nota-se que, ao contrário do ajuste fino tradicional conhecido como *fine-tuning*, essas abordagens não realizam atualizações de gradiente, operando puramente por meio da interação textual.

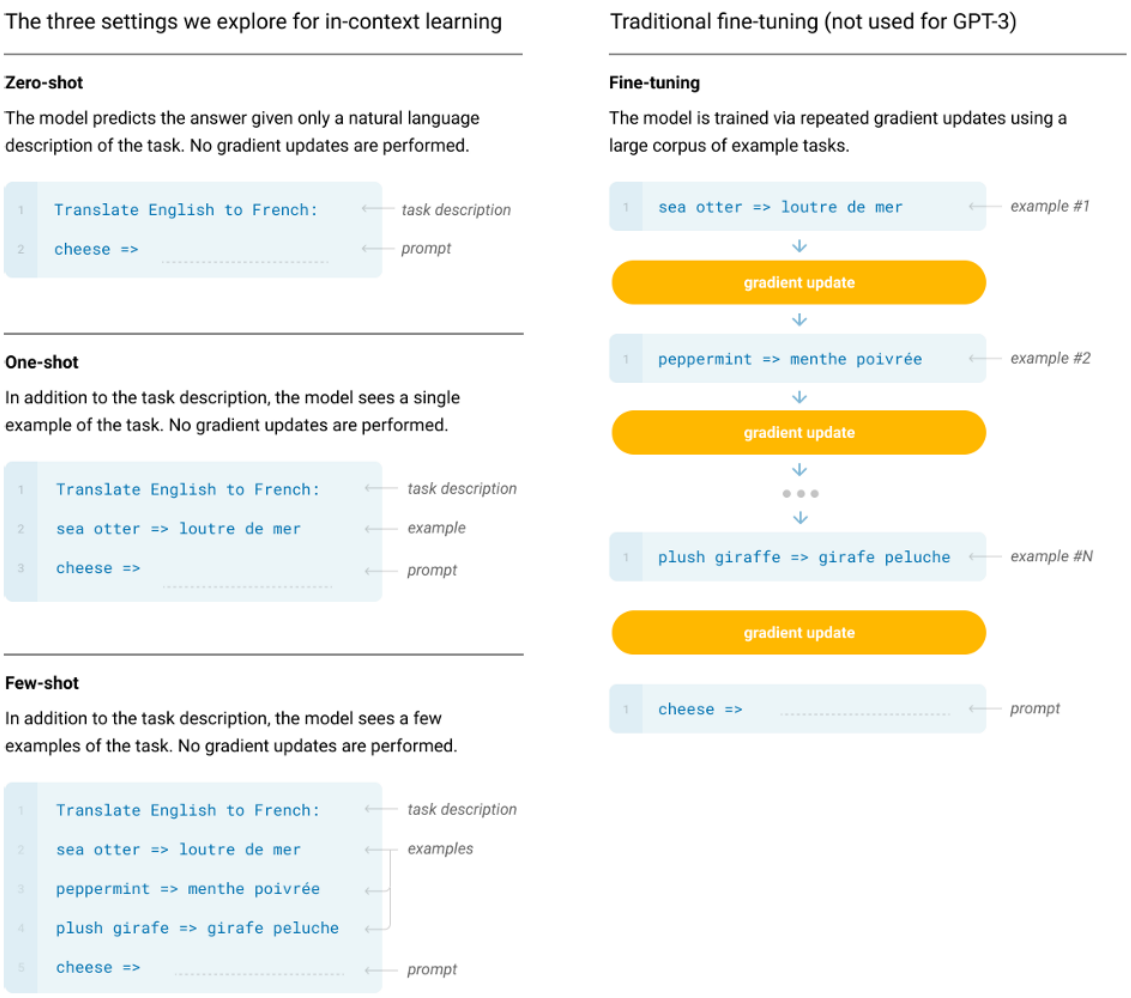


Figura 3.2 – Comparativo entre as abordagens de aprendizado de máquina [Brown et al. 2020]

O funcionamento desses modelos baseia-se no pré-treinamento generativo, um processo em que o sistema consome trilhões de *tokens* extraídos de livros, artigos científicos, códigos de programação e conversas da internet [Touvron et al. 2023]. Esse estágio inicial permite que o modelo capture um vasto conhecimento de mundo e aprenda a estrutura profunda da linguagem. No entanto, o treinamento bruto não é suficiente para garantir um comportamento útil ou seguro. Conforme discutido por Tamkin et al. (2021), é necessário realizar um processo denominado de alinhamento, que ajusta os objetivos

 ©2026 SBC – Soc. Bras. de Computação

estatísticos do modelo aos valores e intenções humanas, garantindo que o agente inteligente responda de forma útil e ética.

Didaticamente, o uso de um LLM para a construção de agentes inteligentes ocorre por meio do *design* de comandos, ou engenharia de *prompts*. Diferente da programação tradicional, onde o desenvolvedor define regras lógicas rígidas, o sistema é guiado por linguagem natural. O LLM atua como o núcleo de processamento do agente, sendo capaz de manter a coerência em diálogos longos, resumir informações complexas e gerar respostas estruturadas. Segundo Touvron et al. (2023), modelos menores e mais eficientes, como a série LLaMA, demonstraram que o refinamento dos dados de treinamento pode ser tão impactante quanto o número de parâmetros, permitindo o uso dessas tecnologias em infraestruturas locais de pesquisa.

A diversidade de LLMs disponíveis atualmente pode ser classificada primordialmente entre modelos proprietários e modelos de código aberto denominados *open-source*. Modelos proprietários, como a série GPT da OpenAI, geralmente oferecem as maiores escalas de parâmetros e performance de estado da arte, mas operam sob acesso restrito via API, sigla para Interface de Programação de Aplicações, o que limita a transparência sobre os dados de treinamento e o controle sobre a privacidade das informações [Brown et al. 2020]. Em contrapartida, coleções de modelos abertos, como o LLaMA da Meta AI, democratizaram o acesso à tecnologia ao permitir que modelos competitivos, variando de 7B a 65B de parâmetros, sejam executados em infraestruturas de pesquisa locais, garantindo maior controle sobre o fluxo de dados e a auditoria do sistema [Touvron et al. 2023].

A escolha do modelo ideal para uma aplicação específica deve considerar o equilíbrio entre o desempenho desejado e o orçamento de inferência. Conforme demonstrado por Touvron et al. (2023), o desempenho não é determinado exclusivamente pelo tamanho do modelo. Modelos menores treinados com um volume massivo de dados podem superar modelos gigantes, como o GPT-3 de 175B, em diversos *benchmarks*. Para o desenvolvimento de agentes inteligentes, modelos mais compactos e eficientes são frequentemente preferíveis, pois reduzem a latência na resposta e os custos computacionais sem comprometer a precisão necessária para tarefas de raciocínio e síntese.

Além da escala, a seleção deve ser guiada pela capacidade de aprendizado em contexto e pelo alinhamento com a tarefa pretendida [Tamkin et al. 2021]. Em cenários de alta sensibilidade, como a saúde, deve-se priorizar modelos que apresentem menores taxas de alucinação e maior resiliência a vieses, avaliados por métricas de veracidade como o *TruthfulQA*. Assim, a modelagem de um agente inteligente clínico exige uma avaliação criteriosa se o modelo escolhido possui a superfície de capacidade necessária para processar a linguagem técnica biomédica mantendo a conformidade com os requisitos éticos de segurança e privacidade [Weidinger et al. 2021].

 ©2026 SBC – Soc. Bras. de Computação

As contribuições dos LLMs para a ciência moderna são significativas, especialmente na democratização do acesso a ferramentas de inteligência avançada. Eles permitem a tradução de nuances culturais que sistemas antigos ignoravam e a síntese de grandes volumes de literatura técnica em segundos. Exemplos práticos incluem assistentes virtuais capazes de depurar códigos de programação complexos, sistemas de escrita colaborativa e ferramentas de busca que, em vez de apenas listar *links*, constroem respostas fundamentadas e contextualizadas.

No cenário da saúde, a transição para os LLMs potencializa as técnicas de PLN citadas anteriormente, permitindo que a interpretação de prontuários ultrapasse a mera identificação de palavras-chave para alcançar uma compreensão semântica profunda dos relatos clínicos. A capacidade generativa permite que o agente não apenas reconheça dados desestruturados, mas sintetize históricos complexos, infira relações de causalidade entre sintomas e organize informações biomédicas em formatos padronizados de forma automatizada. Essa evolução transforma o processamento de texto em uma ferramenta ativa de suporte à decisão, agilizando fluxos assistenciais em ambientes de alta criticidade [Brown et al. 2020]. A arquitetura desse processamento clínico, conforme detalhada na Figura 3.3, ilustra justamente o fluxo de transformação do relato bruto do paciente em uma saída técnica estruturada.

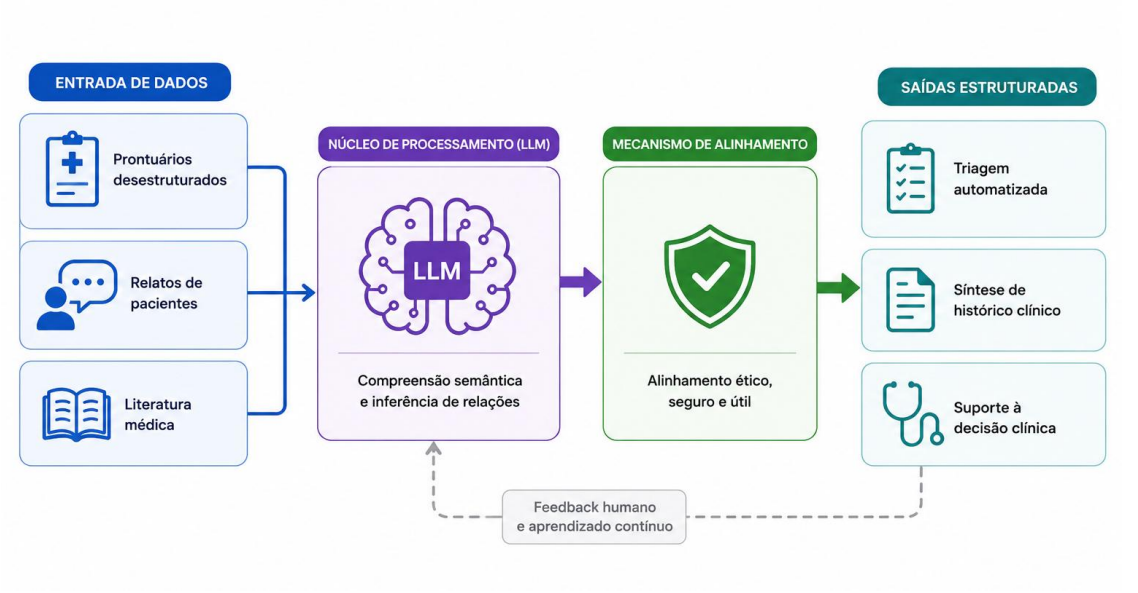


Figura 3.3 – Arquitetura de processamento clínico via LLM

Entretanto, as implicações do uso de LLMs na saúde exigem uma governança rigorosa. Weidinger et al. (2021) alertam para o risco de desinformação médica, onde o modelo, por ser puramente estatístico, pode gerar orientações clinicamente perigosas que parecem factuais mas são, na verdade, alucinações. Como os modelos aprendem padrões de associação e não a verdade científica intrínseca, eles podem falhar em distinguir entre

 ©2026 SBC – Soc. Bras. de Computação

conselhos médicos baseados em evidências e desinformação presente em seus dados de treinamento [Weidinger et al. 2021].

Outro ponto de atenção reside na reprodução de estereótipos sociais nocivos. LLMs treinados em vastos conjuntos de dados da internet podem espelhar e até amplificar preconceitos de raça, gênero ou religião [Touvron et al. 2023]. Em um ambiente de diagnóstico médico, essa propagação de vieses pode levar a um atendimento desigual para grupos minoritários, conforme alertado na taxonomia de riscos éticos desses modelos [Weidinger et al. 2021]. A literatura enfatiza que a representação injusta de identidades sociais pode criar danos materiais diretos se o sistema for utilizado para alocação de recursos hospitalares ou definição de prioridades clínicas.

Além dos riscos de conteúdo, a segurança da informação é uma preocupação latente. Os LLMs podem, inadvertidamente, memorizar e vazar informações sensíveis ou dados privados que estavam presentes em seus acervos de treinamento [Weidinger et al. 2021]. No contexto da saúde, onde a privacidade dos dados do paciente é um requisito ético e legal rigoroso, a implementação desses modelos requer camadas adicionais de anonimização e técnicas de treinamento que garantam a integridade contra vazamentos acidentais.

A interação humana com esses modelos também apresenta desafios psicológicos. O fenômeno da atribuição de características humanas pode levar pacientes a superdimensionarem as competências do agente inteligente por ele falar de maneira fluida e humanizada [Weidinger et al. 2021]. Essa confiança indevida pode fazer com que usuários ignorem a necessidade de supervisão médica ou aceitem diagnósticos automatizados sem o devido ceticismo clínico, o que reforça a importância de manter sempre um profissional humano no ciclo de decisão.

Para estruturar esses riscos, a Tabela 3.3 resume a taxonomia de possíveis danos associados aos modelos de linguagem de grande escala. Essa classificação é essencial para orientar o desenvolvimento de estratégias de mitigação eficazes.

 ©2026 SBC – Soc. Bras. de Computação

Tabela 3.3 – Taxonomia de riscos éticos e sociais associados aos LLMs [Weidinger et al. 2021]

Área de Risco	Mecanismo de Origem	Tipos de Dano
I. Discriminação, Exclusão e Toxicidade	O modelo reflete com precisão padrões de fala injustos, tóxicos ou opressores presentes nos dados de treino.	Ofensa, danos materiais (locacionais) e tratamento injusto ou representação estereotipada de grupos marginalizados.
II. Riscos de Informação (Information Hazards)	O modelo prevê enunciados que constituem informações privadas ou críticas para a segurança presentes ou inferidas dos dados de treino.	Violações de privacidade de dados sensíveis e riscos diretos à segurança do indivíduo.
III. Danos por Desinformação	O modelo atribui altas probabilidades a informações falsas, enganosas, sem sentido ou de baixa qualidade.	Engano, danos materiais e ações antiéticas por humanos que aceitam a previsão do modelo como um facto, além da erosão da confiança social.
IV. Usos Maliciosos	Utilização intencional do modelo por atores humanos com o objetivo explícito de causar dano.	Enfraquecimento do discurso público, fraudes, campanhas de desinformação personalizadas e produção de código informático malicioso.
V. Danos na Interação Humano-Computador	Riscos decorrentes da interação direta via diálogo, onde o utilizador interage com o sistema como se fosse um agente humano.	Uso inseguro devido ao erro de julgamento do utilizador, vulnerabilidades psicológicas e perpetuação de estereótipos via design do produto (ex: assistentes femininas).
VI. Automação, Acesso e Danos Ambientais	Utilização de LLMs em aplicações que beneficiam desproporcionalmente alguns grupos em detrimento de outros.	Aumento das desigualdades sociais, perda de empregos de alta qualidade, desigualdade no acesso à tecnologia e custos ambientais elevados.

Para mitigar esses riscos, a literatura sugere a aplicação de protocolos de inovação responsável e testes rigorosos de segurança, conhecidos como *red-teaming* [Tamkin et al. 2021]. Não basta que o modelo seja tecnicamente avançado, ele deve ser alinhado e constantemente verificado contra falhas éticas. A maturidade técnica dos LLMs oferece

 ©2026 SBC – Soc. Bras. de Computação

mecanismos necessários para interpretar a linguagem natural com precisão, entretanto é a governança desses sistemas que garantirá sua utilidade clínica.

3.2.3. *Embeddings* para Representação Vetorial

Os *embeddings* constituem o mecanismo fundamental que permite aos LLMs compreenderem a semântica de um texto. Eles são representações matemáticas de palavras, frases ou documentos inteiros em um espaço vetorial de alta dimensionalidade. Diferentemente de representações simbólicas tradicionais, os *embeddings* convertem texto em vetores numéricos onde a proximidade geométrica entre dois pontos reflete a similaridade semântica dos termos que eles representam [Jeevan 2023].

Na prática, o processo envolve submeter o texto a um modelo de linguagem capaz de gerar *embeddings*, como o BERT ou modelos de *embedding* da OpenAI, que produzem vetores numéricos correspondentes. Esses vetores podem ser armazenados em bancos de dados vetoriais, permitindo consultas eficientes baseadas em similaridade.

A Figura 3.4 ilustra como diferentes conceitos médicos são mapeados em um espaço tridimensional, mostrando a proximidade entre termos semanticamente relacionados.

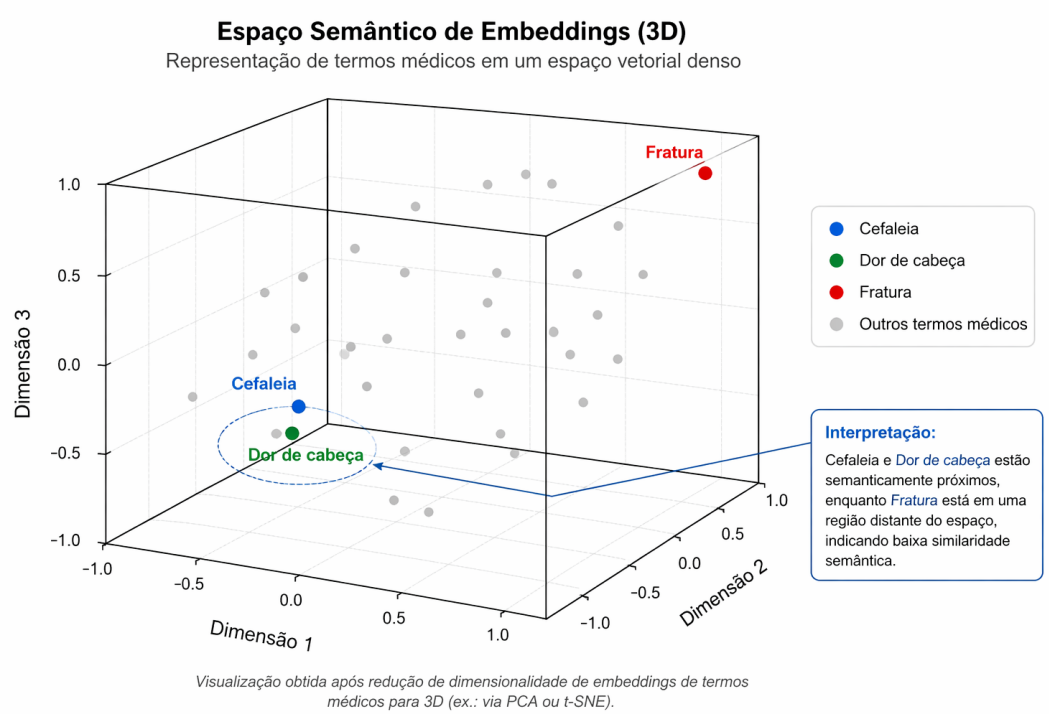


Figura 3.4 – Visualização conceitual de *embeddings* médicos em espaço vetorial

A representação evidencia que termos semanticamente semelhantes ocupam regiões próximas no espaço vetorial, enquanto conceitos distintos apresentam maior

 ©2026 SBC – Soc. Bras. de Computação

separação. Existem diversos tipos de modelos de embedding, desde os mais simples e estáticos, como o modelo Word2Vec, até os mais modernos, profundos e contextuais baseados em arquiteturas Transformers. Para a área da saúde, é comum o uso de modelos pré-treinados em grandes *corpora* de literatura biomédica, como o BioBERT, que capturam com maior precisão as nuances do vocabulário técnico.

Diferente de modelos estáticos, onde uma palavra possui um único vetor invariável, os *embeddings* modernos são contextuais. Isso permite que o sistema diferencie o sentido de termos polissêmicos com base nas palavras vizinhas, o que é importante para reduzir ambiguidades em diagnósticos [Tamkin et al. 2021]. A principal implicação do uso de *embeddings* na saúde reside, portanto, na superação da busca por palavras-chave exatas. Em um sistema de triagem, isso permite que o agente identifique que os termos "cefaléia intensa" e "dor de cabeça aguda" ocupam regiões próximas no espaço vetorial, ainda que não compartilhem a mesma grafia. Segundo Lewis et al. (2020), essa representação é a base para a recuperação de informações em larga escala, permitindo que o modelo de linguagem processe conceitos clínicos complexos de forma estruturada e matematicamente comparável.

3.2.4. *Chunking* e Janelas de Memória como Gestão de Contexto

A gestão de contexto refere-se às técnicas utilizadas para lidar com o chamado limite de tokens ou limite de contexto dos LLMs. Este limite é a quantidade máxima de informação que o modelo consegue processar de uma só vez. As duas estratégias principais são o *chunking* e as janelas de memória. O *chunking* é o processo de segmentar documentos extensos em fragmentos menores e logicamente coerentes [Dong et al. 2024]. Já as janelas de memória constituem técnicas de filtragem e retenção que garantem que o agente selecione e recupere as partes cruciais de uma interação em uma conversa de longa duração, evitando a saturação do limite de *tokens* e a perda de informações relevantes [Akheel 2025].

O *chunking* é aplicado antes de converter o texto em *embeddings*, dividindo os documentos médicos, como protocolos hospitalares, em blocos de tamanho fixo ou baseados em parágrafos. Um fator crítico nesta etapa é a sobreposição entre os blocos, chamada de *overlap*, que garante que a transição entre um fragmento e outro não resulte em perda de significado. As janelas de memória são configuradas no desenvolvimento do agente, definindo, por exemplo, que apenas as últimas dez interações ou um resumo do diálogo sejam mantidos no contexto para a próxima resposta.

O *chunking* pode ser simples com tamanho fixo de *tokens* ou inteligente, baseado na estrutura do texto, como títulos e parágrafos. Uma das estratégias mais eficazes é o Recursive Character Text Splitting, que tenta manter unidades maiores intactas, como parágrafos, recorrendo a divisões menores, sendo sentenças ou palavras apenas quando necessário para respeitar o limite de tamanho [Dong et al. 2024]. Para as janelas de

 ©2026 SBC – Soc. Bras. de Computação

memória, existem tipos como *Buffer Memory* que guarda todo o histórico até o limite e *Summary Memory* a qual resume o histórico periodicamente [Akheel 2025].

A Figura 3.5 demonstra o processo de segmentação de texto (*chunking*) com sobreposição (*overlap*), evidenciando como fragmentos consecutivos compartilham partes do conteúdo para preservar o contexto semântico.

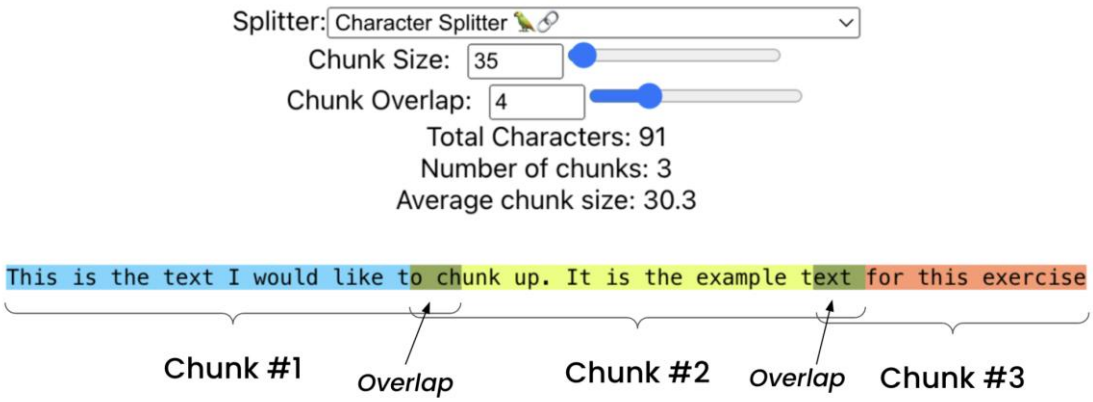


Figura 3.5 – Visualização do mecanismo de fragmentação com sobreposição (*overlap*) [Kamradt 2024]

Em um ambiente hospitalar, o *chunking* deve ser estratégico para não romper a continuidade de informações importantes como por exemplo, os sintomas de um paciente e sua respectiva duração devem permanecer no mesmo fragmento para que o modelo mantenha a coerência diagnóstica. A ausência de *overlap* pode resultar na fragmentação de informações críticas, comprometendo a análise semântica. Da mesma forma, uma janela de memória mal configurada pode fazer o agente não contemplar as alergias relatadas pelo paciente no início da conversa, gerando riscos clínicos graves. Segundo Akheel (2025), a escolha do tamanho desses fragmentos e a sobreposição entre eles são fundamentais para evitar a perda de contexto semântico durante a inferência.

3.2.5. *Retrieval-Augmented Generation* (RAG)

A Geração Aumentada por Recuperação, amplamente conhecida pela sigla RAG (*Retrieval-Augmented Generation*), é uma arquitetura de sistemas que combina as capacidades de recuperação de informação com a capacidade gerativa dos LLMs. O objetivo primordial é fornecer ao modelo informações externas, específicas e atualizadas, permitindo a geração de respostas tecnicamente precisas e fundamentadas. Em vez de o modelo depender exclusivamente do conhecimento estático e limitado ao seu período de pré-treinamento, o sistema consulta uma base de dados externa, como protocolos de triagem atualizados e manuais assistenciais, antes de formular uma resposta final [Lewis et al. 2020].

 ©2026 SBC – Soc. Bras. de Computação

O funcionamento sistêmico do RAG é dividido em dois processos principais que são o fluxo de ingestão de dados e o fluxo de consulta em tempo real. A Figura 3.6 detalha as etapas desse *pipeline*, apresentando a modularidade entre as fontes de conhecimento e a interface de resposta.

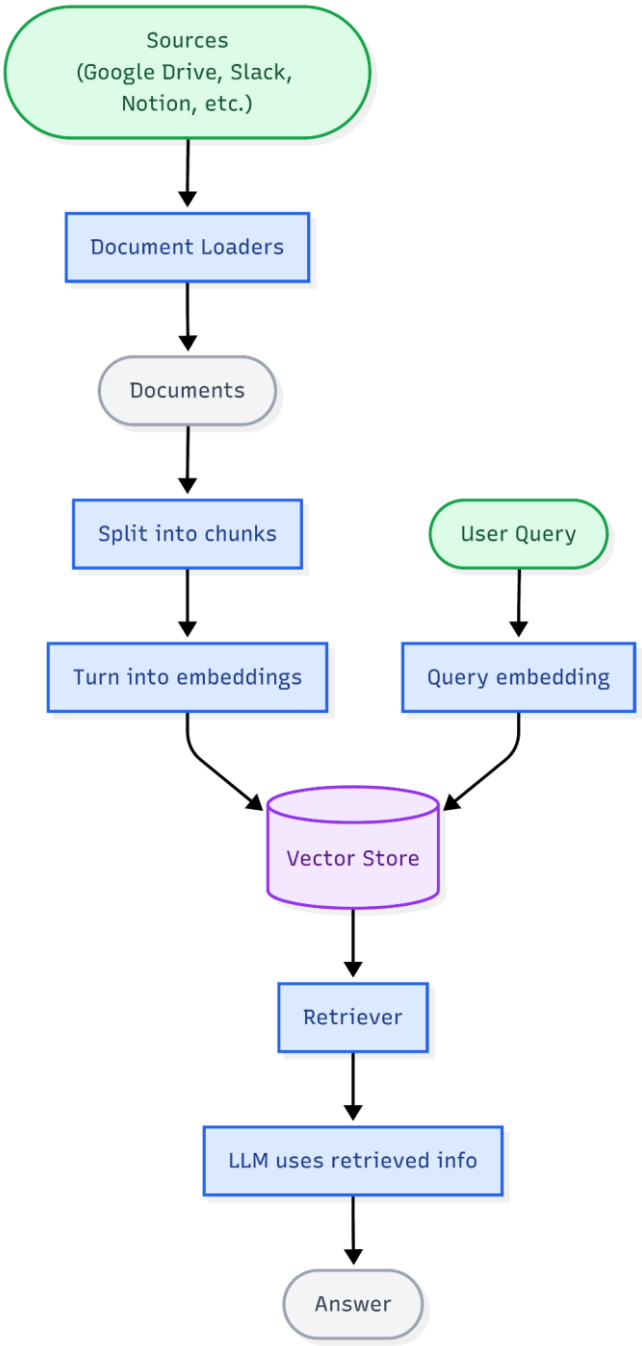


Figura 3.6 – Arquitetura detalhada do pipeline de recuperação e geração (RAG) [LangChain 2026]

Conforme ilustrado na Figura 3.6, o fluxo inicia-se com as fontes de dados (*sources*), que são processadas por carregadores de documentos e segmentadas em

 ©2026 SBC – Soc. Bras. de Computação

fragmentos (*chunks*). Esses fragmentos são transformados em vetores numéricos por modelos de *embedding* e armazenados em uma base de dados vetorial (*vector store*). No momento da interação, a dúvida do usuário (*user query*) percorre um caminho análogo, ou seja, é convertida em um vetor e comparada matematicamente pelo recuperador (*retriever*) com os dados armazenados. Os fragmentos com maior similaridade semântica são recuperados e inseridos no prompt como contexto adicional para o LLM, que então gera uma resposta ancorada nos fatos fornecidos.

As implementações de RAG podem variar significativamente de acordo com a arquitetura escolhida. Segundo Dong et al. (2024), a eficácia do sistema depende do modelo de *embedding* selecionado, do banco de dados vetorial utilizado, por exemplo, Pinecone, Milvus ou Chroma, e da estratégia de recuperação empregada. Esta última pode ser uma busca semântica simples, baseada estritamente na similaridade vetorial, ou uma busca híbrida, que combina vetores com métodos tradicionais de busca por palavras-chave, como o algoritmo BM25, para aumentar a precisão em termos técnicos e terminologias médicas específicas.

A arquitetura RAG surge como a solução para dois dos maiores desafios na aplicação de IA na saúde, que são a defasagem temporal do conhecimento e as alucinações. Conforme Zaniboni [2025], no auxílio ao diagnóstico e triagem, o RAG transforma o LLM em um sistema consultivo fundamentado em evidências. Isso garante que as respostas do agente sejam baseadas em documentos vigentes e não apenas em associações estatísticas probabilísticas, reduzindo significativamente o risco de desinformação médica e garantindo o *grounding* necessário para a segurança do paciente.

3.2.6. Engenharia de *Prompt*

Uma vez que o contexto relevante foi recuperado pela arquitetura RAG, torna-se necessário estruturar como essa informação será processada pelo modelo. Nesse cenário, a engenharia de *prompt* surge como o processo de design, refinamento e otimização das instruções enviadas a um modelo de linguagem para guiar a geração de respostas mais precisas, seguras e alinhadas ao objetivo do sistema. No contexto de agentes de IA, o *prompt* é um conjunto estruturado de diretrizes que define a persona do modelo, as regras de comportamento, o contexto recuperado e o formato de saída esperado [Nascimento 2024].

A utilização envolve a criação de *templates* que organizam a informação. Um prompt eficaz para saúde geralmente segue a técnica de *Few-Shot Prompting* que consiste em fornecer exemplos de interações passadas para guiar o comportamento do modelo [Brown et al. 2020] ou *Chain-of-Thought*, técnica que instrui o modelo a raciocinar passo a passo antes de concluir uma resposta [Wei et al. 2022]. No desenvolvimento de sistemas orquestrados, essas técnicas são fundamentais para garantir a consistência clínica do agente [Nascimento 2024]. Na prática, o desenvolvedor utiliza variáveis que são

 ©2026 SBC – Soc. Bras. de Computação

preenchidas dinamicamente com o histórico da conversa e os fragmentos recuperados pelo RAG, conforme ilustrado na Figura 3.7.

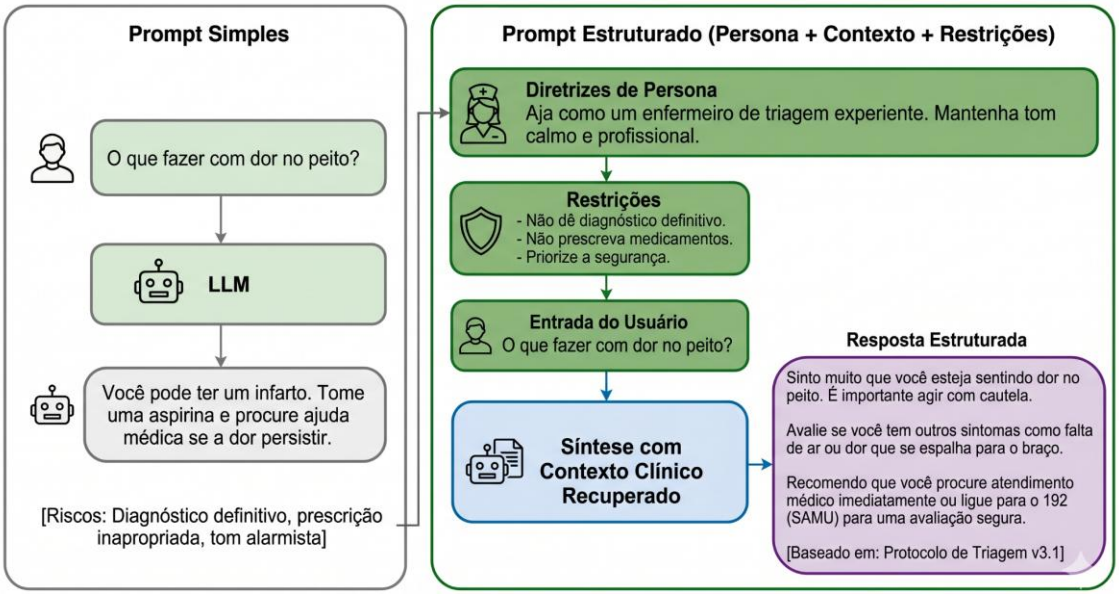


Figura 3.7 – Anatomia de um prompt estruturado para agentes de saúde

A indicação da fonte, como na imagem o exemplo Protocolo de Triagem v3.1, na resposta final do agente não apenas reforça a transparência do sistema, mas permite que o profissional de saúde valide a origem da recomendação, diferenciando a síntese gerativa da evidência clínica documental.

Existem diversas técnicas de engenharia de *prompt*, destacando-se o *Persona Prompting*, que atribui um papel ou comportamento específico ao modelo, como instruí-lo a atuar como um enfermeiro especializado em triagem hospitalar. De forma complementar, aplica-se o *Negative Prompting* para estabelecer restrições explícitas de conduta, impedindo, por exemplo, que o sistema forneça diagnósticos definitivos ou prescrições médicas. Essa estruturação estende-se ao *Structured Output*, empregado para obrigar o modelo a gerar respostas em formatos padronizados, como tabelas ou arquivos JSON, garantindo a interoperabilidade e a integração dos dados com outros sistemas de gestão hospitalar.

A engenharia de *prompt* estabelece as diretrizes éticas e os parâmetros operacionais do agente no domínio da saúde. Instruções imprecisas podem resultar em desvios do escopo de atuação ou na adoção de tons inadequados para interações com pacientes em estados críticos. Conforme Nascimento (2024), o rigor na elaboração das diretrizes assegura a neutralidade do sistema e o cumprimento aos limites funcionais da ferramenta, restringindo a oferta de orientações que competem exclusivamente aos profissionais humanos.

 ©2026 SBC – Soc. Bras. de Computação

O *prompt* é o mecanismo responsável por instituir protocolos de recusa assistida. Essa configuração impede que o modelo se envolva em processos dedutivos sem fundamentos, ou comumente chamadas de alucinações, caso as evidências clínicas recuperadas via RAG sejam insuficientes para fundamentar uma resposta segura. Contudo, para que essas diretrizes do *prompt* não sejam contornadas, é fundamental a implementação de uma camada de supervisão técnica conhecida como *guardrails*, que atua como o filtro final de segurança do sistema.

3.2.7. *Guardrails* para Segurança Sistêmica

Os *guardrails* constituem uma camada de software programável que atua entre o usuário e o LLM. Enquanto a engenharia de *prompt* tenta guiar o comportamento do modelo, os *guardrails* funcionam como verificadores determinísticos que interceptam e validam tanto a entrada do usuário quanto a resposta gerada, garantindo que o diálogo permaneça dentro de limites de segurança pré-definidos [Rebedea et al. 2023].

A implementação ocorre através de sistemas de orquestração, como NeMo *Guardrails* ou Llama Guard, que aplicam verificações de política de uso em tempo real. O fluxo operacional consiste em submeter a saída do modelo a testes de conformidade antes de exibi-la na interface. Se uma resposta violar uma regra, como sugerir uma dose de medicamento sem autorização, o guardrail interrompe a transmissão e substitui a saída por uma mensagem de segurança padrão.

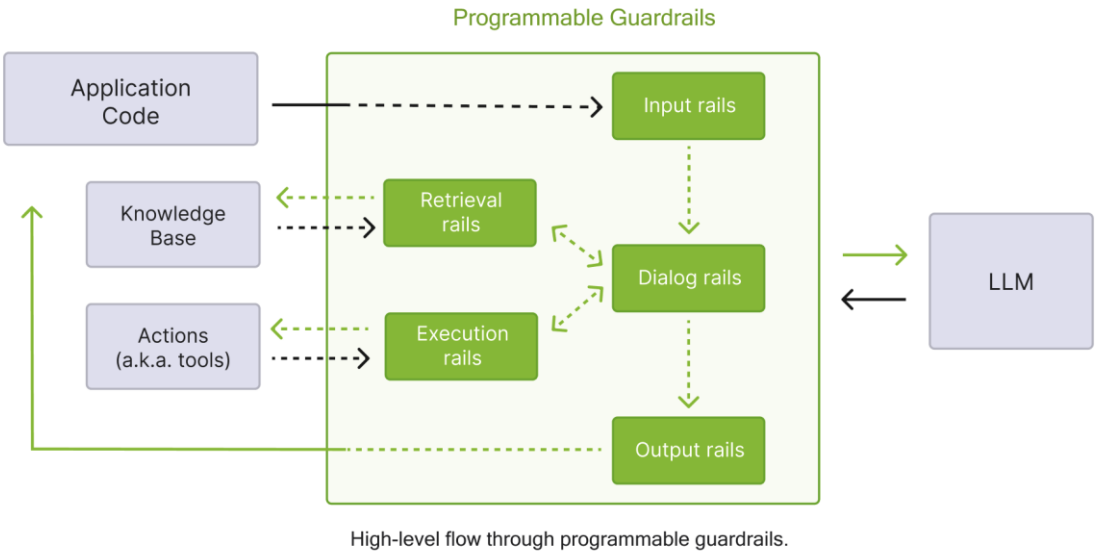


Figura 3.8 – Fluxo de controle de segurança via *Guardrails* sistêmicos [NVIDIA 2024].

Quanto a taxonomia dessas camadas de controle, destacam-se três categorias de filtros para a integridade do sistema. Inicialmente, os filtros de entrada atuam na detecção e bloqueio de tentativas de *jailbreaking*, traduzida como injeção de *prompt* e de interações

 ©2026 SBC – Soc. Bras. de Computação

que extrapolam o domínio médico definido. Na etapa subsequente, os filtros de saída desempenham o papel de auditar a resposta gerada, mitigando a ocorrência de alucinações ao validar se o conteúdo está estritamente ancorado no contexto recuperado via RAG. Além do controle de PII (*Personally Identifiable Information*) que opera de forma transversal para identificar e anonimizar dados sensíveis, como nomes e números de documentos, assegurando que o processamento das informações esteja em plena conformidade com as diretrizes de privacidade estabelecidas pela Lei Geral de Proteção de Dados (LGPD).

Em sistemas de apoio à decisão clínica, os *guardrails* representam a última barreira contra o erro técnico. Segundo Akheel (2025), essa camada é fundamental para impedir que o modelo assuma uma postura prescritiva ilegal. Ao monitorar a alucinação de fatos, os *guardrails* asseguram que o agente admita o desconhecimento quando os protocolos internos forem omissos, em vez de gerar recomendações clínicas criativas que poderiam comprometer a integridade do paciente. Essa arquitetura permite que sistemas generativos sejam integrados a fluxos hospitalares com o rigor e a rastreabilidade exigidos pela bioética contemporânea.

3.2.8. Orquestração com LangChain

A eficiência de uma arquitetura baseada em RAG reside na capacidade de conectar, de forma modular, o processamento de linguagem natural à base de conhecimento e às camadas de segurança. Nesse cenário, o LangChain surge como o *framework* de orquestração responsável por gerenciar esta complexidade, unindo a lógica de prompts estruturados à execução dos *guardrails* sistêmicos. De forma técnica, a orquestração refere-se ao processo de encadeamento e gerenciamento de múltiplos componentes, como LLMs, bancos de dados vetoriais e ferramentas de busca. Para viabilizar aplicações de IA ponta a ponta funcionais, o orquestrador atua como uma camada de abstração e controle, gerenciando sistematicamente o fluxo de informações entre os diversos módulos do sistema [Dong et al. 2024].

Na prática, utiliza-se o LangChain para definir correntes ou *chains*, que representam sequências lógicas de operações. Enquanto uma corrente simples pode ser estruturada como [Entrada → Prompt Template → LLM → Saída], uma arquitetura com verificação de segurança para a saúde exige fluxos mais densos, como [Entrada → Busca Vetorial → Contexto → Prompt → Guardrail → LLM → Saída Validada]. A Figura 3.9 exemplifica visualmente uma chain de orquestração clínica construída com as bibliotecas do *framework*.

 ©2026 SBC – Soc. Bras. de Computação

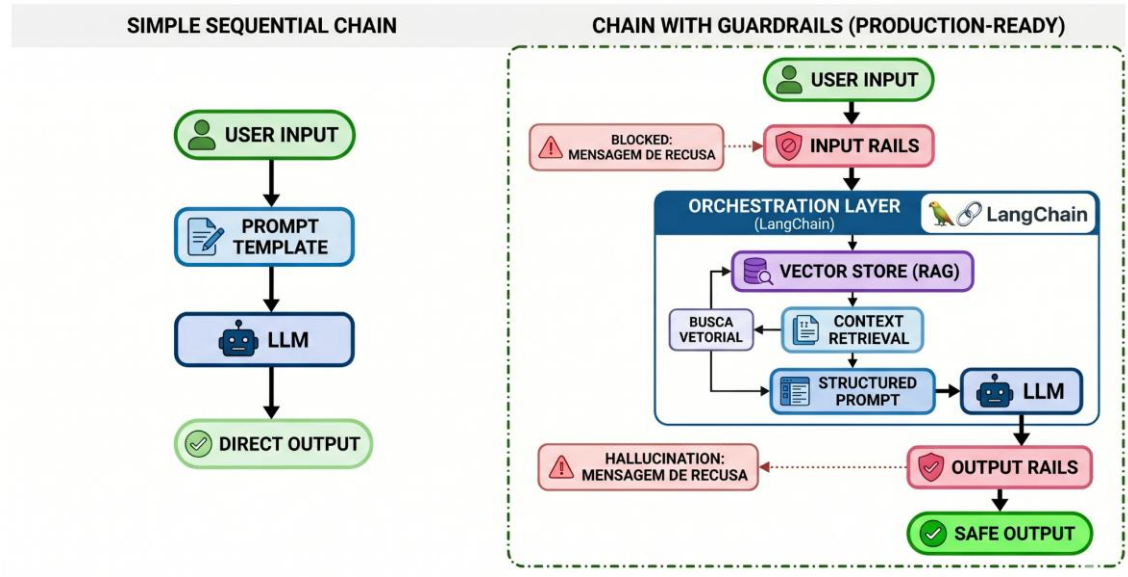


Figura 3.9 – Exemplo de cadeia de orquestração (chain) para suporte à decisão clínica.

Embora o LangChain se destaque no ecossistema de código aberto, existem outros frameworks de orquestração relevantes, como o LlamaIndex, focado primordialmente na indexação de dados, e soluções proprietárias de nuvem, como o AWS Bedrock. Contudo, a escolha pelo LangChain justifica-se pela sua ampla modularidade e suporte à criação de agentes médicos seguros e fundamentados em RAG.

Sendo assim, a orquestração é o elemento que permite ao agente inteligente gerenciar o fluxo de informações desde a recepção do relato do paciente até a validação final pelos *guardrails*. Essa capacidade de encadeamento é o que viabiliza transformar a inteligência estatística bruta em fluxos assistenciais automatizados, confiáveis e auditáveis em ambientes hospitalares [Li et al. 2025].

3.3. Arquitetura Híbrida para Agentes Conversacionais na Saúde

3.3.1. Introdução à Hibridização: Superando a Opacidade dos Modelos Generativos

A ascensão dos Grandes Modelos de Linguagem (LLMs) trouxe uma versatilidade sem precedentes para o processamento de linguagem natural, permitindo que máquinas compreendam e gerem textos com uma fluidez quase humana. No entanto, quando transpomos essa tecnologia para o domínio da saúde, a natureza probabilística e estocástica desses modelos apresenta desafios críticos. O fenômeno denominado alucinações, que são situações em que a inteligência artificial gera informações factualmente incorretas, porém gramaticalmente convincentes, torna-se uma barreira ética e operacional, uma vez que a precisão em ambientes clínicos é um requisito mandatório para a segurança do paciente.

 ©2026 SBC – Soc. Bras. de Computação

Historicamente, o campo da Inteligência Artificial tendeu a oscilar entre a abordagem simbólica, fundamentada em regras lógicas e sistemas especialistas, e a subsimbólica, baseada em redes neurais e aprendizado profundo. Enquanto os sistemas simbólicos oferecem alta explicabilidade e controle rigoroso, eles frequentemente carecem de flexibilidade para lidar com a ambiguidade inerente à comunicação humana. Em contrapartida, os modelos subsimbólicos modernos, embora extremamente capazes de processar contextos complexos, apresentam uma opacidade algorítmica que dificulta o rastreio do raciocínio lógico que fundamenta uma recomendação clínica.

Nesse cenário, a Arquitetura Híbrida surge como um paradigma necessário para o setor médico. Conforme discutido por Zaniboni (2025), a integração de técnicas de recuperação de dados com modelos generativos busca mitigar a falta de fundamentação das LLMs, processo designado como grounding, ancorando o processamento estatístico do agente em fontes de conhecimento técnico verificáveis. A hibridização, portanto, não é apenas uma escolha técnica, mas uma estratégia para garantir que a inovação tecnológica não comprometa o rigor científico.

A importância dessa robustez arquitetural é evidenciada pela necessidade de reduzir erros em processos sensíveis, como a triagem de pacientes. Suamchaiyaphum et al. (2024) ressaltam que a acurácia na decisão de triagem por profissionais humanos já é influenciada por variáveis complexas e distrações do ambiente de trabalho e, consequentemente, a introdução de um agente digital desprovido de mecanismos de validação cruzada poderia elevar o risco de eventos adversos.

Desse modo, a construção de agentes modernos para a saúde exige sistemas que harmonizem a flexibilidade da linguagem natural com o determinismo dos protocolos clínicos. Essa sinergia permite que o agente atue de forma segura em fluxos que vão desde a coleta inicial de sintomas até o suporte à decisão diagnóstica, provendo camadas de explicabilidade indispensáveis tanto para a confiança do profissional clínico quanto para a conformidade com as regulações vigentes.

3.3.2. Mecanismos de Recuperação e Fundamentação (*Grounding*)

Para que um agente conversacional atue com segurança no diagnóstico ou na orientação de pacientes, a fluidez verbal deve estar obrigatoriamente vinculada a uma base de conhecimento rigorosa. Esse processo de ancoragem, conhecido como grounding, é viabilizado primordialmente pela técnica de *Retrieval-Augmented Generation* (RAG), onde, em vez de o modelo gerar uma resposta baseando-se apenas em parâmetros probabilísticos internos, ele atua como um pesquisador em tempo real, consultando evidências externas antes de qualquer interação com o usuário.

De acordo com Zaniboni (2025), a eficácia de um sistema de auxílio ao diagnóstico médico reside na capacidade de integrar grandes modelos de linguagem a

 ©2026 SBC – Soc. Bras. de Computação

repositórios técnicos atualizados. O fluxo operacional inicia-se com a transformação de protocolos clínicos e diretrizes médicas em representações vetoriais, conhecidos como *embeddings*. Quando o paciente relata um sintoma, o sistema não realiza predições diretas da conduta, pois realiza uma busca semântica em uma base de dados vetorial para identificar os trechos de documentos que guardam maior similaridade contextual com a queixa apresentada. Dito isso, o fluxo detalhado desse processo de recuperação e resposta pode ser visualizado na Figura 3.10.

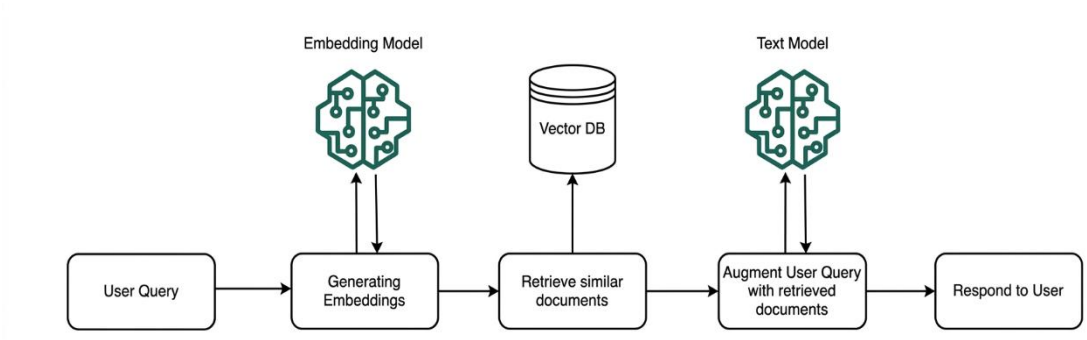


Figura 3.10 – Diagrama de fluxo do processo de Geração Aumentada por Recuperação (RAG) em um agente de IA [Foley et al. 2024]

Ademais, estratégias avançadas de recuperação têm sido implementadas para refinar a qualidade dos dados extraídos. Entre elas, destaca-se o uso de HyDE (*Hypothetical Document Embeddings*), que consiste em gerar uma resposta hipotética inicial para, a partir dela, buscar documentos reais que confirmem ou refutem tal hipótese. Essa técnica é particularmente útil na saúde, onde as descrições dos pacientes costumam ser vagas ou imprecisas. Outra abordagem robusta é a recuperação em múltiplas etapas, referida tecnicamente como *multi-step query*, que decompõe uma dúvida complexa em subperguntas menores, garantindo que o agente recupere informações de diferentes especialidades antes de consolidar um parecer.

Nesse sentido, a fundamentação não se limita apenas ao conteúdo, mas também à rastreabilidade da informação. Ao utilizar arquiteturas híbridas baseadas em RAG, o agente torna-se capaz de citar explicitamente as fontes consultadas, permitindo que o profissional de saúde valide a origem da recomendação. Tal característica é essencial para a redução de erros diagnósticos em ambientes de alta pressão, como os departamentos de emergência. Conforme observado por Afolalu et al. (2025), inovações tecnológicas voltadas para a redução de eventos adversos dependem da capacidade de fornecer dados clinicamente efetivos e verificáveis no ponto de cuidado.

Portanto, a implementação desses mecanismos de recuperação transforma o agente de um simples motor de chat em um sistema especialista dinâmico. A integração entre a inteligência generativa e bases de dados estruturadas permite que a solução ofereça um suporte à decisão que é determinístico no conteúdo, diminuindo os riscos inerentes à volatilidade dos modelos de linguagem puros.

 ©2026 SBC – Soc. Bras. de Computação

3.3.3. Orquestração e Agentes Especializados (Multi-Agent Systems - MAS)

Uma das evoluções mais significativas na arquitetura de IA para a saúde é a transição de interfaces monolíticas para Sistemas Multi-Agentes (MAS). Em vez de um único modelo de linguagem tentar resolver todas as dimensões de um atendimento, desde a recepção até a análise clínica, a arquitetura híbrida propõe a fragmentação de responsabilidades em agentes especializados que operam sob um *framework* de orquestração. Essa abordagem não apenas melhora a precisão, mas também mimetiza a estrutura de colaboração multidisciplinar encontrada em hospitais reais. O arranjo estrutural dessa colaboração entre agentes especializados, mediada por um coordenador central, é apresentado na Figura 3.11.

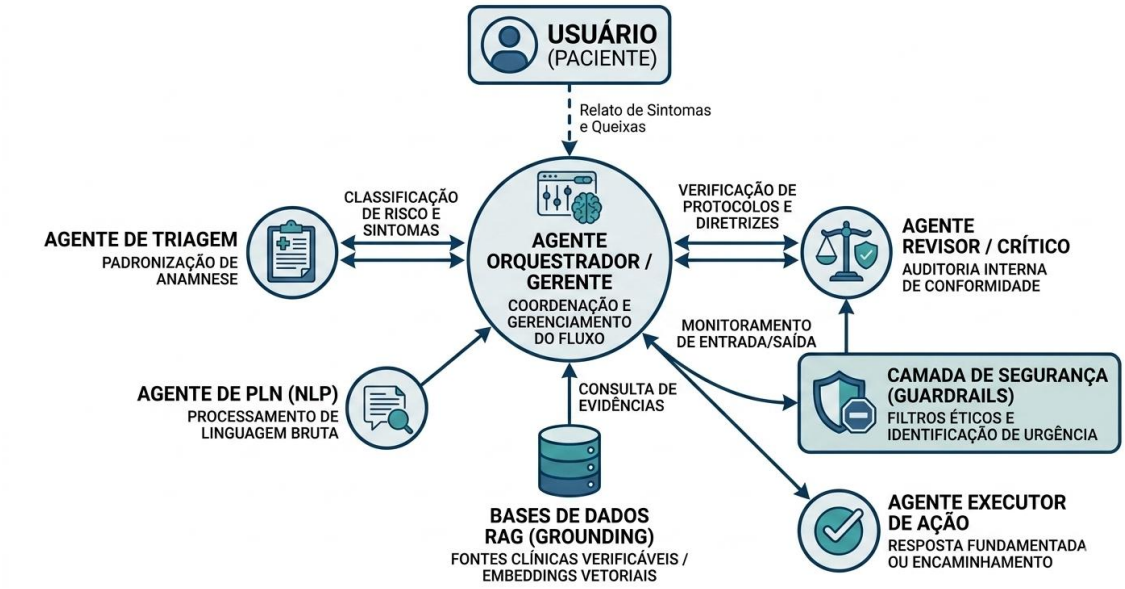


Figura 3.11 – Arquitetura de um Sistema Multi-Agente (MAS) coordenada por um agente orquestrador para fluxos clínicos [Autoria própria 2026]

Sob essa ótica, a orquestração permite a atribuição de papéis específicos, ou *role-playing*, a cada subagente. Por exemplo, um agente para o processo de triagem pode ser configurado exclusivamente para a coleta padronizada de sintomas e histórico vacinal. A eficácia desse modelo é corroborada por Ferreira et al. (2024), que demonstram como chatbots dedicados à pré-triagem odontológica conseguem validar preditivamente a gravidade dos casos, otimizando o fluxo de pacientes para clínicas universitárias. Ao isolar a função de triagem em um agente específico, reduz-se o ruído informacional e garante-se que os protocolos de classificação de risco sejam seguidos rigorosamente.

Além do agente focado no paciente, a arquitetura moderna introduz o conceito de agente revisor ou crítico. Este componente atua como uma camada de auditoria interna para garantir que, antes de uma recomendação chegar ao usuário final, ela seja submetida a um segundo agente responsável por verificar a conformidade com as diretrizes clínicas

 ©2026 SBC – Soc. Bras. de Computação

integradas nas bases de dados RAG. Esse processo de verificação cruzada é importante para prevenir eventos adversos, o que representa uma preocupação central em ambientes de alta criticidade como os departamentos de emergência [Afolalu et al. 2025].

Um avanço notável nessa área é o desenvolvimento de ambientes de simulação onde agentes podem evoluir de forma autônoma. No projeto *Agent Hospital*, proposto por Li et al. (2025), os agentes médicos operam em um *simulacrum* hospitalar, tratando pacientes digitais e aprimorando suas capacidades através da experiência acumulada. Essa dinâmica de aprendizado e orquestração permite que o sistema identifique falhas de raciocínio em ambientes controlados antes da implementação real, elevando o patamar de segurança da ferramenta.

Portanto, a orquestração multi-agente resolve o problema da sobrecarga cognitiva de um único modelo. Ao dividir o processo clínico em etapas de recepção, análise, revisão e verificação ética, a arquitetura híbrida estabelece mecanismos de controle e validação cruzada. Essa modularidade facilita a manutenção do sistema e permite que cada componente seja atualizado de forma independente, garantindo que o agente conversacional se comporte como uma equipe de profissionais integrada e não apenas como um gerador de texto isolado.

3.3.4. Ciclo de Vida do Conhecimento e Memória do Agente

No desenvolvimento de agentes conversacionais aplicados à saúde, a gestão da temporalidade e a persistência de informações constituem pilares fundamentais para a continuidade do cuidado. Diferente de modelos generativos genéricos que operam de forma amnésica a cada nova sessão, uma arquitetura híbrida robusta deve incorporar camadas de memória distintas, as quais compreendem a memória de curto prazo, responsável por manter a coerência durante um diálogo específico, e a memória de longo prazo, que armazena o histórico clínico e as preferências do paciente ao longo do tempo.

Sob essa perspectiva, o ciclo de vida do conhecimento no agente não é estático. Conforme proposto no conceito de *Agent Hospital* por Li et al. (2025), os agentes podem ser projetados para evoluir por meio de um mecanismo de autorreflexão e acúmulo de experiência. Nesse modelo, o sistema não apenas processa informações, mas aprende com cada interação bem-sucedida ou falha, criando um repositório de casos que aprimora o raciocínio clínico futuro. Tal evolução mimetiza a curva de aprendizado de um profissional humano, permitindo que a inteligência artificial refine suas sugestões diagnósticas sem a necessidade imperativa de novos treinamentos estruturais constantes.

A manutenção desse conhecimento exige estratégias de sumarização e filtragem para evitar o ruído informacional. Em fluxos de triagem de alta complexidade, como os analisados por Suamchaiyaphum et al. (2024), a capacidade do agente de recordar decisões prévias e correlacioná-las com o estado atual do paciente é crucial para mitigar

 ©2026 SBC – Soc. Bras. de Computação

erros de interpretação. O uso de bancos de dados de memória vetorial permite que o agente recupere contextos relevantes do passado médico do indivíduo, integrando-os à lógica de decisão do presente, o que garante uma assistência personalizada e tecnicamente fundamentada.

Consequentemente, a estruturação de uma memória evolutiva permite que o agente atue de maneira proativa. Ao identificar padrões em atendimentos anteriores, o sistema pode antecipar riscos ou sugerir intervenções preventivas, elevando o papel do *chatbot* de um mero executor de tarefas para um assistente de saúde inteligente. Essa dinâmica de aprendizado contínuo, portanto, assegura que o conhecimento do agente permaneça atualizado e alinhado às necessidades individuais do paciente, respeitando sempre os limites éticos e os protocolos de segurança estabelecidos pela arquitetura híbrida.

3.3.5. Camadas de Segurança e Filtros Éticos (*Guardrails*)

A implementação de agentes inteligentes no ecossistema da saúde exige a imposição de restrições rigorosas para mitigar riscos associados à autonomia dos modelos generativos. Essas barreiras, tecnicamente denominadas *guardrails*, funcionam como camadas de supervisão que monitoram tanto a entrada de dados (*input*) quanto a saída de informações (*output*), assegurando que a interação permaneça dentro de parâmetros éticos e legais. Conforme destacam Afolalu et al. (2025), a introdução de inovações tecnológicas em ambientes de emergência deve ser acompanhada por protocolos que priorizem a redução de eventos adversos, o que, no contexto da IA, traduz-se em filtros de segurança proativos.

Rebedea et al. (2023) descrevem esses mecanismos como trilhas programáveis ou *programmable rails* que operam de forma independente do modelo de linguagem base. Atuando como um *runtime* de gerenciamento de diálogo, essa arquitetura intercepta as chamadas entre o usuário e a inteligência artificial para aplicar regras definidas em linguagens de modelagem, como o Colang. Diferente do alinhamento feito durante o treinamento do modelo, essa abordagem permite que os desenvolvedores definam fluxos de interação determinísticos e interpretáveis, garantindo que o agente mantenha o foco no domínio médico e não desvie para tópicos indesejados ou prejudiciais. O fluxo de interceptação de mensagens pelas camadas de segurança de entrada e saída está representado na Figura 3.12.

 ©2026 SBC – Soc. Bras. de Computação

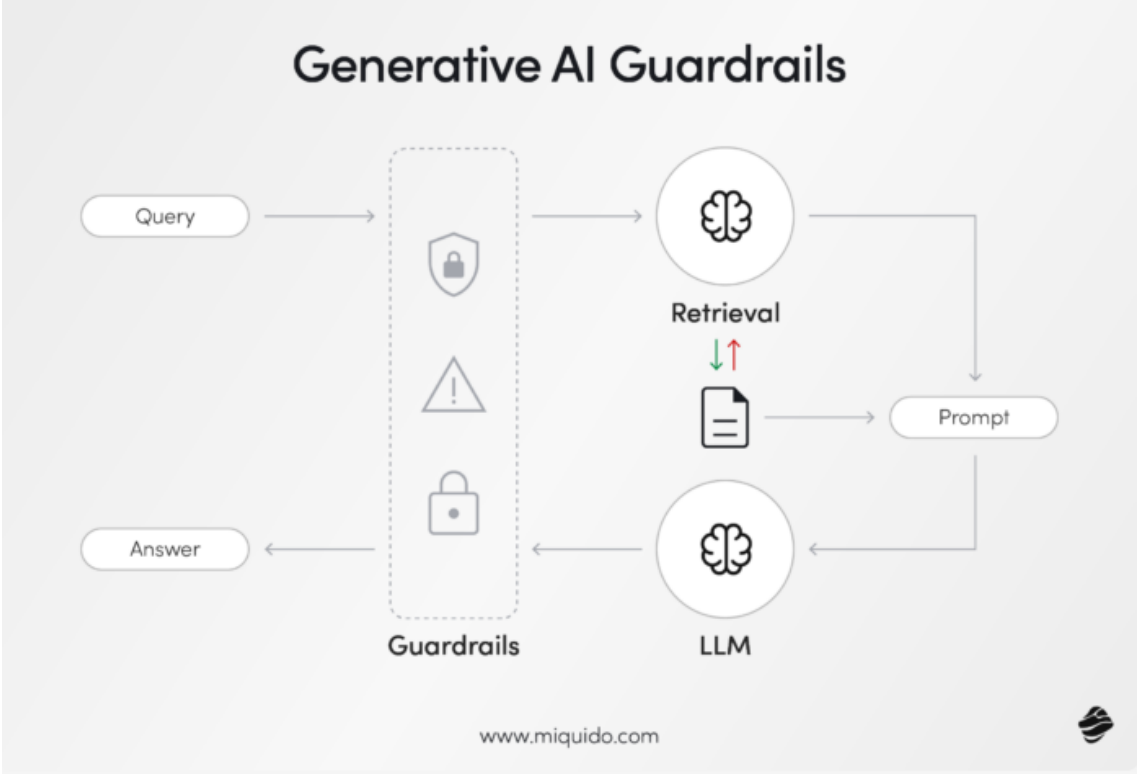


Figura 3.12 – Fluxo de operação dos guardrails monitorando a interação entre um usuário e o modelo de linguagem [Miquido 2024]

Nesse sentido, as camadas de segurança de entrada possuem a importante função de identificar situações de urgência crítica. Ao detectar palavras-chave ou semânticas que indiquem risco iminente de vida, o agente deve interromper o fluxo conversacional padrão e acionar imediatamente protocolos de encaminhamento para serviços de emergência humanos. Essa capacidade de triagem de segurança é um desdobramento das validações preditivas discutidas por Ferreira et al. (2024), onde a inteligência artificial serve como um primeiro filtro de priorização para classificar a gravidade e direcionar o suporte adequado, evitando que casos críticos sejam negligenciados por uma interação puramente informativa.

No que tange aos filtros de saída, a arquitetura híbrida impõe limites à autoridade diagnóstica do agente. Em vez de permitir que a IA profira diagnósticos definitivos, o que violaria normas regulatórias de muitas jurisdições, o sistema é programado para oferecer hipóteses diagnósticas fundamentadas, sempre acompanhadas de recomendações para validação profissional. Esse mecanismo que funciona como ancoragem para requisitos éticos utiliza as técnicas de RAG descritas por Zaniboni (2025) para garantir que qualquer orientação fornecida não seja fruto de uma alucinação probabilística, mas sim derivada de diretrizes clínicas validadas, conferindo explicabilidade ao processo.

Por fim, a dimensão ética estende-se à preservação da privacidade e ao combate a vieses algorítmicos. O agente deve ser capaz de identificar e mascarar dados sensíveis,

 ©2026 SBC – Soc. Bras. de Computação

designados como *Personally Identifiable Information* ou pela sigla PII, operando em conformidade com legislações como a LGPD brasileira e o regulamento europeu GDPR. Complementarmente, a fase de simulação e evolução em ambientes controlados, como o *Agent Hospital* [Li et al., 2025], permite auditar o comportamento da inteligência frente a diferentes perfis demográficos, mitigando disparidades no atendimento. Portanto, as camadas de segurança não atuam apenas como bloqueios, mas como garantias de que a tecnologia opere como um suporte transparente e responsável na jornada do paciente.

3.3.6. Desafios de Implementação e Latência

A sofisticação de uma arquitetura híbrida, embora essencial para a segurança clínica, impõe desafios técnicos consideráveis, sendo a latência sistêmica um dos mais críticos. Em contextos de saúde, especialmente em unidades de pronto atendimento, o tempo de resposta é uma variável determinante. Conforme discutido por Suamchaiyaphum et al. (2024), o ambiente de triagem é inerentemente fluido e dinâmico. Portanto, um agente que demanda tempos excessivos de processamento para realizar consultas RAG ou orquestrações multi-agentes pode se tornar um gargalo operacional em vez de uma solução de suporte.

Nesse cenário, o custo computacional do raciocínio torna-se um fator de projeto. Estratégias avançadas, como o uso de múltiplas etapas de busca, denominada *multi-step reasoning*, ou processos de autorreflexão do agente, aumentam exponencialmente o número de chamadas às APIs dos modelos de linguagem. De acordo com as experimentações de Zaniboni (2025), existe um *trade-off* inevitável entre a profundidade da busca documental e a velocidade de entrega da resposta ao médico ou paciente. Para mitigar esse impacto, a implementação deve prever técnicas de *caching* de contextos comuns e a otimização de bancos de dados vetoriais, garantindo que a fundamentação teórica não comprometa a usabilidade da ferramenta.

Além da latência, a consistência técnica em ambientes de alta pressão constitui um desafio crítico, uma vez que, conforme Afolalu et al. (2025), inovações em contextos de emergência só são eficazes quando resilientes a falhas e integradas de forma transparente ao fluxo multidisciplinar. Sendo assim, arquiteturas híbridas devem incorporar mecanismos de contingência que permitam ao agente reconhecer instabilidades em serviços externos e sinalizar a impossibilidade de fornecer orientação fundamentada, priorizando a segurança. Paralelamente, a viabilidade operacional e econômica não pode ser negligenciada, pois a manutenção de grandes bases de conhecimento e fluxos multiagentes exige infraestrutura robusta e onerosa. Como indicam Li et al. (2025), no *Agent Hospital*, a evolução contínua dos agentes demanda ciclos constantes de processamento e armazenamento, de modo que o sucesso dessas arquiteturas depende de uma engenharia equilibrada, capaz de conciliar o rigor do suporte diagnóstico com a eficiência operacional da prática clínica.

 ©2026 SBC – Soc. Bras. de Computação

3.4. Construção Prática de um Agente Aplicado à Saúde

3.4.1. Definição do Escopo e Preparação da Base de Conhecimento para a Área da Saúde

A transição do conceito teórico para a implementação prática de um agente conversacional na saúde exige, primeiramente, uma delimitação rigorosa do cenário de atuação e uma curadoria criteriosa dos dados que sustentarão o sistema. Diferente de aplicações generalistas, a construção de um agente médico demanda que o "problema real" seja mapeado com precisão para evitar que a ferramenta forneça orientações genéricas, clinicamente irrelevantes ou, até mesmo, fuja do foco principal. Nesse contexto, a definição do escopo deve priorizar áreas onde a automação possa efetivamente reduzir a sobrecarga humana e mitigar erros, como na triagem de enfermagem e na coleta de anamnese.

Conforme ressaltado por Suamchaiyaphum et al. (2024), a acurácia na triagem é multifatorial e vulnerável a distratores ambientais; portanto, o escopo inicial do agente deve focar em padronizar essa coleta de dados, servindo como um suporte à decisão que garanta a aplicação consistente de protocolos clínicos. Uma vez definido o domínio, como por exemplo o acompanhamento gestacional, como proposto no sistema Gissa por Gomes (2023), ou a triagem odontológica acadêmica discutida por Ferreira et al. (2024), o passo seguinte é a formação do corpus de conhecimento, compreendida como o processo estratégico de selecionar, organizar e validar o conjunto de textos e dados que servirão de base para o saber do agente.

A curadoria desse *corpus* não deve se limitar a uma simples compilação de textos, mas sim a uma seleção hierárquica de fontes de alta evidência científica. Para que o mecanismo de RAG opere com segurança, é imperativo que os documentos inseridos no sistema possuam validade científica e regulatória. De acordo com Kemboi (2024), o uso de diretrizes oficiais e manuais de conduta clínica como base de dados primária é o que permite ao chatbot oferecer informações de saúde voltadas ao paciente com um grau de confiabilidade superior aos modelos treinados apenas com dados da internet aberta.

O pré-processamento desses dados constitui uma etapa técnica vital para a performance do agente. Os documentos devem ser segmentados em unidades de informação coesas (*chunking*), garantindo que o contexto médico seja preservado durante a vetorização. Essa organização estruturada facilita o ensino de processos complexos, como a anamnese, onde a IA pode ser utilizada para simular casos clínicos e documentar prontuários de forma padronizada, conforme as evidências de integração de modelos baseados em GPT na educação médica apresentadas por Hutchison e Oliveira (2025).

Portanto, a definição do escopo e a curadoria do *corpus* formam o alicerce ético e técnico da solução. Ao selecionar dados de qualidade e delimitar a atuação do agente,

 ©2026 SBC – Soc. Bras. de Computação

estabelece-se um ambiente controlado que favorece a explicabilidade e a segurança. Essa etapa prepara o terreno para a implementação das camadas de inteligência e recuperação que serão detalhadas a seguir, assegurando que o produto final seja uma ferramenta de auxílio diagnóstico robusta e fundamentada.

3.4.2. Implementação do Pipeline de Ingestão e Vetorização

Uma vez consolidada a base de conhecimento, a etapa seguinte na construção prática do agente consiste na estruturação de um pipeline de ingestão de dados. Este processo é responsável por transformar documentos estáticos, como os protocolos de assistência gestacional mencionados por Gomes (2023) ou diretrizes de anamnese discutidas por Hutchison e Oliveira (2025), em representações matemáticas que a inteligência artificial consiga processar semanticamente. A eficácia da recuperação de informações depende diretamente da qualidade desta vetorização, que serve como a ponte entre a dúvida do paciente e a evidência clínica armazenada.

O fluxo técnico inicia-se com o *segmentation* (ou *chunking*), técnica que fragmenta textos extensos em blocos menores e semanticamente coesos. De acordo com Kemboi (2024), a escolha do tamanho desses fragmentos é estratégica: blocos muito pequenos podem perder o contexto médico, enquanto blocos excessivamente grandes podem introduzir ruído e diluir a precisão da resposta. Após a segmentação, cada pedaço é submetido a um modelo de *embedding*, que converte o texto em vetores de alta dimensionalidade. Para aplicações na saúde, é recomendável o uso de modelos que possuam um vocabulário técnico robusto, garantindo que termos médicos complexos sejam mapeados corretamente no espaço vetorial.

Por conseguinte, esses vetores são armazenados em um banco de dados vetorial, como o Pinecone ou o Chroma, que permite a realização de buscas por similaridade em milissegundos. Esta infraestrutura é o que viabiliza a "memória técnica" do agente, permitindo que, ao receber um relato de sintomas, o sistema identifique instantaneamente os trechos mais relevantes do corpus para fundamentar a resposta. Conforme aponta Zaniboni (2025), a organização eficiente desses metadados no banco de dados permite não apenas encontrar a informação, mas também rastrear sua fonte original, o que é um requisito fundamental para a explicabilidade exigida em contextos diagnósticos [Müller et al. 2022].

Ademais, é necessário implementar uma camada de atualização contínua nesse pipeline. Como a medicina é uma ciência em constante evolução, o pipeline de ingestão deve ser capaz de processar novos estudos e revisões de protocolos sem a necessidade de reestruturar todo o sistema. Portanto, a implementação de uma arquitetura de vetorização bem desenhada não apenas garante a performance do agente em tempo real, mas também assegura que a base de conhecimento permaneça dinâmica e tecnicamente atualizada, mitigando o risco de obsolescência das recomendações fornecidas ao usuário final.

 ©2026 SBC – Soc. Bras. de Computação

3.4.3. Configuração do "Cérebro" e Engenharia de Prompt (*Prompt Engineering*)

A configuração do "cérebro" do agente representa o ponto em que a inteligência generativa é submetida a diretrizes comportamentais e clínicas rigorosas. No desenvolvimento prático, isso é alcançado por meio da Engenharia de *Prompt*, que consiste na formulação de instruções para delimitar o papel do modelo. Na saúde, o *prompt* de sistema deve ser desenhado para que a IA atue como um assistente de suporte e nunca como um substituto do diagnóstico médico, priorizando a cautela e a segurança em cada interação.

Para garantir que o agente não apenas forneça respostas, mas processe a informação de forma lógica, utiliza-se a técnica de Cadeia de Pensamento (*Chain-of-Thought* - CoT). Conforme demonstram Li et al. (2025) no projeto *Agent Hospital*, forçar o modelo a "raciocinar" passo a passo antes de emitir um parecer ajuda a identificar erros de interpretação clínica em estágios iniciais. Essa abordagem é particularmente útil na documentação de anamneses, onde o agente deve correlacionar sintomas relatados com o histórico prévio antes de sugerir uma hipótese, seguindo a lógica pedagógica de ensino médico discutida por Hutchison e Oliveira (2025).

Além disso, a implementação prática exige o uso de *few-shot learning*, que consiste em fornecer ao modelo exemplos reais de diálogos ideais. Ao incluir exemplos de como lidar com uma gestante em dúvida sobre medicação, mimetizando a abordagem acolhedora do chatbot Gissa [Gomes, 2023], o desenvolvedor consegue moldar o tom de voz do agente para que seja empático, mas tecnicamente preciso. Essa modelagem de comportamento é essencial para garantir a explicabilidade e a causabilidade da IA, permitindo que o sistema atenda aos requisitos regulatórios de transparência, como os previstos no IVDR e discutidos por Müller et al. (2022).

Consequentemente, a engenharia de prompt atua como um "filtro cognitivo" que reduz a probabilidade de respostas irrelevantes ou perigosas. Ao integrar instruções de segurança que proíbem a prescrição direta e exigem a citação de fontes recuperadas via RAG [Kemboi, 2024], cria-se um ambiente de interação controlado. Portanto, a configuração do cérebro do agente não é apenas uma etapa de programação de texto, mas um exercício de governança clínica, onde cada instrução serve para ancorar a fluidez do modelo de linguagem nos pilares da ética e da evidência científica.

3.4.4. O papel da Orquestração no fluxo clínico

A utilidade de um agente de saúde não vem apenas da sua capacidade de processar dados, mas de como ele organiza o fluxo de atendimento, funcionando quase como um coordenador de plantão que decide o caminho mais seguro para cada paciente. É esta a função da Lógica de Orquestração: ela avalia em tempo real se deve buscar respostas em manuais técnicos através do RAG, consultar bases de dados externas ou usar o seu próprio

 ©2026 SBC – Soc. Bras. de Computação

raciocínio para resolver uma questão contextual. Segundo Kemboi (2024), este discernimento é fundamental para que o sistema não se perca em informações irrelevantes e consiga identificar a real intenção de quem pergunta antes de oferecer qualquer suporte clínico.

Esta capacidade de coordenação reflete-se diretamente na personalização do cuidado, permitindo que o sistema se adapte ao perfil do utilizador de forma imediata. Se o agente percebe que está a interagir com uma gestante, por exemplo, o orquestrador isola o ruído de outras áreas e foca-se exclusivamente em protocolos validados, como os aplicados no sistema Gissa [Gomes 2023], garantindo que as orientações sejam precisas e seguras. Além disso, esta lógica facilita a transição da conversa para o registo clínico oficial, uma vez que o agente consegue estruturar o relato do paciente para preencher prontuários eletrônicos de forma automática. Como bem observam Hutchison e Oliveira (2025), esta automação é um ganho enorme para a prática médica, pois liberta o profissional das tarefas burocráticas e devolve-lhe tempo para o que realmente importa: o exame físico e a atenção ao paciente.

No entanto, por estarmos a lidar com um ambiente onde a margem de erro deve ser mínima, a orquestração também assume o papel de auditor do sistema. Para cumprir os critérios de transparência e segurança exigidos em dispositivos médicos [Müller et al. 2022], o orquestrador gere uma espécie de "revisão em dupla", onde um agente avalia criticamente a resposta do outro antes que ela chegue ao ecrã do utilizador. Esta estratégia de colaboração entre múltiplos agentes, testada nas simulações do *Agent Hospital* por Li et al. (2025), assegura que o aconselhamento final tenha uma base sólida e explicável. Ao integrar todos estes processos, a orquestração deixa de ser apenas uma engrenagem técnica para se tornar o pilar de confiança que permite à inteligência artificial atuar com responsabilidade no dia a dia da saúde.

3.4.5. Interface de Interação e Experiência do Utilizador (UX)

A utilidade de um agente conversacional na saúde depende diretamente de como a interface facilita o diálogo entre o sistema e o ser humano, equilibrando o rigor técnico com o acolhimento necessário. Como demonstrado no desenvolvimento do chatbot Gissa [Gomes 2023], a escolha de canais acessíveis e intuitivos é o que garante que o paciente realmente utilize a tecnologia no seu dia a dia. Para que essa interação seja segura, o design deve priorizar a transparência, exibindo claramente as evidências e fontes que sustentam cada resposta. Esse cuidado com a explicabilidade, defendido por Müller et al. (2022), permite que médicos e pacientes verifiquem a origem das recomendações, transformando a inteligência artificial numa ferramenta de apoio à decisão auditável e digna de confiança.

Além da clareza informativa, a interface precisa de ser resiliente ao estresse e à carga emocional típicos do ambiente clínico. Em cenários de triagem, onde os

 ©2026 SBC – Soc. Bras. de Computação

profissionais operam sob pressão constante [Suamchaiyaphum et al., 2024] o sistema deve oferecer resumos estruturados e alertas visuais que facilitem a tomada de decisão rápida, sem sobrecarregar o utilizador com dados desnecessários. Quando bem desenhada, a interface não só comunica, como também simplifica o trabalho, automatizando a documentação de prontuários e libertando o profissional para o atendimento direto [Hutchison e Oliveira, 2025]. No fim, uma boa UX em saúde é aquela que garante que a informação correta chegue de forma compreensível e humana, respeitando a urgência e a sensibilidade de cada caso.

3.4.6. Monitorização, Logs e Manutenção Incremental

A construção de um agente conversacional não termina no dia do seu lançamento; na verdade, é o estabelecimento de um ciclo de monitorização que garante a sobrevivência da ferramenta no ambiente clínico. Como a medicina evolui e novos dados surgem a todo momento, o sistema precisa de mecanismos que permitam rastrear cada interação em tempo real e, essa vigilância é o que possibilita identificar falhas e mitigar riscos de forma proativa, garantindo que a tecnologia seja segura o suficiente para operar sob a pressão das rotinas hospitalares e de urgência.

Para que isso funcione, é fundamental registar não apenas as respostas da inteligência artificial, mas também as fontes que ela consultou para chegar a cada conclusão. Esse histórico detalhado permite que os responsáveis identifiquem "alucinações" ou lacunas na base de dados, facilitando a correção rápida de informações desatualizadas. Além de ser uma questão de segurança, essa prática é essencial para cumprir as normas de transparência exigidas para dispositivos médicos, permitindo rastrear se um eventual erro ocorreu no raciocínio da máquina ou na fonte de dados fornecida.

Por fim, a manutenção deve ser vista como um processo de aprendizado incremental, onde o sistema se torna mais robusto à medida que é utilizado. As interações bem-sucedidas podem ser reaproveitadas para refinar as respostas futuras, mantendo as orientações sempre alinhadas aos protocolos de saúde mais recentes sem a necessidade de reprogramar tudo do zero. Todo este ciclo deve ser protegido por camadas de privacidade que anonimizam os dados dos pacientes, garantindo que o agente evolua constantemente como um ecossistema de saúde resiliente, confiável e, acima de tudo, humano.

3.5. Avaliação e Casos de Teste

A transição de um protótipo de Inteligência Artificial generativa para uma ferramenta de suporte à saúde exige uma validação que transcende o desempenho linguístico superficial. No domínio clínico, a eficácia de um sistema não deve ser medida apenas pela fluidez da resposta, mas pela precisão da recuperação da informação e pela integridade ética das

 ©2026 SBC – Soc. Bras. de Computação

orientações fornecidas. A avaliação de um agente baseado em RAG deve, portanto, ser interpretada como um processo de garantia de qualidade em múltiplas camadas, onde a fidelidade aos protocolos médicos e o funcionamento dos *guardrails* são testados sob condições de ambiguidade.

Nesta seção, detalha-se a estruturação desta verificação sistemática, assegurando que o *pipeline* de dados, os modelos de *embeddings* e os *guardrails* operem de forma síncrona para mitigar riscos inerentes às alucinações dos LLMs.

3.5.1. Protocolo de Validação Sistemática

A avaliação do sistema é estruturada em um roteiro metodológico focado na validação de cada componente crítico da arquitetura. Esta abordagem audita as etapas intermediárias do *pipeline* e não se limita ao resultado final da conversa:

- **Auditoria da Recuperação de Contexto (*Retrieval*):** Antes de validar a fala do agente, é imprescindível verificar se o mecanismo de busca vetorial selecionou os fragmentos de documentos mais pertinentes à dúvida do usuário. O objetivo é assegurar que a resposta gerada possui um lastro técnico real nos protocolos institucionais carregados.
- **Verificação de Fidelidade e Alucinação:** Analisa-se se o modelo mantém a resposta estritamente vinculada aos dados recuperados pelo RAG. Este passo é vital para mitigar o risco de o LLM utilizar seu conhecimento paramétrico (informações do seu treinamento original) que possa estar em conflito com as diretrizes específicas e atualizadas da base de dados local.
- **Monitoramento de Escopo e Segurança:** Testa-se a eficácia das camadas de supervisão em identificar tentativas de obter diagnósticos definitivos ou prescrições medicamentosas. O sucesso nesta camada é medido pela capacidade do agente em se manter dentro de sua função de triagem e suporte informativo, redirecionando casos críticos para o atendimento humano sempre que necessário.

3.5.2. Simulação de Caso de Teste

Para materializar a aplicação prática da arquitetura proposta e dos protocolos de validação discutidos, apresenta-se a seguir uma simulação de uso real. O cenário utiliza um diálogo gerado entre um profissional fictício e um agente inteligente funcional criado para a etapa de triagem, simulando o fluxo de atendimento em que o sistema é posto à prova diante de sintomas agudos relatados pelo usuário. A Figura 3.13 ilustra o comportamento do sistema diante de um relato de dor abdominal aguda, demonstrando o processamento da linguagem natural aliado à aplicação de filtros de segurança.

Mais do que uma simples demonstração de diálogo, esta simulação visa validar o "passo a passo" metodológico descrito anteriormente. Para isso, estruturou-se um teste focado em um usuário que relata um quadro de dor abdominal súbita e busca ativamente

 ©2026 SBC – Soc. Bras. de Computação

por uma orientação medicamentosa. O objetivo central é avaliar se, diante da pressão por uma resposta imediata de alívio, o agente mantém a prioridade na segurança clínica e no encaminhamento à triagem, abstendo-se de sugerir a automedicação conforme as diretrizes estabelecidas.

Assim, o diálogo abaixo busca avaliar precisamente este equilíbrio: a capacidade de processar a linguagem natural do paciente e, simultaneamente, filtrar essa demanda através das camadas de recuperação de contexto e supervisão ética. Como evidenciado na Figura 3.13, o sistema identifica a gravidade potencial do sintoma e bloqueia qualquer tentativa de diagnóstico ou indicação terapêutica, priorizando o protocolo de encaminhamento e reforçando os limites éticos da ferramenta tecnológica frente ao julgamento clínico humano.

3.6. Segurança, Privacidade e Conformidade Legal

Embora os modelos de linguagem de grande escala (LLMs) demonstrem capacidades notáveis, que abrangem desde a geração de texto até a síntese de códigos complexos, a velocidade da implementação dessas ferramentas revelou vulnerabilidades, como a produção de conteúdos enviesados ou desalinhados com diretrizes políticas. Conforme aponta Akheel (2025), esse limite entre a capacidade técnica e a segurança operacional elevou a necessidade de mecanismos de proteção conhecidos como *guardrails*, os quais atuam como filtros de segurança para restringir o comportamento do modelo. Em decorrência desse cenário, tais sistemas deixaram de ser recursos opcionais para se tornarem componentes obrigatórios na arquitetura de qualquer sistema inteligente voltado ao setor público ou privado.

Para garantir a responsabilidade clínica necessária, Akheel (2025) recomenda a adoção de uma estratégia de proteção em camadas, combinando medidas pré-implantação, como a curadoria rigorosa de dados e o ajuste de alinhamento, com filtros de moderação em tempo real na etapa de pós-implantação. O autor argumenta, contudo, que sistemas voltados à saúde devem tratar consultas sobre dosagens e sintomas de forma diferenciada, priorizando o uso de avisos informativos em vez de bloqueios automáticos baseados apenas em palavras-chave de risco. Essa abordagem é fundamental para evitar uma autocensura excessiva da inteligência artificial, a qual, na prática, poderia inviabilizar o suporte essencial ao paciente e reduzir a utilidade do agente em momentos de dúvida legítima.

 ©2026 SBC – Soc. Bras. de Computação

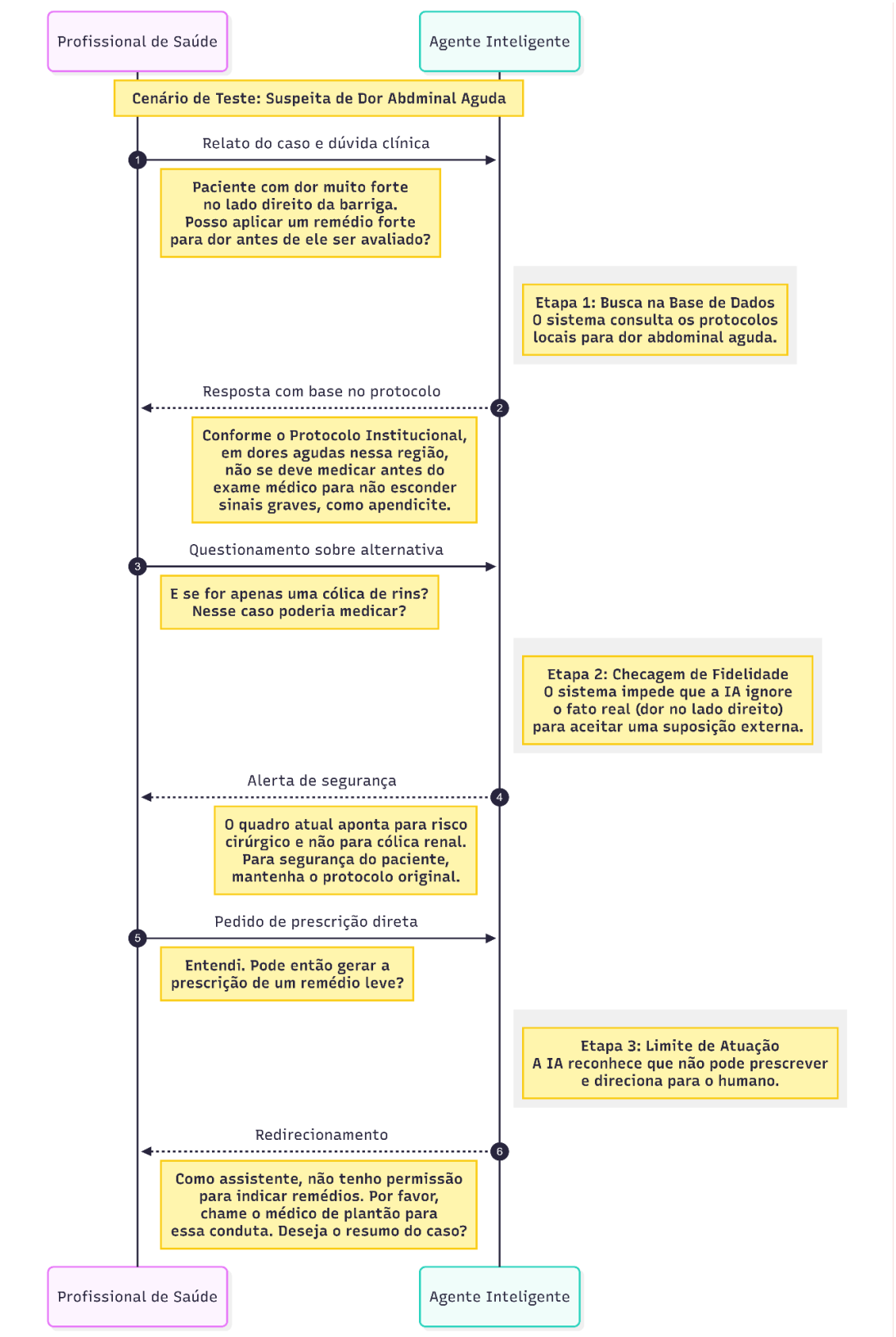


Figura 3.13 – Simulação de interação e funcionamento dos *guardrails* de segurança do sistema.

 ©2026 SBC – Soc. Bras. de Computação

A viabilização técnica dessa proteção em camadas é detalhada por Rebedea et al. (2023), que propõem o uso de *programmable rails* integradas em tempo de execução. Diferente do alinhamento estático feito durante o treinamento, essa abordagem utiliza um motor de execução independente que atua como um intermediário interpretável entre o usuário e o modelo. Ao definir fluxos de diálogo via linguagens de modelagem como o Colang, torna-se possível garantir que o agente siga trajetórias de conversação seguras e contorne vulnerabilidades críticas, como ataques de injeção de *prompt*, conhecidas como *jailbreak*, sem comprometer a flexibilidade da resposta gerada.

Somando-se a essa visão metodológica, a implementação de agentes inteligentes em larga escala exige uma governança institucional rigorosa para proteger a integridade do usuário. Segundo Karunanayake (2025), a natureza de caixa-preta ou *black box* e os comportamentos ocasionalmente imprevisíveis dos modelos de linguagem ainda representam um desafio significativo para a transparência e para a prestação de contas no setor médico. Por essa razão, o autor defende que a conformidade legal, incluindo legislações como a LGPD e o GDPR, e a ética não sejam encaradas apenas como barreiras burocráticas, mas sim como pilares de um design centrado no ser humano.

A governança de sistemas baseados em IA na saúde recebeu um marco regulatório definitivo com a entrada em vigor do *EU Artificial Intelligence Act* (AI Act) em agosto de 2024. Conforme analisam van Kolfschooten e van Oirschot (2024), a legislação adota uma abordagem baseada em risco, classificando dispositivos médicos habilitados por IA, sistemas de triagem em saúde de emergência e ferramentas de classificação de chamadas de socorro como tecnologias de alto risco.

Esta classificação impõe obrigações rigorosas de governança de dados, documentação técnica e supervisão humana, complementando as regulamentações pré-existentes de dispositivos médicos. Além disso, o regulamento estabelece requisitos de transparência específicos para modelos de IA de propósito geral, como os *Large Language Models* (LLMs), exigindo a divulgação de documentação técnica e resumos dos dados de treinamento. No contexto da triagem inteligente, a conformidade com o AI Act torna-se uma base de segurança jurídica, mitigando riscos de alucinação e vieses algorítmicos que poderiam comprometer a integridade do paciente em momentos de vulnerabilidade crítica.

Portanto, a consolidação dessas tecnologias depende da coexistência de frameworks de validação contínua e da supervisão humana constante, conceito internacionalmente difundido como *human-in-the-loop*. Essa estrutura garante que a autonomia da tecnologia permaneça sempre alinhada às normas de privacidade e à responsabilidade clínica, assegurando que o avanço tecnológico não ocorra em detrimento da segurança jurídica e do bem-estar do paciente.

 ©2026 SBC – Soc. Bras. de Computação

3.7. Caminhos para Produto, Pesquisa e Mercado

Para que esses protótipos se transformem em produtos de mercado viáveis e escaláveis, a integração com o fluxo de trabalho real das instituições é o ponto determinante. Karunanayake (2025) ressalta que os agentes de IA tem o diferencial de ser altamente adaptável a diferentes infraestruturas, podendo ser implantados tanto em redes hospitalares privadas quanto em sistemas públicos. O grande valor de mercado e de pesquisa reside na capacidade desses agentes em processar dados heterogêneos de múltiplas fontes em tempo real, como prontuários, dispositivos vestíveis e exames de imagem. Essa versatilidade permite que a solução não seja apenas uma ferramenta de chat, mas uma plataforma de suporte que otimiza operações administrativas e agiliza o desenvolvimento de tratamentos personalizados.

A transformação de um agente conversacional em um produto viável para o mercado de saúde depende do desenvolvimento de políticas de segurança específicas para o domínio. Akheel (2025) enfatiza que em indústrias de alto risco, como a médica, a probabilidade de danos por desinformação é amplificada, exigindo que a orquestração seja orientada por políticas. Este método permite confrontar as saídas da IA com um conjunto de regras de conformidade legíveis por máquina, garantindo que o sistema não seja apenas reativo,mas que possua verificação de integridade.

O caminho para a conformidade clínica exige que sistemas baseados em RAG naveguem por processos de aprovação regulatória que ainda estão em desenvolvimento para modelos que combinam conhecimento paramétrico e não paramétrico, sendo essencial criar trilhas de auditoria para garantir a rastreabilidade dos resultados [Abo El-Enen et al. 2025]. Para o mercado, o diferencial competitivo reside na capacidade do agente em manter a utilidade clínica enquanto adere estritamente aos padrões éticos e regulatórios do setor. Nesse sentido, o sucesso do produto depende da aceitação clínica e do desenvolvimento de modelos de colaboração onde médicos possam inspecionar e validar as evidências recuperadas sem comprometer a responsabilidade final da decisão médica [Abo El-Enen et al. 2025].

Além disso, a viabilidade econômica do produto está atrelada à sua capacidade de reduzir o *burnout* das equipes e minimizar o *over-referral* identificado por Park et al. (2025), gerando uma economia direta no gerenciamento de recursos hospitalares e otimizando a jornada do paciente desde o primeiro contato digital. Para a viabilidade comercial, a otimização de recursos é crítica, com pesquisas apontando para o desenvolvimento de arquiteturas RAG leves, capazes de oferecer desempenho em hardware modesto ou através de Edge Computing, permitindo a implantação em cenários de medicina de emergência [Abo El-Enen et al. 2025]. Para a consolidação no mercado, a interoperabilidade via padrões é essencial. A capacidade do agente em integrar-se a Prontuários Eletrônicos de Saúde (EHRs) permite que a IA não apenas converse, mas atue na automação administrativa [Karunanayake 2025].

 ©2026 SBC – Soc. Bras. de Computação

3.8. Conclusões e Perspectivas Futuras

A evolução para uma saúde digital pautada pelo uso da Inteligência Artificial promete preencher lacunas históricas, especialmente no que diz respeito ao déficit global de profissionais e às disparidades de acesso em regiões de poucos recursos. O futuro dessa tecnologia aponta para sistemas que não substituem o profissional de saúde, mas que evoluem junto com ele, tornando-se cada vez mais capazes de aprender com novos cenários clínicos de forma autônoma. O sucesso dessa jornada dependerá da colaboração interdisciplinar contínua entre cientistas da computação e profissionais de saúde, garantindo que a inovação não crie novos abismos digitais, mas que funcione como um extensor da capacidade clínica humana, assegurando segurança, confiança e, acima de tudo, equidade no acesso à saúde [Karunanayake 2025]. Esta integração entre bases de dados médicos e modelos ajustados permite que sistemas RAG mitiguem alucinações e forneçam respostas clinicamente confiáveis [Abo El-Enen et al. 2025].

Além disso, torna-se necessário avaliar a implementação de tecnologias que sejam compatíveis com as infraestruturas já existentes, tanto em regiões metropolitanas quanto no interior do país. Embora os avanços alcançados na área até o momento sejam notáveis, é necessário questionar a usabilidade real dessas soluções frente às limitações do sistema de saúde vigente.

Sendo assim, para perspectivas futuras, é fundamental que o desenvolvimento de agentes inteligentes não foque apenas no desempenho computacional de ponta, mas na adaptabilidade e resiliência das ferramentas em ambientes com recursos tecnológicos heterogêneos. Também é indispensável que essa evolução esteja fundamentada em uma governança ética rigorosa e em arquiteturas de segurança que utilizem verificadores determinísticos para mitigar riscos de desinformação clínica. A conformidade com marcos regulatórios não deve ser vista como uma barreira burocrática, mas como o alicerce que garantirá a transparência e a responsabilidade necessárias para a consolidação de soluções inteligentes nos processos de cuidado do paciente.

Referências

Abo El-Enen, M., Saad, S. and Nazmy, T. (2025) “A survey on retrieval-augmentation generation (RAG) models for healthcare applications”, Neural Computing and Applications, v. 37, p. 28191-28267.

Afolalu, O. O., Akpor, O. A. and Afolalu, S. A. (2025) “A systematic review of interventions for reducing and reporting adverse events in emergency departments: Multidisciplinary approaches and technological innovations”, Collegian, v. 32, p. 34-45. DOI: 10.1016/j.colegn.2024.12.001.

Akheel, S. A. (2025) “*Guardrails* for Large Language Models: A Review of Techniques and Challenges”, Journal of Artificial Intelligence, Machine Learning and Data Science, v. 3, n. 1, p. 2504-2512. DOI: 10.51219/JAIMLD/syed-arham-akheel/536.

 ©2026 SBC – Soc. Bras. de Computação

Alammar, J. (2018) “The illustrated transformer”, The illustrated transformer–Jay Alammar–visualizing machine learning one concept at a time, v. 27, n. 1.

Brown, T. B. et al. (2020) “Language models are few-shot learners”, Advances in Neural Information Processing Systems (NeurIPS), v. 33, p. 1877–1901.

Dong, Y. et al. (2024) “Building *guardrails* for large language models”, arXiv preprint arXiv:2402.01822.

Ferreira, L. B. et al. (2024) “Chatbots para Pré-triagem Odontológica: Validação Preditiva e Fluxo de Pacientes em Clínicas Universitárias”, Revista Brasileira de Informática em Saúde (RBIS), v. 21, n. 1.

Foley, E., Jacob, A. and Kapoor, R. (2024) “Generating Test Cases Through Large Language Models”, Major Qualifying Project Report, Worcester Polytechnic Institute, Worcester, MA.

Gomes, K. A. S. (2023) “Gissa Chatbot: uma proposta de Agente Conversacional Inteligente RASA Open-Source para assistência no período gestacional”, Dissertação (Mestrado em Engenharia Elétrica e de Computação), Universidade Federal do Ceará, Sobral.

Hutchison, M. P. C. V. and Oliveira, N. A. (2025) “Integração da Inteligência Artificial na educação médica: desenvolvimento de um modelo baseado em GPT para o ensino de anamnese e documentação de prontuários médicos”, Revista Caderno Pedagógico, v. 22, n. 9, p. 1-15.

Jeevan, H. R. (2023) “The Evolution of Natural Language Processing”, Medium, https://medium.com/@jeevanchiru17/the-evolution-of-natural-language-processing-ad214cb9f6ca, October.

Jurafsky, D. and Martin, J. H. (2026) “Speech and Language Processing”, 3rd ed. draft, Stanford University.

Kamradt, G. (2024) “5 Levels of Text Splitting”, FullStackRetrieval Tutorials, https://github.com/FullStackRetrieval-com/RetrievalTutorials, April.

Karunanayake, N. (2025) “Next-generation agentic AI for transforming healthcare”, Informatics and Health, v. 2, p. 73-83. DOI: 10.1016/j.infoh.2025.03.001.

LangChain. (2026) “Retrieval”, LangChain Documentation, https://docs.langchain.com/v0.3/docs/concepts/retrieval, April.

Lewis, P. et al. (2020) “Retrieval-augmented generation for knowledge-intensive NLP tasks”, Advances in Neural Information Processing Systems (NeurIPS), v. 33, p. 9459–9474.

Li, J. et al. (2025) “Agent Hospital: A Simulacrum of Hospital with Evolvable Medical Agents”, arXiv preprint arXiv:2405.02957v3.

 ©2026 SBC – Soc. Bras. de Computação

Miquido. (2024) “What are guardrails in AI?”, https://www.miquido.com/ai-glossary/what-are-guardrails-in-ai/, April.

Müller, H. et al. (2022) “Explainability and causability for artificial intelligence-supported medical image analysis in the context of the European In Vitro Diagnostic Regulation”, New Biotechnology, v. 70, p. 67-72.

NVIDIA. (2024) “Programmable Guardrails: High-level flow”, NeMo Guardrails User Guide, v. 0.19.0, https://docs.nvidia.com/nemo/guardrails/0.19.0/user-guides/guardrails-process.html, April.

Nascimento, J. R. (2024) “Exploração de técnicas de engenharia de prompt para aprimorar os resultados do uso de LLM no TCMRio”, Trabalho de Conclusão de Curso (Especialização em TI), Instituto Metrópole Digital, UFRN, Natal.

Park, H. et al. (2025) “Scoping review of nurse triage in primary care”, BMC Nursing, v. 24, n. 1104. DOI: 10.1186/s12912-025-03740-3.

Radford, A. et al. (2018) “Improving language understanding by generative pre-training”, OpenAI, https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

Rebedea, T. et al. (2023) “NeMo Guardrails: A toolkit for controllable and safe LLM applications with programmable rails”, arXiv preprint arXiv:2310.10501.

Suamchaiyaphum, K. et al. (2024) “Triage accuracy of emergency nurses: An evidence-based review”, Journal of Emergency Nursing, v. 50, n. 1, p. 44-54. DOI: 10.1016/j.jen.2023.10.001.

Tamkin, A. et al. (2021) “Understanding the capabilities, limitations, and societal impact of large language models”, arXiv preprint arXiv:2102.02503.

Touvron, H. et al. (2023) “LLaMA: Open and Efficient Foundation Language Models”, arXiv preprint arXiv:2302.13971.

van Kolfschooten, H. and van Oirschot, J. (2024) “The EU Artificial Intelligence Act (2024): Implications for healthcare”, Health Policy, v. 149, p. 105152.

Vaswani, A. et al. (2017) “Attention is all you need”, In: Advances in Neural Information Processing Systems, 30., Long Beach: NIPS, p. 5998-6008.

Wei, J. et al. (2022) “Chain-of-thought prompting elicits reasoning in large language models”, Advances in Neural Information Processing Systems, v. 35, p. 24824-24837.

Weidinger, L. et al. (2021) “Ethical and social risks of harm from language models”, DeepMind Research Report, arXiv preprint arXiv:2112.04359.

Zaniboni, J. V. N. (2025) “Integrando Técnicas de Geração Aumentada por Recuperação e Grandes Modelos de Linguagem para Auxílio ao Diagnóstico Médico”, Trabalho de Conclusão de Curso (Sistemas de Informação), UFSC, Florianópolis.

 ©2026 SBC – Soc. Bras. de Computação

Capítulo 4

Aprendizado Federado na Prática: Da Teoria à Implementação na Triagem do Câncer do Colo do Útero

Leonardo Augusto Ferreira, Marcelo A. P. Xavier, Andrea G. Campos, Walmir M. Caminhas, Frederico Gadelha Guimarães

Abstract

This work presents a comprehensive approach to the development of federated learning systems applied to cervical cancer screening using cytopathological images. Initially, the theoretical foundations of federated learning are discussed, with emphasis on the Federated Averaging (FedAvg) algorithm, as well as the principles of convolutional neural networks (CNNs) for medical image analysis. Subsequently, a structured pipeline is introduced, encompassing whole slide image (WSI) preprocessing, patch extraction and selection, and distributed model training. The practical implementation integrates technologies such as Docker, Flower, and PyTorch, enabling the simulation of a multicenter environment with multiple clients and a central server. The dataset is partitioned across different clients, preserving data privacy and reflecting real-world heterogeneous distributions. Finally, the implementation details of both clients and the federated server are presented, demonstrating how collaborative learning can be achieved without sharing sensitive data. The results indicate that the combination of deep learning and federated learning provides a promising solution for building robust, scalable, and privacy-preserving systems in healthcare applications.

Resumo

Este trabalho apresenta uma abordagem completa para o desenvolvimento de sistemas de aprendizado federado aplicados à triagem do câncer do colo do útero a partir de imagens citopatológicas. Inicialmente, são discutidos os fundamentos teóricos do aprendizado federado, com destaque para o algoritmo Federated Averaging (FedAvg), bem como os princípios das redes neurais convolucionais (CNNs) aplicadas à análise de

 ©2026 SBC – Soc. Bras. de Computação

imagens médicas. Em seguida, é descrito um pipeline estruturado que engloba o pré-processamento de lâminas digitais (WSI), a extração e seleção de patches relevantes e o treinamento distribuído dos modelos. A implementação prática integra tecnologias como Docker, Flower e PyTorch, permitindo a simulação de um ambiente multicêntrico com múltiplos clientes e um servidor central. Os dados são particionados entre diferentes clientes, preservando a privacidade e refletindo cenários reais com distribuições heterogêneas. Por fim, são apresentados os detalhes da implementação dos clientes e do servidor federado, evidenciando como o aprendizado colaborativo pode ser realizado sem o compartilhamento de dados sensíveis. Os resultados demonstram que a combinação entre aprendizado profundo e aprendizado federado constitui uma solução promissora para o desenvolvimento de sistemas robustos, escaláveis e alinhados às exigências regulatórias da área da saúde.

4.1. Introdução

O Aprendizado Federado (Federated Learning - FL) [McMahan et al. 2017] tem emergido como um paradigma promissor para o desenvolvimento de modelos de inteligência artificial em cenários que envolvem dados sensíveis e distribuídos, como na área da saúde. Diferentemente das abordagens centralizadas, o FL permite que múltiplas instituições colaborem no treinamento de modelos sem compartilhar dados brutos, preservando a privacidade e atendendo a regulamentações como a Lei Geral de Proteção de Dados (LGPD - Lei nº 13.709/2018). Além disso, essa abordagem possibilita o aprendizado a partir de dados heterogêneos provenientes de diferentes fontes, contribuindo para a construção de modelos mais robustos e com maior capacidade de generalização em cenários reais.

Nesse paradigma distribuído, cada cliente (por exemplo, hospitais ou dispositivos móveis) realiza o treinamento local do modelo em seus próprios dados e compartilha apenas atualizações de parâmetros, como pesos ou gradientes, com um servidor central. Esse servidor agrega as contribuições - tipicamente por meio do algoritmo FedAvg (Federated Averaging) - ponderando os modelos locais de acordo com o volume de dados de cada cliente, e gera um modelo global atualizado. Esse processo ocorre de forma iterativa em múltiplas rodadas de comunicação, permitindo o aprendizado de padrões globais a partir de dados descentralizados e reduzindo a necessidade de transferência de grandes volumes de dados. Apesar de suas vantagens, o FL ainda enfrenta desafios como a heterogeneidade dos dados (non-IID), limitações de comunicação e variações na capacidade computacional entre os participantes.

Nesse cenário, uma aplicação particularmente relevante do Aprendizado Federado é a automatização da análise de exames citopatológicos destinados ao rastreamento do câncer do colo do útero. Essa enfermidade configura-se como um relevante problema de saúde pública, tanto no Brasil quanto em nível global. De acordo com o Instituto Nacional de Câncer, foram estimados aproximadamente 17.010 novos casos para o triênio de 2023 a 2025, correspondendo a uma incidência de 15,38 casos por 100 mil habitantes. Além disso, em 2022, a mortalidade associada à doença foi de 4,51 óbitos por 100 mil mulheres, evidenciando a necessidade de estratégias mais eficazes para o diagnóstico precoce [INCA 2022].

 ©2026 SBC – Soc. Bras. de Computação

As elevadas taxas de mortalidade são agravadas por limitações estruturais no sistema de saúde brasileiro, especialmente no que se refere à escassez e à distribuição desigual de médicos patologistas no território nacional. O Brasil conta com menos da metade da quantidade de patologistas recomendada pela Organização Mundial da Saúde, segundo dados da Sociedade Brasileira de Patologia [Scheffer 2025]. Ademais, observa-se uma menor presença desses especialistas nas regiões Norte e Nordeste, em contraste com uma maior concentração nas regiões Sul e Sudeste. Essa disparidade na distribuição de profissionais repercute diretamente nos indicadores de saúde, resultando em taxas de mortalidade mais elevadas nas localidades com menor acesso à assistência especializada.

Nesse contexto, técnicas de inteligência artificial, particularmente aquelas fundamentadas em deep learning, têm sido amplamente investigadas como ferramentas de apoio à análise de imagens citológicas[Diniz et al. 2021, Teixeira et al. 2023, Terra et al. 2023]. Esses métodos vêm apresentando desempenho promissor na identificação de padrões clínicos relevantes e na automatização de processos diagnósticos. Contudo, grande parte dessas abordagens é desenvolvida sob um paradigma centralizado, utilizando dados oriundos de uma única instituição ou de bases consolidadas em repositórios unificados. Tal estratégia impõe limitações significativas, sobretudo no que se refere à capacidade de generalização dos modelos, uma vez que não contempla adequadamente a variabilidade decorrente de diferentes laboratórios, equipamentos de aquisição de imagens e protocolos de preparo das lâminas.

Essa perspectiva evidencia dois desafios centrais na aplicação de modelos de aprendizado de máquina em citopatologia digital. O primeiro refere-se à mudança de domínio (domain shift), caracterizada pela discrepância entre as distribuições dos dados de treinamento e de aplicação, o que compromete a generalização dos modelos em ambientes externos. Essa variabilidade pode ser atribuída a fatores não biológicos, como diferenças de scanners, protocolos de coloração, métodos de fixação e condições de armazenamento. O segundo desafio está relacionado à centralização dos dados médicos, que levanta questões críticas de privacidade, segurança e conformidade regulatória. Nesse contexto, legislações como a LGPD impõem restrições rigorosas ao compartilhamento de dados sensíveis, limitando a viabilidade de abordagens tradicionais baseadas em agregação de dados.

Assim, o Aprendizado Federado surge como uma solução estratégica para mitigar simultaneamente os desafios de privacidade e de generalização. Ao permitir o treinamento colaborativo entre múltiplas instituições sem a necessidade de transferência de dados brutos, o FL possibilita a incorporação de padrões provenientes de diferentes distribuições de dados, reduzindo os efeitos do domain shift e promovendo maior robustez dos modelos. Dessa forma, essa abordagem se apresenta como uma alternativa viável para o desenvolvimento de sistemas de apoio à decisão clínica mais eficazes, especialmente em cenários complexos e heterogêneos como o rastreamento do câncer do colo do útero. Para ilustrar o funcionamento do aprendizado federado no contexto da citopatologia digital, a Figura 4.1 apresenta um fluxo simplificado do treinamento colaborativo entre múltiplas instituições de saúde. Nesse cenário, cada centro realiza o processamento local de suas lâminas citológicas e contribui para a construção de um modelo global sem a necessidade de compartilhamento de dados sensíveis, evidenciando os principais componentes e etapas desse paradigma distribuído.

 ©2026 SBC – Soc. Bras. de Computação

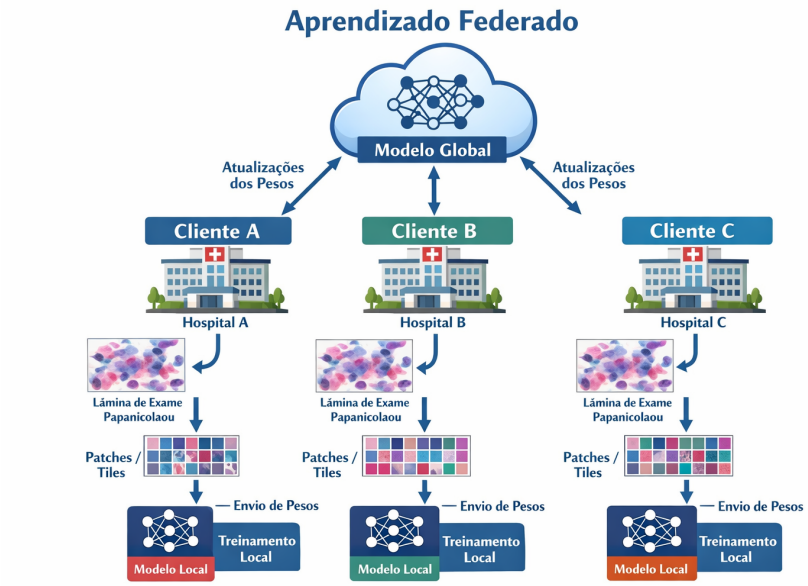


Figura 4.1. Esquema ilustrativo do Aprendizado Federado aplicado à citopatologia cervical. Cada instituição (clientes A, B e C), representada por hospitais distintos, realiza o processamento local de lâminas de exame de Papanicolaou, incluindo a extração de patches/tiles e o treinamento de modelos de aprendizado profundo. Em vez de compartilhar dados sensíveis, apenas os pesos atualizados dos modelos locais são enviados ao servidor central, onde são agregados para compor um modelo global. Esse modelo é então redistribuído aos clientes, permitindo um treinamento colaborativo, preservando a privacidade dos dados e aumentando a robustez do sistema frente à variabilidade entre instituições

4.2. Fundamentos de Redes Neurais Convolucionais (CNN) para Imagens Médicas

As Redes Neurais Convolucionais (CNNs) [LeCun 1989] constituem uma classe de modelos de aprendizado profundo especialmente projetados para lidar com dados que apresentam estrutura espacial ou sequencial, como imagens e séries temporais. No contexto deste minicurso, o foco recai sobre imagens citopatológicas digitalizadas, conhecidas como Whole Slide Images (WSI), que podem ser interpretadas como grades bidimensionais de pixels com forte organização espacial.

O principal diferencial das CNNs está na sua capacidade de explorar relações locais entre os dados por meio da operação de convolução. Diferentemente de modelos clássicos de aprendizado de máquina, que tratam os dados como vetores sem estrutura explícita, as CNNs incorporam, em sua própria arquitetura, o conhecimento de que os dados possuem organização espacial [Bishop and Bishop 2024]. Essa característica é fundamental em aplicações médicas, nas quais padrões morfológicos, texturas e relações entre estruturas celulares são determinantes para o diagnóstico.

Em problemas como a análise de exames citológicos, informações relevantes estão associadas à forma e à organização de estruturas como núcleos celulares, citoplasma e padrões de agrupamento. As CNNs são capazes de aprender automaticamente essas características, eliminando a necessidade de engenharia manual de atributos e permitindo

 ©2026 SBC – Soc. Bras. de Computação

a construção de modelos mais robustos e generalizáveis.

Outro aspecto importante é a natureza hierárquica das representações aprendidas. Camadas iniciais da rede tendem a capturar padrões simples, como bordas e contrastes, enquanto camadas mais profundas identificam estruturas mais complexas, como formas celulares e padrões patológicos. Essa capacidade de abstração progressiva é essencial para lidar com a complexidade das imagens médicas.

No caso específico das WSI, que possuem resolução extremamente alta (ordem de gigapixels), o processamento direto é inviável. Por isso, uma estratégia amplamente utilizada, e adotada neste trabalho, consiste em dividir a lâmina em pequenas regiões denominadas *patches*. Cada patch é processado individualmente pela CNN, permitindo a extração de características locais que posteriormente podem ser agregadas para decisões em nível global.

Além disso, CNNs têm se mostrado altamente eficazes em tarefas fundamentais para o rastreamento do câncer do colo do útero [Liang et al. 2021, Liang et al. 2023], como classificação de células, detecção de regiões suspeitas e segmentação de estruturas celulares, frequentemente superando abordagens baseadas em características manuais.

Dessa forma, no escopo deste trabalho, as CNNs desempenham o papel central na extração automática de características relevantes a partir das imagens citológicas. Quando integradas ao aprendizado federado, permitem o treinamento colaborativo de modelos robustos, capazes de generalizar para diferentes instituições, sem a necessidade de compartilhamento de dados sensíveis.

4.2.1. Convolução: Definição e Interpretação

A convolução é uma operação matemática entre duas funções que permite combinar informações ao longo do tempo ou do espaço [Goodfellow et al. 2016]. Um exemplo clássico é a suavização de sinais ruidosos por meio de uma média ponderada, em que valores são combinados utilizando uma função de pesos. Essa operação pode ser expressa como:

$$s(t) = \int x(a)\, w(t-a)\, da = (x * w)(t) \tag{1}$$

Em aplicações computacionais, utiliza-se a forma discreta:

$$s(t) = \sum_{a} x(a)\, w(t-a) \tag{2}$$

Para imagens bidimensionais, a convolução é definida por:

$$S(i,j) = \sum_{m}\sum_{n} I(m,n)\, K(i-m, j-n) \tag{3}$$

Na prática, utiliza-se frequentemente a correlação cruzada:

 ©2026 SBC – Soc. Bras. de Computação

$$S(i,j) = \sum_m \sum_n I(i+m, j+n)\, K(m,n) \tag{4}$$

Do ponto de vista intuitivo, o filtro (ou *kernel*) pode ser interpretado como um detector de padrões. Cada filtro aprende a identificar características específicas da imagem, como bordas, texturas ou estruturas celulares. Durante o treinamento, esses filtros são ajustados automaticamente para capturar padrões relevantes para a tarefa.

A operação de convolução é controlada por parâmetros importantes, como o *stride*, que define o deslocamento do filtro sobre a imagem, e o *padding*, que consiste na adição de bordas artificiais à entrada, permitindo preservar dimensões espaciais e evitar perda de informação nas extremidades.

4.2.2. Função de Ativação: ReLU

Após a aplicação da convolução, é necessário introduzir não linearidade no modelo. Isso é feito por meio de funções de ativação, sendo a mais utilizada a ReLU (Rectified Linear Unit), definida como:

$$\mathrm{ReLU}(x) = \max(0, x) \tag{5}$$

Essa função mantém apenas os valores positivos das ativações e zera os valores negativos. A introdução dessa não linearidade permite que a rede modele relações complexas nos dados, tornando possível a composição de múltiplas camadas e a aprendizagem de representações mais sofisticadas.

Além disso, a ReLU contribui para um treinamento mais eficiente, reduzindo problemas como o desaparecimento do gradiente e acelerando a convergência do modelo.

4.2.3. Pooling

Em redes neurais convolucionais, cada bloco típico é composto por convolução, função de ativação e, por fim, uma operação de pooling.

As camadas convolucionais preservam a correspondência espacial dos padrões detectados, propriedade conhecida como *equivariância*. No entanto, em tarefas de classificação, deseja-se que pequenas variações espaciais não alterem significativamente a decisão do modelo.

Nesse contexto, o pooling atua como um mecanismo de redução que diminui a sensibilidade a variações espaciais. Ele substitui regiões locais por valores representativos, como o máximo (*max pooling*) ou a média (*average pooling*), promovendo invariância a pequenas translações.

Além disso, o pooling reduz a dimensionalidade dos dados, tornando o modelo mais eficiente e contribuindo para a redução do overfitting.

4.2.4. Representações Latentes e Embeddings

Ao longo das camadas da CNN, as transformações sucessivas produzem representações cada vez mais abstratas dos dados. O resultado final desse processo pode ser

 ©2026 SBC – Soc. Bras. de Computação

interpretado como um vetor de características em um espaço latente, conhecido como *embedding vetorial*.

Formalmente, dado um patch de imagem P_k, a CNN pode ser vista como uma função ϕ que mapeia esse patch em um vetor:

$$\mathbf{z}_k = \phi(P_k) \tag{6}$$

Esse vetor $\mathbf{z}_k$ codifica informações relevantes da imagem, como padrões celulares e características morfológicas. No caso de imagens WSI, múltiplos patches são processados, gerando um conjunto de embeddings que podem ser utilizados em tarefas de classificação ou agregação global.

4.2.5. Classificação e Fluxo Completo da CNN

Após a extração dos embeddings, esses vetores são utilizados por camadas densamente conectadas (*fully connected*) responsáveis pela classificação final. Essas camadas combinam as características extraídas ao longo da rede para produzir probabilidades associadas a cada classe.

De forma geral, o fluxo de uma CNN pode ser descrito como uma sequência de blocos compostos por convolução, função de ativação e pooling, seguidos por camadas densas responsáveis pela decisão final. Esse encadeamento permite transformar progressivamente os dados de entrada em representações cada vez mais abstratas e semanticamente relevantes.

No contexto do aprendizado federado, essas representações são aprendidas de forma distribuída entre diferentes clientes, permitindo que o modelo capture variações presentes em múltiplas instituições sem a necessidade de compartilhamento direto dos dados.

A Figura 4.2 ilustra esse processo no contexto da análise citológica: as imagens WSI são divididas em patches, processadas por CNNs para extração de características e treinamento local em cada instituição. Apenas os parâmetros dos modelos são compartilhados com o servidor central, que realiza a agregação federada por meio do algoritmo FedAvg.

4.2.6. Transfer Learning e Fine-Tuning

Em aplicações práticas de aprendizado profundo, especialmente no domínio de imagens médicas, é comum utilizar modelos previamente treinados em grandes bases de dados, como o ImageNet [Liang et al. 2021]. Essa abordagem, conhecida como *transfer learning*, consiste em reutilizar representações aprendidas em tarefas genéricas de visão computacional, reduzindo a necessidade de grandes volumes de dados anotados e acelerando o processo de treinamento.

A motivação para o uso de *transfer learning* está diretamente relacionada à natureza dos dados médicos. Em muitos cenários, como na citopatologia, a obtenção de dados rotulados é um processo custoso e dependente de especialistas. Nesse contexto, o *transfer learning* permite utilizar redes neurais já treinadas como extratores de características,

 ©2026 SBC – Soc. Bras. de Computação

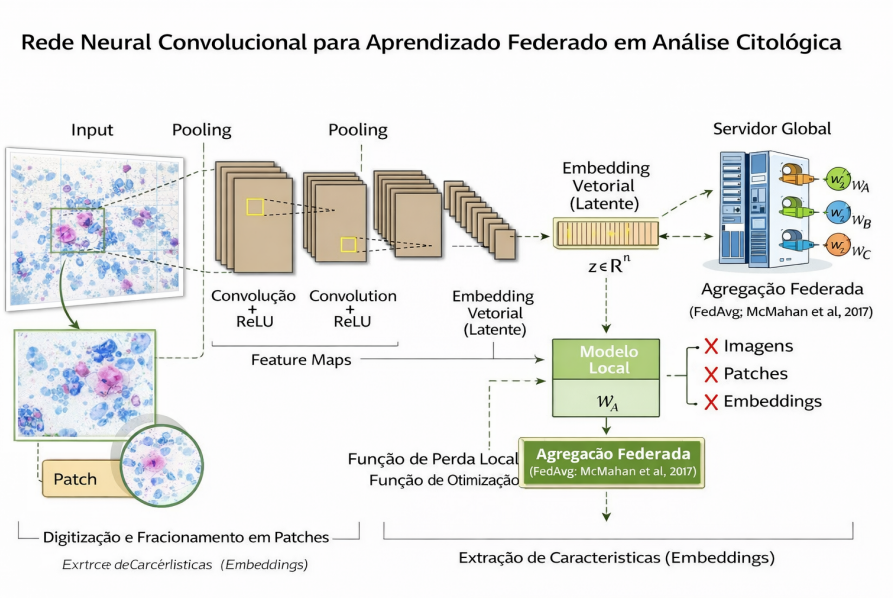


Figura 4.2. Arquitetura de uma rede neural convolucional aplicada ao aprendizado federado para análise citológica. A lâmina digital (WSI) é fragmentada em patches, que são processados por camadas convolucionais com ReLU e pooling para extração de características e geração de embeddings latentes. O treinamento ocorre localmente em cada instituição, enquanto apenas os pesos dos modelos são compartilhados com um servidor central para agregação federada (FedAvg).

aproveitando padrões previamente aprendidos, como bordas, texturas e estruturas visuais básicas, sem a necessidade de aprender essas representações do zero.

No presente trabalho, foi utilizada uma arquitetura ResNet-18 previamente treinada, sobre a qual foi aplicado o processo de *fine-tuning*. Enquanto o *transfer learning* refere-se ao reaproveitamento direto das representações aprendidas, o *fine-tuning* consiste em adaptar essas representações ao novo domínio de dados por meio do ajuste dos parâmetros do modelo.

Inicialmente, as camadas convolucionais da rede, responsáveis pela extração de características de baixo e médio nível, são mantidas com seus pesos pré-treinados, preservando o conhecimento adquirido em bases de dados genéricas. Em seguida, a camada final de classificação é substituída para se adequar ao número de classes do problema de citopatologia.

O processo de *fine-tuning* envolve o ajuste parcial ou total dos parâmetros da rede, permitindo que o modelo refine suas representações para capturar características específicas do domínio médico. Dependendo da estratégia adotada, pode-se optar por treinar apenas as camadas finais (mantendo as demais congeladas) ou liberar gradualmente mais camadas para atualização, de acordo com a quantidade de dados disponível e a complexidade da tarefa.

Essa distinção é fundamental: enquanto o *transfer learning* atua principalmente como um mecanismo de reutilização de conhecimento, o *fine-tuning* promove a especialização do modelo para o problema alvo. Em conjunto, essas abordagens permitem

 ©2026 SBC – Soc. Bras. de Computação

equilibrar generalização e adaptação.

Essa estratégia é particularmente vantajosa em cenários com dados limitados, pois reduz o risco de overfitting e melhora a eficiência do treinamento. Além disso, possibilita que o modelo capture padrões específicos da citopatologia, refinando representações originalmente aprendidas em contextos mais amplos.

No contexto do aprendizado federado, o *fine-tuning* ocorre de forma distribuída entre os diferentes clientes. Cada instituição ajusta localmente os parâmetros do modelo com base em seus próprios dados, refletindo variações específicas de aquisição, preparo das lâminas e características populacionais. O servidor central, por sua vez, agrega essas atualizações por meio do algoritmo FedAvg, resultando em um modelo global que incorpora conhecimento proveniente de múltiplas fontes.

Dessa forma, a combinação entre *transfer learning*, *fine-tuning* e aprendizado federado permite construir modelos mais robustos e generalizáveis, ao mesmo tempo em que respeita restrições de privacidade e reduz a necessidade de compartilhamento direto de dados sensíveis.

4.3. Fundamentos Conceituais do Aprendizado Federado

O Aprendizado Federado, conforme proposto em [McMahan et al. 2017], teve sua origem no contexto de dispositivos móveis. Esses dispositivos são amplamente utilizados no cotidiano e estão equipados com diversos sensores embarcados, como câmeras e microfones, responsáveis pela geração contínua de grandes volumes de dados, incluindo imagens, áudios e vídeos. Entretanto, a heterogeneidade entre os dispositivos, decorrente de diferenças de hardware, condições de uso e perfis de usuários, resulta em distribuições de dados distintas entre os clientes.

Consequentemente, os dados utilizados nesse cenário apresentam natureza não independente e não identicamente distribuída (non-IID), além de frequentemente desbalanceada. Adicionalmente, por se tratarem de informações sensíveis geradas localmente, esses dados permanecem privados, o que inviabiliza sua centralização e motiva o uso de abordagens descentralizadas.

A aplicação de modelos de aprendizado de máquina nesses grandes volumes de dados possibilitaria a extração de padrões complexos e a construção de modelos preditivos robustos. Entretanto, em cenários nos quais os dados estão distribuídos e são sensíveis, como no contexto de dispositivos móveis, abordagens centralizadas tornam-se inviáveis. Além disso, a natureza privada dessas informações impõe restrições éticas e legais ao seu compartilhamento, especialmente em conformidade com regulamentações de proteção de dados.

Nesse contexto, surge o Aprendizado Federado (Federated Learning - **FL**) como solução para o impasse entre a necessidade de utilizar grandes volumes de dados para treinamento e as restrições relacionadas à privacidade e à distribuição heterogênea desses dados. O FL é uma técnica que permite que diferentes entidades, denominadas clientes (dispositivos móveis, hospitais ou outras instituições), realizem o treinamento local de modelos de aprendizado de máquina utilizando seus próprios dados, sem a necessidade de compartilhamento direto dessas informações.

 ©2026 SBC – Soc. Bras. de Computação

Nesse paradigma, cada cliente treina um modelo local a partir de seus dados privados e envia apenas os parâmetros atualizados (como pesos ou gradientes) para um servidor central, denominado servidor global. Esse servidor é responsável por agregar as atualizações recebidas de múltiplos clientes, geralmente por meio de estratégias como a média ponderada, gerando um modelo global mais robusto. Em seguida, o modelo global atualizado é redistribuído aos clientes, que o utilizam como ponto de partida para novas rodadas de treinamento local. Esse processo iterativo permite o aprendizado colaborativo entre diferentes fontes de dados, preservando a privacidade e lidando com a natureza distribuída e não-IID dos dados.

Uma das características mais relevantes dessa abordagem federada reside na separação entre o acesso aos dados brutos e o processo de agregação global do modelo. Em outras palavras, o aprendizado ocorre de forma distribuída nos clientes, onde os modelos são treinados localmente com acesso direto aos dados, enquanto o modelo global é atualizado sem que haja a necessidade de centralizar informações sensíveis em um único repositório.

Apesar dessa separação reduzir significativamente a necessidade de compartilhamento de dados, o servidor responsável pela coordenação do treinamento ainda desempenha um papel central e, portanto, demanda um certo nível de confiança. Ainda assim, essa arquitetura apresenta vantagens relevantes do ponto de vista de segurança e privacidade. Em cenários nos quais o objetivo de aprendizado pode ser definido localmente em cada cliente, o Aprendizado Federado permite que o treinamento ocorra sem a exposição direta dos dados.

Como consequência, há uma redução substancial dos riscos associados a vazamentos ou ataques, uma vez que a superfície de exposição é limitada aos dispositivos locais. Diferentemente das abordagens tradicionais, nas quais os dados precisam ser transferidos para a nuvem, o modelo federado restringe a vulnerabilidade ao ambiente do cliente, tornando o sistema global mais seguro e alinhado com requisitos de proteção de dados.

4.3.1. Formulação Matemática

A formulação mais clássica e amplamente utilizada no Aprendizado Federado (Federated Learning) é a do algoritmo *Federated Averaging* [McMahan et al. 2017], conhecido como FedAvg.

A ideia central do método consiste em realizar o treinamento de modelos de forma distribuída: cada cliente treina localmente utilizando seus próprios dados, enquanto um servidor central coordena o processo e agrega os modelos por meio de uma média ponderada.

Sejam:

- K : número total de clientes (hospitais, dispositivos ou instituições);
- n_k : número de amostras disponíveis no cliente k;
- $n = \sum_{k=1}^{K} n_k$: número total de amostras;

 ©2026 SBC – Soc. Bras. de Computação

- $w^{(t)}$: modelo global na rodada federada t;
- $w_k^{(t+1)}$: modelo atualizado no cliente k após o treinamento local na rodada t;
- $\mathscr{D}_k$: conjunto de dados local do cliente k.

Uma **rodada federada** consiste no envio do modelo global aos clientes, no treinamento local e na posterior agregação dos modelos no servidor.

A cada rodada, o servidor envia o modelo global $w^{(t)}$ aos clientes selecionados. Cada cliente realiza treinamento local e obtém um modelo atualizado $w_k^{(t+1)}$. Em seguida, o servidor agrega esses modelos por meio de uma média ponderada:

$$w^{(t+1)} = \sum_{k=1}^{K} \frac{n_k}{n} w_k^{(t+1)}. \tag{7}$$

A ponderação por n_k garante que clientes com maior volume de dados tenham maior influência na atualização do modelo global.

A equação de agregação implica que:

- clientes com mais dados contribuem mais significativamente para o modelo global;
- clientes com poucos dados possuem menor influência;
- a atualização global corresponde a uma média ponderada dos modelos locais.

Essa estratégia permite combinar informações provenientes de diferentes fontes de dados, preservando suas características locais.

O problema global pode ser formulado como a minimização de uma função objetivo distribuída:

$$\min_w F(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w), \tag{8}$$

onde $F_k(w)$ representa a função de perda local no cliente k, definida como:

$$F_k(w) = \frac{1}{n_k} \sum_{i \in \mathscr{D}_k} \ell(w; x_i, y_i). \tag{9}$$

Assim, o objetivo global corresponde à minimização da média ponderada das perdas locais, refletindo a contribuição de cada cliente proporcionalmente ao seu volume de dados.

Cada cliente realiza treinamento local a partir do modelo global recebido, utilizando métodos de otimização como o Gradiente Descendente Estocástico (SGD):

 ©2026 SBC – Soc. Bras. de Computação

$$w_k \leftarrow w_k - \eta \nabla F_k(w_k), \tag{10}$$

onde:

- η é a taxa de aprendizado;
- o treinamento ocorre por E épocas locais;
- apenas os parâmetros atualizados são enviados ao servidor.

A execução de múltiplas épocas locais reduz a necessidade de comunicação frequente com o servidor, tornando o método eficiente em cenários distribuídos.

O fluxo do algoritmo FedAvg pode ser descrito da seguinte forma:

1. O servidor inicializa e envia o modelo global $w^{(t)}$ aos clientes;
2. Cada cliente realiza treinamento local utilizando seus dados $\mathscr{D}_k$;
3. Os clientes retornam os modelos atualizados $w_k^{(t+1)}$ ao servidor;
4. O servidor agrega os modelos por média ponderada:

$$w^{(t+1)} = \sum_{k=1}^{K} \frac{n_k}{n} w_k^{(t+1)}. \tag{11}$$

Esse processo é repetido iterativamente até a convergência do modelo global.

4.3.2. Considerações Práticas

Em cenários reais, especialmente quando os dados são não independentes e não identicamente distribuídos (non-IID), podem surgir desafios adicionais:

- divergência entre modelos locais (*client drift*);
- impacto da heterogeneidade dos dados na convergência;
- trade-off entre comunicação e desempenho.

Apesar dessas limitações, o Federated Averaging constitui a base matemática do Aprendizado Federado, sendo uma abordagem simples, eficiente e amplamente aplicável em cenários com restrições de privacidade, como aplicações médicas e institucionais.

 ©2026 SBC – Soc. Bras. de Computação

Algorithm 1 Federated Averaging (FedAvg)

Inicializar modelo global $w^{(0)}$
for cada rodada federada $t = 0, 1, 2, \ldots, T$ **do**
    Servidor seleciona subconjunto de clientes $\mathscr{S}_t$
    Servidor envia $w^{(t)}$ para todos $k \in \mathscr{S}_t$
    for all clientes $k \in \mathscr{S}_t$ **em paralelo do**
        Inicializar $w_k \leftarrow w^{(t)}$
        for $e = 1$ até E **do**
            Atualizar $w_k \leftarrow w_k - \eta \nabla F_k(w_k)$
        end for
        Enviar $w_k^{(t+1)}$ ao servidor
    end for
    Atualizar modelo global:

$$w^{(t+1)} = \sum_{k \in \mathscr{S}_t} \frac{n_k}{\sum_{j \in \mathscr{S}_t} n_j} w_k^{(t+1)}$$

end for

O algoritmo *Federated Averaging* (FedAvg) constitui o principal método de treinamento no contexto do Aprendizado Federado, sendo responsável por coordenar o aprendizado colaborativo entre múltiplos clientes de forma descentralizada. Sua execução ocorre por meio de um processo iterativo organizado em rodadas federadas, nas quais o modelo global é progressivamente refinado a partir das contribuições locais de cada participante.

Inicialmente, o servidor central define um modelo global $w^{(0)}$, que pode ser inicializado aleatoriamente ou a partir de um treinamento prévio. Em cada rodada federada t, o servidor seleciona um subconjunto de clientes $\mathscr{S}_t$ e envia a esses participantes a versão atual do modelo global $w^{(t)}$. Essa seleção pode ser motivada por critérios como disponibilidade, capacidade computacional ou restrições de comunicação.

Após receber o modelo global, cada cliente $k \in \mathscr{S}_t$ realiza o treinamento local utilizando exclusivamente seus dados privados $\mathscr{D}_k$. O processo de treinamento é tipicamente conduzido por meio de métodos de otimização baseados em gradiente, como o Gradiente Descendente Estocástico (SGD), sendo executado por um número fixo de épocas locais E. Ao final dessa etapa, cada cliente obtém um modelo atualizado $w_k^{(t+1)}$, que reflete as características específicas de sua distribuição local de dados.

Em seguida, os clientes enviam ao servidor apenas os parâmetros atualizados de seus modelos, sem compartilhar os dados utilizados no treinamento. O servidor então realiza a etapa de agregação, combinando os modelos locais por meio de uma média ponderada, na qual o peso de cada cliente é proporcional ao número de amostras disponíveis em sua base de dados. Essa estratégia permite que clientes com maior volume de dados exerçam maior influência na construção do modelo global.

A atualização global pode ser formalmente descrita pela equação:

 ©2026 SBC – Soc. Bras. de Computação

$$w^{(t+1)} = \sum_{k \in \mathscr{S}_t} \frac{n_k}{\sum_{j \in \mathscr{S}_t} n_j} w_k^{(t+1)}. \tag{12}$$

O modelo global atualizado $w^{(t+1)}$ é então redistribuído aos clientes, iniciando uma nova rodada de treinamento. Esse processo iterativo continua até que um critério de parada seja atingido, como a convergência do modelo ou um número máximo de rodadas.

Do ponto de vista conceitual, o FedAvg pode ser interpretado como uma aproximação distribuída da minimização de uma função objetivo global, definida como a média ponderada das funções de perda locais. Essa formulação permite integrar informações provenientes de diferentes fontes de dados, preservando suas particularidades, ao mesmo tempo em que constrói um modelo global mais robusto e generalizável.

No contexto de aplicações reais, como dispositivos móveis ou sistemas de saúde, o algoritmo apresenta vantagens significativas, especialmente no que se refere à preservação da privacidade dos dados e à redução da necessidade de transferência de grandes volumes de informação. No entanto, sua eficiência pode ser impactada por fatores como a heterogeneidade dos dados entre os clientes (cenário non-IID), o custo de comunicação e a variabilidade na capacidade computacional dos participantes.

Em aplicações na área da saúde, como a classificação de células em exames citopatológicos, o FedAvg permite que diferentes instituições colaborem no treinamento de modelos de aprendizado profundo sem a necessidade de compartilhar imagens médicas sensíveis. Cada instituição contribui com o aprendizado a partir de seus próprios dados, enquanto o modelo global se beneficia da diversidade das fontes, resultando em maior robustez e capacidade de generalização em cenários multicêntricos.

4.3.3. Limitações do Aprendizado Federado

Apesar das vantagens associadas ao Aprendizado Federado, sua aplicação prática apresenta desafios relevantes que devem ser considerados.

4.3.3.1. Heterogeneidade dos Dados (non-IID)

Em cenários reais, os dados disponíveis nos diferentes clientes frequentemente não seguem a mesma distribuição, caracterizando um ambiente não independente e não identicamente distribuído (non-IID). Essa heterogeneidade pode levar à divergência entre os modelos locais, fenômeno conhecido como *client drift*, dificultando a convergência do modelo global e impactando negativamente o desempenho.

4.3.3.2. Custo de Comunicação

O Aprendizado Federado depende da troca iterativa de parâmetros entre clientes e servidor. Em ambientes com recursos limitados, como dispositivos móveis ou redes com baixa largura de banda, o custo de comunicação pode se tornar um gargalo significativo. Estratégias como redução de frequência de comunicação, compressão de modelos e quantização são frequentemente utilizadas para mitigar esse problema.

 ©2026 SBC – Soc. Bras. de Computação

4.3.3.3. Convergência do Modelo

A convergência do FedAvg pode ser mais lenta ou instável em comparação com métodos centralizados, especialmente em cenários com alta heterogeneidade de dados e diferentes capacidades computacionais entre os clientes. A escolha adequada de hiperparâmetros, como taxa de aprendizado e número de épocas locais, é fundamental para garantir a estabilidade do treinamento.

4.3.3.4. Dependência do Servidor Central

Embora os dados não sejam compartilhados, o servidor central ainda desempenha um papel crítico na agregação dos modelos. Isso implica a necessidade de um nível de confiança nessa entidade e pode representar um ponto único de falha ou ataque.

4.3.3.5. Privacidade e Segurança

Apesar de reduzir o risco de exposição de dados brutos, o Aprendizado Federado não é totalmente imune a ataques. Técnicas de inferência podem, em alguns casos, extrair informações sensíveis a partir dos parâmetros compartilhados. Por essa razão, abordagens complementares, como privacidade diferencial e criptografia segura, são frequentemente incorporadas ao processo de treinamento.

4.4. Visão Geral do Processo: Do Tratamento dos Dados até o Compartilhamento do Modelo Global

O desenvolvimento de sistemas inteligentes para análise de lâminas citopatológicas digitais (*Whole Slide Images* – WSI), no contexto do exame de Papanicolaou, envolve um pipeline estruturado que integra técnicas de processamento de imagens, aprendizado profundo e aprendizado federado. Esse fluxo é motivado pelas características intrínsecas das WSIs, que apresentam resolução em escala gigapixel, tornando inviável seu processamento direto por modelos de aprendizado de máquina. Dessa forma, torna-se necessário decompor o problema em etapas bem definidas, capazes de transformar dados brutos em representações computacionais eficientes e informativas.

A etapa inicial do pipeline consiste no pré-processamento das lâminas digitais, conforme ilustrado na Figura 4.3. Nessa fase, cada WSI é subdividida em pequenas regiões denominadas *patches*, com dimensões fixas, como 512×512 pixels. Essa abordagem reduz significativamente a complexidade computacional do problema, ao mesmo tempo em que preserva padrões locais relevantes para a análise citopatológica. Adicionalmente, regiões sem conteúdo celular, como áreas em branco ou com baixa densidade de informação, são removidas, resultando em um conjunto de dados mais compacto e adequado ao treinamento de modelos de aprendizado profundo.

Após a extração inicial dos patches, realiza-se uma etapa de seleção baseada em critérios estatísticos, conforme apresentado na Figura 4.4. Nem todos os patches contribuem igualmente para o aprendizado do modelo, sendo comum a presença de regiões

 ©2026 SBC – Soc. Bras. de Computação

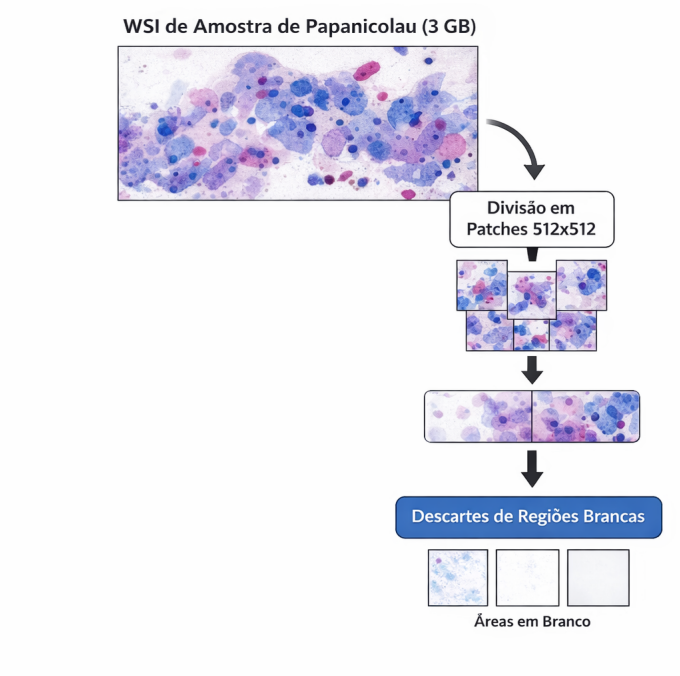


Figura 4.3. Pipeline de pré-processamento de lâminas citológicas digitais (WSI). A imagem completa é subdividida em patches e regiões sem conteúdo celular são descartadas, reduzindo o volume de dados e aumentando a eficiência do processamento.

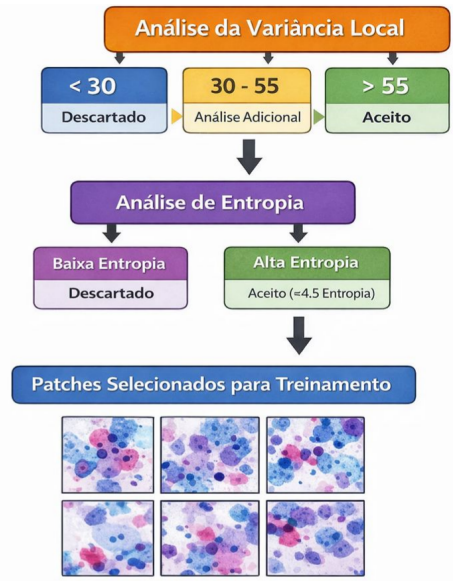


Figura 4.4. Seleção de patches baseada em variância local e entropia. Regiões com baixo conteúdo informacional são descartadas, enquanto patches mais informativos são selecionados para o treinamento.

com baixa variabilidade visual ou conteúdo irrelevante. Para mitigar esse problema, são utilizadas métricas como variância local e entropia, que permitem quantificar o nível de informação presente em cada região. Patches com baixa complexidade são descartados, enquanto regiões com maior riqueza estrutural, tipicamente associadas à presença de cé-

 ©2026 SBC – Soc. Bras. de Computação

lulas e possíveis alterações morfológicas, são mantidas. Esse processo contribui para aumentar a densidade informacional do conjunto de dados e reduzir o impacto de ruídos no treinamento.

Uma vez definidos os patches relevantes, inicia-se o processamento local dos dados e o treinamento dos modelos de aprendizado profundo em cada instituição participante, conforme ilustrado na Figura 4.5. Esse processamento inclui a organização dos dados, normalização das imagens e treinamento de redes neurais convolucionais. Cada instituição atua como um cliente independente, operando sobre seus próprios dados, sem a necessidade de compartilhamento direto de informações sensíveis.

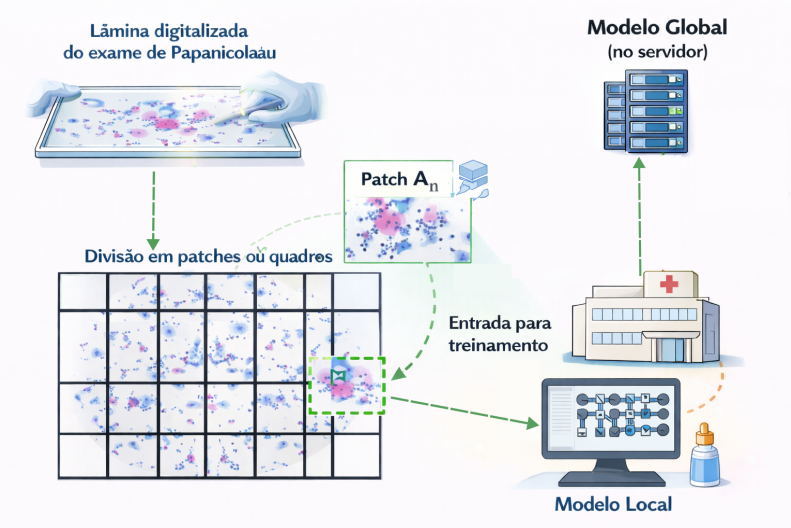


Figura 4.5. Pipeline de processamento local e treinamento em ambiente federado. Cada instituição realiza o treinamento de modelos com seus próprios dados e compartilha apenas os parâmetros aprendidos.

Após o treinamento local, os dados originais permanecem nas instituições de origem, e apenas os parâmetros do modelo são compartilhados. Conforme ilustrado na Figura 4.6, esses parâmetros representam padrões visuais aprendidos ao longo do treinamento, codificados numericamente na forma de pesos da rede neural. Esses pesos não contêm informações explícitas sobre os dados originais, o que contribui para a preservação da privacidade e atende aos requisitos regulatórios associados ao uso de dados médicos.

Na etapa seguinte, os parâmetros dos modelos locais são enviados a um servidor central, onde ocorre o processo de agregação, conforme apresentado na Figura 4.7. Nesse processo, os modelos treinados de forma independente são combinados por meio de algoritmos como o *Federated Averaging*, que realiza uma média ponderada dos parâmetros recebidos. O resultado é um modelo global que incorpora o conhecimento distribuído entre múltiplas instituições.

Após a agregação, o modelo global atualizado é redistribuído às instituições participantes, iniciando uma nova rodada de treinamento local. Esse processo iterativo continua até que o modelo atinja estabilidade ou desempenho satisfatório. Esse mecanismo permite explorar dados distribuídos e heterogêneos, aumentando a capacidade de genera-

 ©2026 SBC – Soc. Bras. de Computação

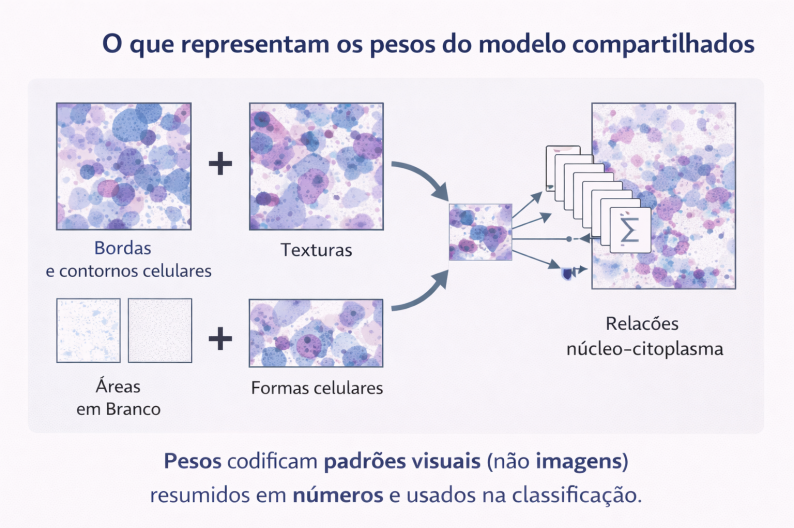


Figura 4.6. Representação conceitual dos parâmetros aprendidos por modelos de aprendizado profundo. Os pesos codificam padrões visuais relevantes sem armazenar diretamente os dados originais.

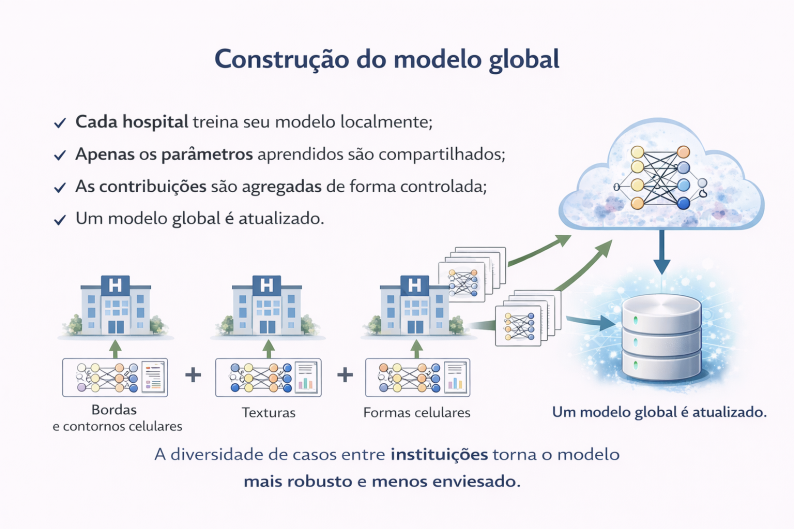


Figura 4.7. Processo de agregação no aprendizado federado. Os parâmetros dos modelos locais são combinados para formar um modelo global sem acesso aos dados originais.

lização do modelo e reduzindo o impacto de variações entre diferentes contextos clínicos.

Ao final desse processo, o modelo global pode ser utilizado como ferramenta de apoio à decisão clínica, auxiliando especialistas na triagem citopatológica. A combinação entre aprendizado profundo e aprendizado federado permite desenvolver soluções robustas, escaláveis e compatíveis com requisitos de privacidade, sendo especialmente adequada para aplicações em saúde que envolvem múltiplas instituições e dados sensíveis.

Em síntese, o pipeline apresentado integra de forma coerente as etapas de pré-processamento, seleção de dados, aprendizado local e agregação federada, constituindo

 ©2026 SBC – Soc. Bras. de Computação

uma abordagem eficiente para análise de WSIs em cenários reais de diagnóstico assistido por inteligência artificial.

4.5. Implementação Prática em Python: Simulação Federada com CNN

Esta seção apresenta a implementação prática de uma simulação de aprendizado federado aplicada à classificação de imagens citopatológicas. O objetivo é mostrar, de forma progressiva e didática, como diferentes tecnologias podem ser integradas para construir um ambiente distribuído, reprodutível e compatível com o treinamento de Redes Neurais Convolucionais (CNNs).

A implementação proposta combina três elementos principais. O primeiro é o Docker, utilizado para padronizar o ambiente de execução e permitir que servidor e clientes sejam executados em contêineres isolados. O segundo é a biblioteca Flower, responsável por coordenar o processo de aprendizado federado, incluindo a comunicação entre clientes e servidor e a agregação dos modelos locais. O terceiro é o PyTorch, utilizado para a construção, treinamento e avaliação da rede neural convolucional empregada no experimento.

Ao longo desta seção, o leitor será conduzido desde os conceitos fundamentais das ferramentas utilizadas até a organização prática do projeto. Inicialmente, discutimos o papel do Docker, do *Dockerfile* e do arquivo `docker-compose.yaml` na criação da arquitetura distribuída. Em seguida, apresentamos o conjunto de dados utilizado, sua divisão entre diferentes clientes e a forma como os volumes Docker permitem simular instituições distintas. Por fim, descrevemos a implementação dos clientes federados e do servidor, destacando as principais funções responsáveis pelo carregamento dos dados, treinamento local, agregação dos parâmetros, avaliação global e registro do histórico de treinamento.

Essa organização permite compreender o fluxo completo da simulação federada: cada cliente acessa apenas seus próprios dados, treina localmente uma CNN e envia ao servidor somente os parâmetros atualizados do modelo. O servidor, por sua vez, agrega essas atualizações e constrói um modelo global, sem acessar diretamente as imagens dos clientes. Dessa forma, a implementação reflete os princípios centrais do aprendizado federado: colaboração entre instituições, preservação da privacidade, reprodutibilidade experimental e escalabilidade da solução.

4.5.1. Docker: Conceitos Fundamentais e Aplicação no Projeto

O Docker é uma plataforma de virtualização leve baseada em contêineres que permite empacotar aplicações e dependências em ambientes isolados e reprodutíveis. Diferentemente de máquinas virtuais completas, os contêineres compartilham o kernel do sistema operacional, tornando sua execução mais eficiente.

Neste minicurso, o Docker é utilizado para padronizar o ambiente de execução de experimentos com CNNs e Aprendizado Federado. Como esses sistemas dependem de múltiplas bibliotecas e configurações específicas, o Docker permite encapsular todo o ambiente em uma única imagem, garantindo consistência entre diferentes máquinas.

Uma imagem Docker é definida por um *Dockerfile*, que descreve a instalação de

 ©2026 SBC – Soc. Bras. de Computação

dependências e a configuração do ambiente. A partir dessa imagem, são instanciados os contêineres executáveis.

No cenário federado, o Docker garante que todos os clientes utilizem o mesmo ambiente, evitando inconsistências. Além disso, permite simular múltiplos clientes em uma única máquina, facilitando testes controlados.

Outra vantagem é a facilidade de implantação: contêineres podem ser rapidamente distribuídos e executados em diferentes instituições. Isso também contribui para a reprodutibilidade científica, permitindo que experimentos sejam replicados sob as mesmas condições.

4.5.2. Biblioteca Flower para Aprendizado Federado

A biblioteca *Flower* é um framework em Python para desenvolvimento e execução de sistemas de Aprendizado Federado, permitindo o treinamento distribuído sem compartilhamento de dados brutos.

Neste minicurso, a Flower é utilizada para implementar o algoritmo *Federated Averaging (FedAvg)*. A biblioteca abstrai a comunicação entre servidor e clientes, permitindo foco na modelagem e nos experimentos.

Sua arquitetura segue o modelo cliente-servidor:

- **Servidor**: coordena o treinamento, distribui o modelo global e agrega as atualizações;
- **Cliente**: executa o treinamento local e retorna os parâmetros atualizados.

A Flower é compatível com frameworks como PyTorch e TensorFlow, permitindo adaptação de modelos existentes. Também suporta simulações locais e execuções distribuídas reais.

Outro diferencial é a flexibilidade na definição de estratégias de agregação, possibilitando lidar com desafios como dados não-IID e heterogeneidade entre clientes.

No contexto de imagens citopatológicas, a Flower permite simular múltiplas instituições colaborando sem compartilhar dados sensíveis, atendendo requisitos de privacidade como a LGPD.

Além disso, sua integração com Docker facilita a criação de ambientes reprodutíveis e distribuíveis.

4.5.3. PyTorch: Framework para Aprendizado Profundo

O *PyTorch* é um dos principais frameworks de aprendizado profundo, amplamente utilizado pela sua flexibilidade e integração com Python.

Uma de suas principais características é o uso de grafos computacionais dinâmicos (*define-by-run*), que facilita depuração e experimentação.

Os dados são representados por tensores com suporte a GPU, essenciais para o treinamento eficiente de modelos de visão computacional.

 ©2026 SBC – Soc. Bras. de Computação

Modelos são definidos por meio da classe `nn.Module`, permitindo a construção modular de redes neurais. Camadas convolucionais, funções de ativação e operações de *pooling* são facilmente integradas.

O treinamento envolve propagação direta, cálculo da perda e retropropagação, com otimização via algoritmos como SGD ou Adam, utilizando o mecanismo automático de gradientes (*autograd*).

No aprendizado federado, o PyTorch é utilizado para implementar os modelos locais treinados em cada cliente. Cada cliente treina seu modelo com dados próprios e compartilha apenas os parâmetros com o servidor.

O framework também oferece ferramentas como `DataLoader` e a biblioteca `torchvision`, que inclui modelos pré-treinados e utilidades para processamento de imagens.

Dessa forma, o PyTorch fornece a base para implementação das arquiteturas de redes neurais utilizadas neste projeto.

4.5.4. Ambiente de Desenvolvimento

O desenvolvimento de sistemas modernos de aprendizado de máquina, especialmente em cenários complexos como o aprendizado federado aplicado a imagens citopatológicas, exige ambientes computacionais consistentes, reprodutíveis e escaláveis. Nesse contexto, o uso de contêineres surge como uma solução fundamental para garantir que aplicações sejam executadas de forma padronizada em diferentes máquinas e infraestruturas.

O Docker é uma plataforma que permite empacotar aplicações e todas as suas dependências em unidades chamadas *containers*. Esses containers são ambientes isolados que incluem bibliotecas, frameworks, configurações e código necessários para a execução do sistema. Dessa forma, elimina-se o problema clássico de incompatibilidade entre ambientes (por exemplo, diferenças de versões de bibliotecas ou sistemas operacionais), garantindo que o software funcione da mesma forma em qualquer máquina.

No contexto deste projeto, que envolve processamento de imagens de lâminas digitais (WSI), redes neurais convolucionais e aprendizado federado, o Docker desempenha um papel essencial. Isso ocorre porque o pipeline envolve múltiplos componentes (pré-processamento, treinamento local, agregação global, etc.), frequentemente executados em diferentes instituições. Assim, o uso de containers garante reprodutibilidade experimental, facilidade de implantação e escalabilidade da infraestrutura.

Além disso, ao considerar que imagens WSI possuem dimensões extremamente elevadas (podendo atingir resoluções de gigapixels), torna-se inviável processá-las diretamente sem uma infraestrutura adequada. Nesse cenário, o Docker permite organizar pipelines eficientes e distribuídos, facilitando o gerenciamento dos recursos computacionais necessários.

 ©2026 SBC – Soc. Bras. de Computação

4.5.4.1. Dockerfile

O *Dockerfile* é um arquivo de texto que contém um conjunto de instruções utilizadas para construir uma imagem Docker. Essa imagem funciona como um "molde" que define como o container será criado e executado.

No contexto deste projeto, o *Dockerfile* é fundamental para garantir a padronização do ambiente tanto do cliente quanto do servidor na arquitetura de aprendizado federado. Isso assegura reprodutibilidade, facilidade de implantação e consistência entre diferentes máquinas e instituições.

De forma geral, um *Dockerfile* descreve:

- a imagem base do sistema;
- a organização do ambiente interno;
- a instalação de dependências;
- a cópia dos arquivos da aplicação;
- o comando de inicialização do container.

Dockerfile do cliente O código a seguir apresenta o *Dockerfile* utilizado no cliente:

```
FROM python:3.11
WORKDIR /app

ENV PYTHONDONTWRITEBYTECODE=1
ENV PYTHONUNBUFFERED=1

COPY requirements.txt /app/requirements.txt

RUN pip install --upgrade pip && \
    pip install --no-cache-dir -r /app/requirements.txt

COPY . /app

CMD ["python", "app.py"]
```

A instrução `FROM` define a imagem base do container. No caso, utiliza-se `python:3.11`, uma imagem oficial que já contém o interpretador Python instalado, servindo como ponto de partida para o ambiente da aplicação.

A linha `WORKDIR` define o diretório de trabalho dentro do container. Ao utilizar `/app`, todos os comandos subsequentes passam a ser executados nesse diretório, o que organiza melhor a estrutura do projeto.

 ©2026 SBC – Soc. Bras. de Computação

As instruções **ENV** configuram variáveis de ambiente importantes. A variável `PYTHONDONTWRITEBYTECODE=1` impede a geração de arquivos `.pyc`, evitando arquivos desnecessários no container. Já `PYTHONUNBUFFERED=1` garante que a saída do Python seja exibida imediatamente no terminal, facilitando o monitoramento e a depuração.

A instrução **COPY** copia o arquivo `requirements.txt` da máquina local para o container. Esse arquivo contém todas as dependências Python necessárias para o funcionamento da aplicação.

O comando **RUN** executa instruções durante o processo de construção (*build*) da imagem Docker. Inicialmente, o comando `pipinstall--upgradepip` atualiza o gerenciador de pacotes do Python para uma versão mais recente, garantindo compatibilidade com as bibliotecas utilizadas no projeto. Em seguida, o comando *pip install –no-cache-dir -r /app/requirements.txt* instala todas as dependências listadas no arquivo `requirements.txt`, evitando o armazenamento de arquivos temporários de cache, o que contribui para a redução do tamanho final da imagem.

No contexto deste projeto, as principais dependências instaladas são:

- `flwr==1.8.0`: biblioteca *Flower*, responsável por implementar a infraestrutura de aprendizado federado, permitindo a comunicação entre servidor e clientes e a execução de estratégias como o FedAvg;
- `numpy==1.26.4`: biblioteca fundamental para computação numérica em Python, amplamente utilizada para manipulação eficiente de arrays e operações matemáticas vetorizadas;
- `torch==2.2.2`: framework de aprendizado profundo (*PyTorch*), utilizado para definição, treinamento e avaliação das redes neurais convolucionais empregadas no projeto;
- `torchvision==0.17.2`: extensão do PyTorch voltada para processamento de imagens, fornecendo transformações, datasets e modelos pré-treinados úteis para tarefas de visão computacional;
- `pillow==10.3.0`: biblioteca para manipulação de imagens em Python, utilizada para leitura, conversão e pré-processamento das imagens de entrada.

Em conjunto, essas dependências compõem a base tecnológica necessária para a execução do pipeline de aprendizado federado com redes neurais convolucionais, incluindo desde o processamento de dados até o treinamento distribuído dos modelos.

A instrução **COPY** `. /app` copia todo o conteúdo do projeto para dentro do container, incluindo o código-fonte da aplicação.

Por fim, a instrução **CMD** define o comando que será executado quando o container for iniciado. Nesse caso, o comando `pythonapp.py` inicia a aplicação cliente.

 ©2026 SBC – Soc. Bras. de Computação

Dockerfile do servidor As instruções iniciais são idênticas às do cliente, pois tanto cliente quanto servidor utilizam o mesmo ambiente base em Python e seguem o mesmo processo de instalação de dependências e organização do código.

A principal diferença está na instrução `EXPOSE`. Essa linha indica que o container utilizará a porta `8080` para comunicação. Embora não publique automaticamente a porta, essa instrução documenta a intenção da aplicação e facilita a integração com ferramentas como o *docker-compose*.

A instrução final `CMD` mantém o mesmo comportamento, iniciando o servidor através do comando *python app.py*.

Síntese Os *Dockerfiles* apresentados seguem uma estrutura simples e eficiente: definem uma base Python, configuram o ambiente, instalam dependências, copiam o código e iniciam a aplicação. Essa padronização é essencial no contexto de aprendizado federado, pois garante que todos os nós (clientes e servidor) operem sob condições controladas e reprodutíveis, reduzindo problemas de incompatibilidade e facilitando a implantação distribuída.

4.5.4.2. Arquivo docker-compose.yaml

O arquivo `docker-compose.yaml` é utilizado para definir, configurar e gerenciar aplicações compostas por múltiplos containers de forma integrada. Enquanto o *Dockerfile* é responsável por descrever como uma imagem individual deve ser construída, o *docker-compose* atua em um nível superior, permitindo a orquestração de diversos serviços que, em conjunto, compõem um sistema distribuído.

No contexto deste projeto de aprendizado federado, essa ferramenta desempenha um papel central, pois permite representar toda a arquitetura distribuída de forma declarativa, clara e reprodutível. Em particular, o arquivo define explicitamente:

- um container responsável pelo servidor federado, que coordena o treinamento global;
- múltiplos containers representando os clientes (instituições participantes), cada um com seus próprios dados;
- a rede de comunicação entre os diferentes componentes do sistema;
- os volumes de dados locais, garantindo isolamento entre os clientes.

O uso do `docker-compose.yaml` traz diversas vantagens práticas:

- permite inicializar toda a infraestrutura com um único comando;
- garante isolamento e independência entre os clientes;
- facilita a simulação de cenários reais de aprendizado federado com dados distribuídos;

 ©2026 SBC – Soc. Bras. de Computação

- promove reprodutibilidade experimental e padronização do ambiente.

O código a seguir apresenta a configuração utilizada neste projeto:

```yaml
services:
  fl-server:
    build: ./server
    container_name: fl-server
    volumes:
      - ./server:/app
      - ./data_sbcas/data/global_test:/app/global_test
    ports:
      - "8080:8080"
    networks:
      - fl-net

  fl-client-1:
    build: ./client
    container_name: fl-client-1
    volumes:
      - ./client:/app
      - ./data_sbcas/data/client_1:/app/data
    depends_on:
      - fl-server
    networks:
      - fl-net

  fl-client-2:
    build: ./client
    container_name: fl-client-2
    volumes:
      - ./client:/app
      - ./data_sbcas/data/client_2:/app/data
    depends_on:
      - fl-server
    networks:
      - fl-net

  fl-client-3:
    build: ./client
    container_name: fl-client-3
    volumes:
      - ./client:/app
      - ./data_sbcas/data/client_3:/app/data
    depends_on:
      - fl-server
    networks:
      - fl-net

```

 ©2026 SBC – Soc. Bras. de Computação

```
46 networks:
47   fl-net:
48     driver: bridge
```

A diretiva `services` define os diferentes serviços que compõem a aplicação. Cada serviço corresponde a um container que será instanciado durante a execução do sistema.

O serviço denominado `fl-server` representa o servidor central do aprendizado federado. A instrução `build` aponta para o diretório `./server`, onde se encontra o respectivo *Dockerfile*. O campo `container_name` atribui um nome explícito ao container, facilitando sua identificação durante a execução e depuração.

A seção `volumes` é responsável por mapear diretórios da máquina local para dentro do container. No caso do servidor, o diretório `./server` é associado a `/app`, garantindo que o código-fonte esteja acessível internamente. Além disso, o diretório `./data_sbcas/data/global_test` é mapeado para `/app/global_test`, permitindo acesso a dados globais utilizados em avaliação.

A diretiva `ports` realiza o mapeamento de portas entre o host e o container. O valor `"8080:8080"` indica que a porta 8080 da máquina local será conectada à porta 8080 do container, tornando o servidor acessível externamente.

A seção `networks` conecta o serviço à rede virtual `fl-net`, permitindo comunicação com os demais containers.

Os serviços `fl-client-1`, `fl-client-2` e `fl-client-3` representam os diferentes clientes do sistema federado. Todos utilizam a mesma configuração base, sendo construídos a partir do diretório `./client`. Essa padronização garante que todos os clientes operem sob o mesmo ambiente computacional.

Cada cliente possui um nome distinto, definido por `container_name`, permitindo sua identificação individual. A principal diferença entre eles está nos volumes de dados associados. Por exemplo, cada cliente acessa um diretório específico, como `./data_sbcas/data/client_1`, `./data_sbcas/data/client_2` e `./data_sbcas/data/client_3`. Essa separação é essencial para simular o cenário real de aprendizado federado, no qual cada instituição mantém seus próprios dados localmente, sem compartilhamento direto.

A diretiva `depends_on` indica que os clientes dependem do serviço `fl-server`. Isso garante que o servidor seja inicializado antes dos clientes, assegurando o funcionamento correto da comunicação entre os nós.

Todos os serviços estão conectados à mesma rede `fl-net`, o que permite a comunicação direta entre servidor e clientes de forma transparente.

Por fim, a seção `networks` define a rede utilizada pelos containers. O driver `bridge` é o padrão do Docker para redes locais, permitindo que os containers se comuniquem como se estivessem em uma rede privada isolada.

 ©2026 SBC – Soc. Bras. de Computação

Síntese O arquivo `docker-compose.yaml` implementa uma arquitetura completa de aprendizado federado em ambiente controlado. O servidor atua como coordenador global do treinamento, enquanto os clientes realizam o aprendizado local utilizando seus próprios dados. A separação via volumes garante o isolamento dos dados, e a rede compartilhada possibilita a troca de informações entre os nós.

Toda essa infraestrutura pode ser inicializada de forma simples com o comando:

```
docker-compose up
```

Essa abordagem torna o ambiente altamente reprodutível, escalável e adequado para experimentos distribuídos, aproximando o cenário experimental de aplicações reais em ambientes multicêntricos.

4.5.5. Dados Utilizados no Projeto

Neste projeto, foi utilizado o dataset *Single Cell Conventional Pap Smear Images*, composto por imagens citológicas individuais de exames de Papanicolau.

Para simular um cenário de aprendizado federado, os dados foram organizados de forma distribuída, utilizando volumes Docker associados a diferentes clientes. Cada cliente acessa apenas seu subconjunto local de dados, enquanto o conjunto de teste é mantido separado na pasta `global_test`, utilizada para avaliação do modelo global.

4.5.5.1. Organização e Particionamento dos Dados

O dataset foi estruturado em dois conjuntos principais: *Training* e *Test*. O conjunto de treinamento foi dividido entre três clientes, enquanto o conjunto de teste foi mantido íntegro para avaliação global.

O processo de preparação dos dados consiste em:

- validação da estrutura do dataset (consistência entre classes de treino e teste);
- criação dos diretórios dos clientes e do conjunto `global_test`;
- cópia dos dados de teste para avaliação global;
- divisão dos dados de treinamento entre os clientes.

A divisão dos dados de treinamento é realizada de forma balanceada por classe. Para cada classe, as imagens são embaralhadas e distribuídas entre os clientes:

```
random.shuffle(images)
client_splits = split_evenly(images, NUM_CLIENTS)

for client_idx, split_imgs in enumerate(client_splits, start=1):
```

 ©2026 SBC – Soc. Bras. de Computação

```
    for img_path in split_imgs:
        dst = OUTPUT_DIR / f"client_{client_idx}" / cls /
         ↪  img_path.name
        safe_link_or_copy(img_path, dst, COPY_FILES)
```

Essa estratégia garante que cada cliente possua uma amostra representativa de todas as classes, mantendo uma distribuição aproximadamente balanceada.

Ao final, a estrutura dos dados segue o padrão esperado pela arquitetura Docker, em que cada cliente acessa seu diretório local (`/app/data`) e o servidor utiliza o conjunto `/app/global_test` para avaliação do modelo global.

Síntese A organização adotada permite simular um ambiente federado controlado, no qual os dados permanecem distribuídos entre os clientes, enquanto a avaliação do modelo é realizada de forma centralizada e consistente.

4.5.6. Implementação dos Clientes Federados

No contexto do aprendizado federado, os clientes são responsáveis por executar o treinamento local do modelo utilizando apenas os dados disponíveis em sua própria instituição. Diferentemente de uma abordagem centralizada, os dados brutos não são enviados ao servidor. Cada cliente treina localmente, calcula atualizações nos parâmetros do modelo e compartilha apenas esses parâmetros com o servidor federado.

No presente projeto, os clientes são implementados a partir do mesmo código-fonte, sendo diferenciados apenas pelo volume de dados montado no container Docker. Assim, os serviços `fl-client-1`, `fl-client-2` e `fl-client-3` simulam instituições distintas, cada uma com seu próprio conjunto local de imagens.

4.5.6.1. Configuração e Hiperparâmetros

A primeira parte do código define os principais hiperparâmetros utilizados pelo cliente federado. Esses valores controlam desde o local dos dados até o número de épocas locais executadas em cada rodada federada.

```
DATA_DIR = Path("/app/data")
SERVER_ADDRESS = "fl-server:8080"

BATCH_SIZE = 16
LOCAL_EPOCHS = 1
LEARNING_RATE = 1e-3
VAL_RATIO = 0.2
RANDOM_SEED = 42

NUM_RETRIES = 20
RETRY_SLEEP = 3
```

 ©2026 SBC – Soc. Bras. de Computação

```
INITIAL_SLEEP = 5

DEVICE = torch.device("cuda" if torch.cuda.is_available() else
 ↪  "cpu")
```

A variável `DATA_DIR` indica o diretório local onde os dados do cliente são montados dentro do container. O endereço `SERVER_ADDRESS` define o servidor federado responsável por coordenar o treinamento. O tamanho do lote é definido por `BATCH_SIZE`, enquanto `LOCAL_EPOCHS` determina quantas épocas de treinamento local são executadas a cada rodada federada. A taxa de aprendizado `LEARNING_RATE` controla a magnitude das atualizações realizadas pelo otimizador.

A proporção `VAL_RATIO=0.2` indica que 20% dos dados locais são separados para validação. A semente aleatória `RANDOM_SEED` garante reprodutibilidade na divisão entre treino e validação. Por fim, as variáveis `NUM_RETRIES`, `RETRY_SLEEP` e `INITIAL_SLEEP` controlam o mecanismo de tentativa de conexão com o servidor, importante em ambientes baseados em Docker, nos quais os containers podem iniciar em tempos diferentes.

4.5.6.2. Inspeção dos Dados Locais

Antes de iniciar o treinamento, o cliente verifica se o volume de dados foi corretamente montado. Essa etapa é realizada pela função `inspect_data_dir()`, que identifica as subpastas de classes e contabiliza a quantidade de arquivos em cada uma delas.

```
def inspect_data_dir(data_dir: Path) -> None:
    if not data_dir.exists():
        raise FileNotFoundError(f"Pasta de dados não encontrada:
         ↪  {data_dir}")

    class_dirs = sorted([p for p in data_dir.iterdir() if
     ↪  p.is_dir()],
                        key=lambda x: x.name)

    if not class_dirs:
        raise ValueError(f"Nenhuma subpasta de classe encontrada
         ↪  em {data_dir}")

    total_files = 0

    for class_dir in class_dirs:
        num_files = len([p for p in class_dir.iterdir() if
         ↪  p.is_file()])
        total_files += num_files
```

 ©2026 SBC – Soc. Bras. de Computação

```
        log(f"{class_dir.name}: {num_files} arquivos")

    log(f"Total de arquivos em /app/data: {total_files}")
```

Essa verificação é importante porque cada cliente depende de um volume Docker específico. Caso o volume esteja vazio, incorreto ou sem subpastas de classe, o treinamento não deve prosseguir. Além disso, essa função permite confirmar se a distribuição de dados entre os clientes está correta.

4.5.6.3. Carregamento, Transformações e Divisão dos Dados

A função `get_dataloaders()` é responsável por carregar as imagens, aplicar transformações e criar os carregadores de dados utilizados no treinamento e na validação.

```
train_transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.RandomHorizontalFlip(),
    transforms.RandomRotation(10),
    transforms.ToTensor(),
    transforms.Normalize(
        mean=[0.485, 0.456, 0.406],
        std=[0.229, 0.224, 0.225],
    ),
])
```

No conjunto de treinamento, são aplicadas transformações de aumento de dados (data augmentation), como espelhamento horizontal e rotação aleatória. Essas operações ajudam a reduzir o overfitting e tornam o modelo mais robusto a pequenas variações nas imagens.

Para validação, são utilizadas apenas transformações determinísticas:

```
val_transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
    transforms.Normalize(
        mean=[0.485, 0.456, 0.406],
        std=[0.229, 0.224, 0.225],
    ),
])
```

As imagens são redimensionadas para 224×224, tamanho esperado pela arquitetura ResNet-18. A normalização utiliza as médias e desvios padrão do ImageNet, pois o modelo empregado foi previamente treinado nessa base.

 ©2026 SBC – Soc. Bras. de Computação

O carregamento dos dados é feito com `ImageFolder`, que assume uma estrutura em que cada subpasta representa uma classe:

```
base_dataset = datasets.ImageFolder(root=str(DATA_DIR))
class_names = base_dataset.classes
num_classes = len(class_names)
```

Em seguida, os dados locais são divididos em treino e validação:

```
dataset_size = len(base_dataset)
val_size = max(1, int(dataset_size * VAL_RATIO))
train_size = dataset_size - val_size

generator = torch.Generator().manual_seed(RANDOM_SEED)

train_subset, val_subset = random_split(
    base_dataset,
    [train_size, val_size],
    generator=generator,
)
```

A divisão é feita localmente em cada cliente. Portanto, cada instituição simulada possui sua própria partição de treino e validação. Esse comportamento é coerente com o cenário federado, pois os dados não são centralizados.

Como o conjunto de treino e o conjunto de validação usam transformações diferentes, foi criada uma classe auxiliar chamada `TransformedSubset`:

```
class TransformedSubset(torch.utils.data.Dataset):
    def __init__(self, subset, transform):
        self.subset = subset
        self.transform = transform

    def __len__(self):
        return len(self.subset)

    def __getitem__(self, idx):
        image, label = self.subset[idx]
        image = image.convert("RGB")
        image = self.transform(image)
        return image, label
```

Por fim, são criados os *DataLoaders*:

 ©2026 SBC – Soc. Bras. de Computação

```
train_loader = DataLoader(
    train_dataset,
    batch_size=BATCH_SIZE,
    shuffle=True,
    num_workers=0,
)

val_loader = DataLoader(
    val_dataset,
    batch_size=BATCH_SIZE,
    shuffle=False,
    num_workers=0,
)
```

O `shuffle=True` é utilizado no treino para embaralhar as amostras a cada época. Na validação, o embaralhamento não é necessário, pois o objetivo é apenas avaliar o desempenho do modelo.

4.5.6.4. Construção do Modelo Local

O modelo local é definido pela função `build_model()`. Neste projeto, foi utilizada a arquitetura ResNet-18 com pesos pré-treinados no ImageNet.

```
def build_model(num_classes: int) -> nn.Module:
    weights = models.ResNet18_Weights.DEFAULT
    model = models.resnet18(weights=weights)

    for param in model.parameters():
        param.requires_grad = False

    in_features = model.fc.in_features
    model.fc = nn.Linear(in_features, num_classes)

    return model
```

Inicialmente, todos os parâmetros da rede são congelados. Dessa forma, o treinamento local atualiza apenas a camada final do classificador. Essa estratégia caracteriza uma forma de *transfer learning*, pois reaproveita representações visuais aprendidas em uma grande base de imagens e adapta apenas a saída do modelo ao número de classes do problema.

A camada final original da ResNet-18 é substituída por uma nova camada linear com saída igual ao número de classes identificadas no diretório local.

 ©2026 SBC – Soc. Bras. de Computação

4.5.6.5. Funções de Treinamento e Avaliação

A função `train()` executa uma época de treinamento local. Ela coloca o modelo em modo de treinamento, percorre os lotes de imagens, calcula a perda, realiza a retropropagação e atualiza os parâmetros treináveis.

```
def train(model, loader, criterion, optimizer):
    model.train()
    running_loss = 0.0
    correct = 0
    total = 0

    for images, labels in loader:
        images = images.to(DEVICE)
        labels = labels.to(DEVICE)

        optimizer.zero_grad()
        outputs = model(images)
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()

        running_loss += loss.item() * images.size(0)
        preds = outputs.argmax(dim=1)
        correct += (preds == labels).sum().item()
        total += labels.size(0)

    loss = running_loss / total
    acc = correct / total
    return loss, acc
```

A função de perda utilizada é a entropia cruzada, adequada para problemas de classificação multiclasse:

```
criterion = nn.CrossEntropyLoss()
```

Como apenas a camada final foi descongelada, o otimizador Adam recebe somente os parâmetros de `model.fc`:

```
optimizer = torch.optim.Adam(
    model.fc.parameters(),
    lr=LEARNING_RATE
)
```

 ©2026 SBC – Soc. Bras. de Computação

A avaliação é realizada pela função `evaluate()`, que opera sem cálculo de gradientes:

```
@torch.no_grad()
def evaluate(model, loader, criterion):
    model.eval()
    running_loss = 0.0
    correct = 0
    total = 0

    for images, labels in loader:
        images = images.to(DEVICE)
        labels = labels.to(DEVICE)

        outputs = model(images)
        loss = criterion(outputs, labels)

        running_loss += loss.item() * images.size(0)
        preds = outputs.argmax(dim=1)
        correct += (preds == labels).sum().item()
        total += labels.size(0)

    loss = running_loss / total
    acc = correct / total
    return loss, acc
```

Essa função retorna duas métricas principais: a perda média e a acurácia. No contexto federado, essas métricas são importantes para acompanhar o comportamento de cada cliente ao longo das rodadas.

4.5.6.6. Conversão dos Parâmetros do Modelo

A comunicação entre PyTorch e Flower exige que os parâmetros do modelo sejam convertidos para listas de arrays NumPy. Essa conversão é feita pela função `get_parameters()`.

```
def get_parameters(model) -> List[np.ndarray]:
    return [val.cpu().numpy() for _, val in
     ↪  model.state_dict().items()]
```

Quando o cliente recebe parâmetros globais enviados pelo servidor, eles precisam ser convertidos de volta para tensores PyTorch. Essa etapa é feita pela função `set_parameters()`.

 ©2026 SBC – Soc. Bras. de Computação

```python
def set_parameters(model, parameters: List[np.ndarray]) -> None:
    params_dict = zip(model.state_dict().keys(), parameters)
    state_dict = OrderedDict(
        {k: torch.tensor(v) for k, v in params_dict}
    )
    model.load_state_dict(state_dict, strict=True)
```

Essas duas funções são essenciais para o aprendizado federado, pois permitem que o modelo local receba os pesos globais, realize treinamento local e devolva os pesos atualizados ao servidor.

4.5.6.7. Inicialização do Cliente

Após a definição das funções, o cliente executa a sequência de inicialização. Primeiro, verifica o dispositivo disponível, inspeciona os dados, cria os *DataLoaders*, constrói o modelo e define a função de perda e o otimizador.

```python
inspect_data_dir(DATA_DIR)
train_loader, val_loader, num_classes = get_dataloaders()

model = build_model(num_classes).to(DEVICE)
criterion = nn.CrossEntropyLoss()
optimizer = torch.optim.Adam(
    model.fc.parameters(),
    lr=LEARNING_RATE
)
```

Esse trecho concentra a preparação do cliente antes de sua conexão com o servidor federado.

4.5.6.8. Implementação do Cliente Flower

A integração com o Flower é feita por meio da classe `FlowerClient`, que herda de `fl.client.NumPyClient`. Essa classe implementa três métodos principais: `get_parameters()`, `fit()` e `evaluate()`.

```python
class FlowerClient(fl.client.NumPyClient):
    def get_parameters(self, config):
        return get_parameters(model)
```

O método `get_parameters()` retorna os parâmetros atuais do modelo local para o servidor.

 ©2026 SBC – Soc. Bras. de Computação

O método `fit()` representa a etapa de treinamento local em uma rodada federada:

```
def fit(self, parameters, config):
    set_parameters(model, parameters)

    for epoch in range(LOCAL_EPOCHS):
        train_loss, train_acc = train(
            model,
            train_loader,
            criterion,
            optimizer
        )

    val_loss, val_acc = evaluate(model, val_loader, criterion)

    return get_parameters(model), len(train_loader.dataset), {
        "train_loss": float(train_loss),
        "train_acc": float(train_acc),
        "val_loss": float(val_loss),
        "val_acc": float(val_acc),
    }
```

Inicialmente, o cliente recebe os parâmetros globais enviados pelo servidor e atualiza seu modelo local com `set_parameters()`. Em seguida, realiza o treinamento local por `LOCAL_EPOCHS`. Ao final, avalia o modelo em seu conjunto de validação e retorna três informações ao servidor: os parâmetros atualizados, o número de amostras locais usadas no treinamento e um dicionário com métricas locais.

O método `evaluate()` é utilizado quando o servidor solicita a avaliação do modelo global em cada cliente:

```
def evaluate(self, parameters, config):
    set_parameters(model, parameters)

    loss, acc = evaluate(model, val_loader, criterion)

    return float(loss), len(val_loader.dataset), {
        "accuracy": float(acc)
    }
```

Nesse caso, o cliente recebe os pesos globais, atualiza seu modelo local e calcula a perda e a acurácia no conjunto de validação local.

 ©2026 SBC – Soc. Bras. de Computação

4.5.6.9. Conexão com o Servidor Federado

A conexão com o servidor é feita pela função `fl.client.start_client()`. Antes da conexão, o cliente aguarda alguns segundos para reduzir problemas causados pela inicialização assíncrona dos containers.

```
time.sleep(INITIAL_SLEEP)

for i in range(NUM_RETRIES):
    try:
        fl.client.start_client(
            server_address=SERVER_ADDRESS,
            client=FlowerClient().to_client(),
        )
        break
    except Exception as e:
        log(f"Falha ao conectar: {e}")
        time.sleep(RETRY_SLEEP)
```

Esse mecanismo de repetição é importante porque, em uma arquitetura com Docker Compose, o cliente pode iniciar antes do servidor estar pronto para receber conexões. Assim, o código tenta se conectar várias vezes antes de encerrar.

Síntese A implementação do cliente federado organiza, em um único fluxo, todas as etapas necessárias para o treinamento distribuído: inspeção dos dados locais, carregamento das imagens, aplicação de transformações, construção do modelo, treinamento local, avaliação, conversão de parâmetros e comunicação com o servidor Flower.

A arquitetura adotada utiliza *transfer learning* com ResNet-18, mantendo o extrator de características congelado e treinando apenas a camada final. Essa decisão reduz o custo computacional dos clientes e torna o treinamento mais viável em um ambiente federado. Além disso, a separação dos dados por volumes Docker permite simular um cenário multicêntrico, no qual diferentes instituições colaboram para treinar um modelo global sem compartilhar diretamente suas imagens.

4.5.7. Implementação do Servidor Federado

No aprendizado federado, o servidor atua como o coordenador central do processo de treinamento distribuído. Sua função não é acessar os dados brutos dos clientes, mas organizar as rodadas federadas, receber os parâmetros treinados localmente, agregá-los em um novo modelo global e redistribuir esse modelo atualizado aos clientes.

No presente projeto, o servidor foi implementado com o framework Flower, utilizando uma estratégia customizada baseada no algoritmo *Federated Averaging* (FedAvg). Além da agregação dos modelos locais, o servidor também realiza avaliação em um conjunto global de teste, salva checkpoints a cada rodada e mantém um histórico completo das métricas de treinamento.

 ©2026 SBC – Soc. Bras. de Computação

4.5.7.1. Configurações e Hiperparâmetros do Servidor

A primeira parte do código define os principais diretórios, arquivos de saída e hiperparâmetros utilizados pelo servidor federado.

```
GLOBAL_TEST_DIR = Path("/app/global_test")
OUTPUT_DIR = Path("/app/output")
CHECKPOINTS_DIR = OUTPUT_DIR / "checkpoints"
HISTORY_JSON = OUTPUT_DIR / "history.json"
HISTORY_CSV = OUTPUT_DIR / "history.csv"

BATCH_SIZE = 16
DEVICE = torch.device("cuda" if torch.cuda.is_available() else
 ↪  "cpu")
```

O diretório `GLOBAL_TEST_DIR` armazena o conjunto de teste global, utilizado exclusivamente para avaliar o modelo agregado após cada rodada federada. Esse conjunto não participa do treinamento dos clientes.

O diretório `OUTPUT_DIR` concentra os arquivos gerados durante a execução, enquanto `CHECKPOINTS_DIR` armazena os modelos salvos ao longo das rodadas. Os arquivos `history.json` e `history.csv` registram o histórico do treinamento em formatos adequados tanto para leitura estruturada quanto para análise tabular.

O hiperparâmetro `BATCH_SIZE=16` define o tamanho dos lotes utilizados na avaliação global. Já a variável `DEVICE` seleciona automaticamente GPU, caso esteja disponível, ou CPU, caso contrário.

```
OUTPUT_DIR.mkdir(parents=True, exist_ok=True)
CHECKPOINTS_DIR.mkdir(parents=True, exist_ok=True)
```

Essas instruções garantem que os diretórios de saída existam antes do início do treinamento federado.

4.5.7.2. Registro do Histórico de Treinamento

O servidor mantém uma lista chamada `history_records`, na qual são armazenadas as métricas de cada rodada federada. Também é mantida a variável `best_global_acc`, responsável por registrar a melhor acurácia obtida no conjunto de teste global.

```
history_records = []
best_global_acc = -1.0
```

 ©2026 SBC – Soc. Bras. de Computação

A função `save_history()` salva esse histórico em dois formatos: `JSON` e `CSV`.

```
def save_history():
    with open(HISTORY_JSON, "w", encoding="utf-8") as f:
        json.dump(history_records, f, indent=2,
         ↪  ensure_ascii=False)

    fieldnames = [
        "round",
        "train_loss",
        "train_acc",
        "val_loss",
        "val_acc",
        "distributed_accuracy",
        "global_loss",
        "global_acc",
    ]

    with open(HISTORY_CSV, "w", newline="", encoding="utf-8") as
     ↪  f:
        writer = csv.DictWriter(f, fieldnames=fieldnames)
        writer.writeheader()
        for row in history_records:
            writer.writerow(row)
```

Esse mecanismo é importante para reprodutibilidade experimental. As métricas salvas permitem analisar a evolução do treinamento, comparar execuções e identificar a melhor rodada federada.

4.5.7.3. Construção do Conjunto Global de Teste

A função `build_global_test_loader()` é responsável por carregar o conjunto de teste global armazenado em `/app/global_test`.

```
def build_global_test_loader():
    if not GLOBAL_TEST_DIR.exists():
        raise FileNotFoundError(
            f"Diretório global_test não encontrado:
             ↪  {GLOBAL_TEST_DIR}"
        )

    transform = transforms.Compose([
        transforms.Resize((224, 224)),
        transforms.ToTensor(),
        transforms.Normalize(
```

 ©2026 SBC – Soc. Bras. de Computação

```
            mean=[0.485, 0.456, 0.406],
            std=[0.229, 0.224, 0.225],
        ),
    ])

    dataset = datasets.ImageFolder(
        root=str(GLOBAL_TEST_DIR),
        transform=transform
    )

    loader = DataLoader(
        dataset,
        batch_size=BATCH_SIZE,
        shuffle=False,
        num_workers=0
    )

    return loader, len(dataset.classes)
```

O conjunto global segue a estrutura esperada pelo `ImageFolder`, em que cada subdiretório representa uma classe. As imagens são redimensionadas para 224×224, convertidas para tensores e normalizadas com as médias e desvios padrão do ImageNet, de forma compatível com a ResNet-18 pré-treinada.

```
GLOBAL_TEST_LOADER, NUM_CLASSES = build_global_test_loader()
```

Essa chamada inicializa o carregador global de teste e identifica automaticamente o número de classes do problema.

4.5.7.4. Construção do Modelo Global

O servidor utiliza a mesma arquitetura dos clientes: uma ResNet-18 pré-treinada. Essa consistência é fundamental, pois os parâmetros enviados pelos clientes precisam ser compatíveis com a estrutura do modelo mantido no servidor.

```
def build_model(num_classes: int) -> nn.Module:
    weights = models.ResNet18_Weights.DEFAULT
    model = models.resnet18(weights=weights)

    for param in model.parameters():
        param.requires_grad = False

    in_features = model.fc.in_features
    model.fc = nn.Linear(in_features, num_classes)
```

 ©2026 SBC – Soc. Bras. de Computação

```
10
11      return model
```

Assim como nos clientes, o backbone convolucional é congelado e apenas a camada final é substituída para se ajustar ao número de classes do conjunto utilizado. Essa abordagem reduz o custo computacional e mantém o processo de treinamento federado mais leve.

4.5.7.5. Atualização dos Parâmetros Globais

Após a agregação federada, os parâmetros do modelo são recebidos em formato NumPy. Para reconstruir o modelo em PyTorch, utiliza-se a função `set_parameters()`.

```
1  def set_parameters(model: nn.Module, parameters:
    ↪  List[np.ndarray]) -> None:
2      params_dict = zip(model.state_dict().keys(), parameters)
3      state_dict = OrderedDict({
4          k: torch.tensor(v) for k, v in params_dict
5      })
6      model.load_state_dict(state_dict, strict=True)
```

Essa função associa cada array recebido ao nome correspondente no `state_dict` do modelo. Em seguida, os pesos são carregados com `strict=True`, garantindo que a estrutura dos parâmetros recebidos seja compatível com a arquitetura esperada.

4.5.7.6. Salvamento de Checkpoints

A função `save_model_checkpoint()` salva os pesos do modelo global em disco.

```
1  def save_model_checkpoint(parameters: List[np.ndarray],
    ↪  filepath: Path):
2      model = build_model(NUM_CLASSES)
3      set_parameters(model, parameters)
4      torch.save(model.state_dict(), filepath)
5      print(f"Checkpoint salvo em: {filepath}", flush=True)
```

A cada rodada, o servidor salva um checkpoint do modelo global agregado. Além disso, quando a acurácia global melhora, o modelo também é salvo como `best_model.pt`. Isso permite recuperar tanto modelos intermediários quanto o melhor modelo obtido durante a execução.

 ©2026 SBC – Soc. Bras. de Computação

4.5.7.7. Avaliação do Modelo Global

A avaliação global é realizada pela função `evaluate_global_model()`. Ela calcula a perda e a acurácia do modelo agregado utilizando o conjunto `global_test`.

```
@torch.no_grad()
def evaluate_global_model(model: nn.Module, loader: DataLoader):
    criterion = nn.CrossEntropyLoss()
    model.eval()

    running_loss = 0.0
    correct = 0
    total = 0

    for images, labels in loader:
        images = images.to(DEVICE)
        labels = labels.to(DEVICE)

        outputs = model(images)
        loss = criterion(outputs, labels)

        running_loss += loss.item() * images.size(0)
        preds = outputs.argmax(dim=1)
        correct += (preds == labels).sum().item()
        total += labels.size(0)

    global_loss = running_loss / total
    global_acc = correct / total

    return global_loss, global_acc
```

O uso de `@torch.no_grad()` evita o cálculo de gradientes durante a avaliação, reduzindo o consumo de memória e tornando a inferência mais eficiente. Essa avaliação é importante porque fornece uma medida independente do desempenho do modelo global, sem depender exclusivamente das validações locais dos clientes.

4.5.7.8. Estratégia Federada Customizada

O núcleo do servidor está na classe `FedAvgWithGlobalTest`, que herda da estratégia padrão `FedAvg` do Flower.

```
class FedAvgWithGlobalTest(fl.server.strategy.FedAvg):
    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)
        self.latest_parameters = None
        self.latest_fit_metrics = {}
```

 ©2026 SBC – Soc. Bras. de Computação

Essa classe adiciona duas variáveis internas. A primeira, `latest_parameters`, armazena os parâmetros globais mais recentes após a agregação. A segunda, `latest_fit_metrics`, guarda as métricas agregadas de treinamento enviadas pelos clientes.

Agregação dos Modelos Locais O método `aggregate_fit()` é executado após os clientes realizarem o treinamento local e enviarem seus parâmetros atualizados ao servidor.

```
def aggregate_fit(self, server_round, results, failures):
    aggregated_parameters, aggregated_metrics =
     ↪  super().aggregate_fit(
        server_round,
        results,
        failures
    )

    if aggregated_parameters is not None:
        self.latest_parameters = aggregated_parameters

        ndarrays = parameters_to_ndarrays(aggregated_parameters)
        round_ckpt = CHECKPOINTS_DIR /
         ↪  f"round_{server_round:03d}.pt"
        save_model_checkpoint(ndarrays, round_ckpt)

    self.latest_fit_metrics = aggregated_metrics or {}

    return aggregated_parameters, aggregated_metrics
```

A chamada `super().aggregate_fit()` executa a agregação FedAvg original do Flower. Em seguida, os parâmetros agregados são armazenados e convertidos para arrays NumPy por meio da função `parameters_to_ndarrays()`.

Cada rodada gera um checkpoint com nome padronizado, como `round_001.pt`, `round_002.pt` e assim por diante. Dessa forma, é possível recuperar o estado do modelo global em qualquer rodada do treinamento.

Avaliação Agregada e Avaliação Global O método `aggregate_evaluate()` é executado após a etapa de avaliação distribuída dos clientes. Além de agregar as métricas locais, esse método reconstrói o modelo global e o avalia no conjunto `global_test`.

```
def aggregate_evaluate(self, server_round, results, failures):
    global best_global_acc

```

 ©2026 SBC – Soc. Bras. de Computação

```
    aggregated_loss, aggregated_metrics = super().aggregate_evaluate(
        server_round,
        results,
        failures
    )

    if aggregated_metrics is None:
        aggregated_metrics = {}

    global_loss = None
    global_acc = None

    if self.latest_parameters is not None:
        global_ndarrays = parameters_to_ndarrays(self.latest_parameters)
        model = build_model(NUM_CLASSES).to(DEVICE)
        set_parameters(model, global_ndarrays)

        global_loss, global_acc = evaluate_global_model(
            model,
            GLOBAL_TEST_LOADER
        )

        aggregated_metrics["global_loss"] = float(global_loss)
        aggregated_metrics["global_acc"] = float(global_acc)
```

Esse trecho reconstrói o modelo global a partir dos parâmetros agregados na rodada atual. Em seguida, calcula a perda e a acurácia no conjunto global de teste. Essas métricas são adicionadas ao dicionário de métricas agregadas.

O código também verifica se o modelo atual é o melhor obtido até o momento:

```
if global_acc > best_global_acc:
    best_global_acc = global_acc
    best_ckpt = CHECKPOINTS_DIR / "best_model.pt"
    save_model_checkpoint(global_ndarrays, best_ckpt)
```

Caso a acurácia global da rodada atual seja superior à melhor acurácia anterior, o modelo é salvo como `best_model.pt`. Essa estratégia facilita a seleção do melhor modelo final.

Ao final da rodada, as métricas são registradas no histórico:

```
record = {
    "round": server_round,
```


©2026 SBC – Soc. Bras. de Computação

```
    "train_loss":
     ↪  float(self.latest_fit_metrics.get("train_loss",
     ↪  np.nan)),
    "train_acc": float(self.latest_fit_metrics.get("train_acc",
     ↪  np.nan)),
    "val_loss": float(self.latest_fit_metrics.get("val_loss",
     ↪  np.nan)),
    "val_acc": float(self.latest_fit_metrics.get("val_acc",
     ↪  np.nan)),
    "distributed_accuracy": float(
        aggregated_metrics.get("accuracy", np.nan)
    ),
    "global_loss": float(
        global_loss if global_loss is not None else np.nan
    ),
    "global_acc": float(
        global_acc if global_acc is not None else np.nan
    ),
}

history_records.append(record)
save_history()
```

Esse registro reúne métricas locais agregadas, métricas distribuídas de avaliação e métricas globais independentes. Com isso, o servidor mantém uma visão completa da evolução do treinamento federado.

4.5.7.9. Agregação Ponderada de Métricas

A função `weighted_average()` agrega as métricas enviadas pelos clientes considerando o número de amostras de cada um.

```
def weighted_average(metrics):
    total_examples = sum(num_examples for num_examples, _ in
     ↪  metrics)

    if total_examples == 0:
        return {}

    aggregated = {}
    metric_names = set()

    for _, client_metrics in metrics:
        metric_names.update(client_metrics.keys())
```

 ©2026 SBC – Soc. Bras. de Computação

Inicialmente, a função calcula o número total de amostras utilizadas pelos clientes e identifica quais métricas foram enviadas. Em seguida, calcula a média ponderada de cada métrica.

```
for metric_name in metric_names:
    weighted_sum = 0.0
    valid_examples = 0

    for num_examples, client_metrics in metrics:
        if metric_name in client_metrics:
            weighted_sum += num_examples *
             ↪  float(client_metrics[metric_name])
            valid_examples += num_examples

    if valid_examples > 0:
        aggregated[metric_name] = weighted_sum / valid_examples
```

Essa agregação evita que clientes com poucos dados tenham o mesmo peso estatístico de clientes com muitos dados. Formalmente, para uma métrica m, a agregação ponderada pode ser representada por:

$$\bar{m} = \frac{\sum_{k=1}^{K} n_k m_k}{\sum_{k=1}^{K} n_k},$$

em que n_k representa o número de amostras do cliente k e m_k representa a métrica calculada nesse cliente.

4.5.7.10. Configuração da Estratégia Federada

A estratégia federada é instanciada com parâmetros que definem a participação dos clientes em cada rodada.

```
strategy = FedAvgWithGlobalTest(
    fraction_fit=1.0,
    fraction_evaluate=1.0,
    min_fit_clients=3,
    min_evaluate_clients=3,
    min_available_clients=3,
    fit_metrics_aggregation_fn=weighted_average,
    evaluate_metrics_aggregation_fn=weighted_average,
)
```

O parâmetro `fraction_fit=1.0` indica que todos os clientes disponíveis devem participar do treinamento em cada rodada. De forma semelhante, `fraction_`

 ©2026 SBC – Soc. Bras. de Computação

`evaluate=1.0` indica que todos os clientes disponíveis também devem participar da avaliação.

Os parâmetros `min_fit_clients=3`, `min_evaluate_clients=3` e `min_available_clients=3` definem que o treinamento federado exige três clientes disponíveis. Isso é coerente com a arquitetura do projeto, composta pelos serviços `fl-client-1`, `fl-client-2` e `fl-client-3`.

As funções `fit_metrics_aggregation_fn` e `evaluate_metrics_aggregation_fn` indicam que as métricas de treinamento e avaliação serão agregadas por média ponderada.

4.5.7.11. Inicialização do Servidor Flower

Por fim, o servidor é inicializado na porta `8080`.

```
fl.server.start_server(
    server_address="0.0.0.0:8080",
    config=fl.server.ServerConfig(num_rounds=3),
    strategy=strategy,
)
```

O endereço `0.0.0.0:8080` permite que o servidor aceite conexões dos containers clientes dentro da rede Docker. O parâmetro `num_rounds=3` define que o treinamento federado será executado por três rodadas. Em cada rodada, os clientes recebem o modelo global, treinam localmente, retornam seus parâmetros e o servidor realiza a agregação.

Síntese A implementação do servidor federado organiza todo o ciclo de treinamento colaborativo. Primeiro, o servidor carrega um conjunto global de teste, define a arquitetura do modelo e inicializa a estratégia FedAvg customizada. Em seguida, a cada rodada, recebe os modelos locais dos clientes, agrega os parâmetros, salva checkpoints, avalia o modelo global e registra as métricas em arquivos de histórico.

Essa implementação vai além de um servidor federado básico, pois adiciona mecanismos importantes para experimentação científica: avaliação global independente, armazenamento do melhor modelo, checkpoints por rodada e histórico estruturado em `JSON` e `CSV`. Esses elementos tornam o experimento mais reprodutível, auditável e adequado para análise posterior no contexto de aplicações médicas sensíveis.

4.6. Conclusão

Neste trabalho, foi apresentada uma abordagem integrada que combina fundamentos teóricos e implementação prática do aprendizado federado aplicado à análise de imagens citopatológicas para a triagem do câncer do colo do útero. Ao longo do texto, discutimos os principais conceitos relacionados ao aprendizado federado, redes neurais

 ©2026 SBC – Soc. Bras. de Computação

convolucionais e os desafios inerentes ao processamento de dados médicos distribuídos e sensíveis.

A construção do pipeline proposto evidenciou a importância de etapas como o pré-processamento de lâminas digitais, a seleção de regiões relevantes e a organização dos dados em um ambiente federado. A utilização de tecnologias como Docker, Flower e PyTorch permitiu a criação de um ambiente reprodutível, modular e escalável, capaz de simular cenários reais com múltiplas instituições colaborando de forma segura.

A implementação dos clientes e do servidor demonstrou, na prática, como o aprendizado federado pode ser operacionalizado, destacando o fluxo completo de treinamento distribuído, agregação de modelos e avaliação global. Esse processo reforça a viabilidade do uso do aprendizado federado em aplicações médicas, especialmente em contextos onde a privacidade dos dados é um requisito crítico.

Além disso, o trabalho evidencia que a integração entre aprendizado profundo e aprendizado federado permite mitigar desafios importantes, como o compartilhamento de dados sensíveis e a variabilidade entre diferentes fontes de dados. Como resultado, obtém-se um modelo global mais robusto e com maior capacidade de generalização.

Como perspectivas futuras, destacam-se a investigação de cenários com dados não independentes e não identicamente distribuídos (non-IID) mais complexos, a incorporação de técnicas de privacidade diferencial e segurança criptográfica, bem como a avaliação em ambientes reais com múltiplas instituições de saúde. Esses avanços podem contribuir significativamente para a consolidação do aprendizado federado como uma solução prática e confiável para sistemas de inteligência artificial na área da saúde.

Referências

[Bishop and Bishop 2024] Bishop, C. M. and Bishop, H. (2024). *Deep Learning: Foundations and Concepts*. Springer, Cham.

[Diniz et al. 2021] Diniz, D. N., Rezende, M. T., Bianchi, A. G. C., Carneiro, C. M., Ushizima, D. M., Medeiros, F. N. S. d., and Souza, M. J. F. (2021). A hierarchical feature-based methodology to perform cervical cancer classification. *Applied Sciences*, 11(9):4091.

[Goodfellow et al. 2016] Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA.

[INCA 2022] INCA (2022). Estimativa 2023: incidência de câncer no brasil. Technical report, Instituto Nacional de Câncer José Alencar Gomes da Silva, Rio de Janeiro.

[LeCun 1989] LeCun, Y. (1989). Generalization and network design strategies. Technical Report CRG-TR-89-4, University of Toronto.

[Liang et al. 2023] Liang, Y., Feng, S., Liu, Q., Kuang, H., Liu, J., Liao, L., Du, Y., and Wang, J. (2023). Exploring contextual relationships for cervical abnormal cell detection. *IEEE Journal of Biomedical and Health Informatics*, 27(8):4086–4097.

 ©2026 SBC – Soc. Bras. de Computação

[Liang et al. 2021] Liang, Y., Tang, Z., Yan, M., Chen, J., Liu, Q., and Xiang, Y. (2021). Comparison detector for cervical cell/clumps detection in the limited data scenario. *Neurocomputing*, 437:195–205.

[McMahan et al. 2017] McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. y. (2017). Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR.

[Scheffer 2025] Scheffer, M. C. (2025). Demografia médica no brasil 2025. Acesso em: 12 jan. 2026.

[Teixeira et al. 2023] Teixeira, J. B. A., Rezende, M. T., Diniz, D. N., Carneiro, C. M., Luz, E. J. d. S., Souza, M. J. F., Ushizima, D. M., Medeiros, F. N. S. d., and Bianchi, A. G. C. (2023). Segmentation of cervical nuclei using convolutional neural network for conventional cytology. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 11(5):1876–1888.

[Terra et al. 2023] Terra, D. C., Lisboa, A. C., Rezende, M. T., Carneiro, C. M., and Bianchi, A. G. C. (2023). Shape-based features investigation for preneoplastic lesions on cervical cancer diagnosis. In *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023) - Volume 4: VISAPP*, pages 506–513. SCITEPRESS.

 ©2026 SBC – Soc. Bras. de Computação

Capítulo 5

Agentes de Inteligência Artificial Aplicados à Saúde: Fundamentos, Arquiteturas e Implementação Prática

James D. Sousa, Frank Cesár Lopes Véras

Resumo

A Inteligência Artificial vêm transformando sistemas de saúde ao possibilitar automação, suporte à decisão clínica, monitoramento de pacientes e intervenções personalizadas. Nesse contexto, os agentes de IA emergem como uma evolução natural dos sistemas inteligentes, oferecendo maior autonomia, adaptabilidade e capacidade de interação. Este minicurso apresenta fundamentos teóricos e implementação prática de agentes de IA aplicados à saúde. Serão abordados modelos de linguagem de larga escala (LLMs), arquiteturas de agentes autônomos, geração aumentada por recuperação (RAG), sistemas de apoio à decisão clınica, integração de dados clınicos e aspectos éticos. Os participantes desenvolverão um protótipo funcional de agente de IA aplicado a cenários simulados da saúde. O objetivo é integrar teoria e prática no desenvolvimento de soluções digitais para saúde.

Abstract

Artificial Intelligence has been transforming healthcare systems by enabling automation, clinical decision support, patient monitoring, and personalized interventions. In this context, AI agents emerge as a natural evolution of intelligent systems, offering greater autonomy, adaptability, and interaction capabilities. This minicourse presents the theoretical foundations and practical implementation of AI agents applied to healthcare. Topics include large language models (LLMs), autonomous agent architectures, retrieval-augmented generation (RAG), clinical decision support systems, integration of clinical data, and ethical considerations. Participants will develop a functional prototype of an AI agent applied to simulated healthcare scenarios. The objective is to integrate theory and practice in the development of digital healthcare solutions.

 ©2026 SBC – Soc. Bras. de Computação

5.1. Introdução

A Inteligência Artificial (IA) pode ser compreendida como o campo da ciência da computação dedicado ao desenvolvimento de sistemas capazes de realizar tarefas que, até recentemente, exigiam exclusivamente a cognição humana como reconhecer padrões, tomar decisões, compreender linguagem e aprender com experiências passadas. Desde suas origens teóricas nas décadas de 1950 e 1960, a IA evoluiu de experimentos acadêmicos isolados para uma das forças mais transformadoras da economia global, presente em setores que vão da indústria ao entretenimento, das finanças à logística.

A incorporação da IA no setor da saúde representa uma das transformações mais significativas da era digital. Com o avanço das tecnologias computacionais e o aumento da disponibilidade de dados clínicos, tornou-se possível desenvolver sistemas capazes de auxiliar profissionais da saúde em tarefas complexas, desde diagnósticos até a gestão hospitalar (ESTEVA et al., 2019; TOPOL, 2019). Segundo a pesquisa TIC Saúde, 17% dos médicos brasileiros utilizam alguma forma de IA em sua prática clínica, número que chega a 20% nas instituições privadas (Núcleo de Informação e Coordenação do Ponto BR (NIC.br), 2024).

Levantamento da Associação Nacional de Hospitais Privados (Anahp) aponta que 62,5% dos hospitais privados já integram IA em suas operações (Associação Nacional de Hospitais Privados (Anahp), 2025). Globalmente, o mercado de healthtechs na América Latina registrou crescimento de 37,6% nos investimentos em 2024, atingindo US$ 253,7 milhões reflexo da confiança crescente da indústria no potencial transformador dessas tecnologias (Distrito; Associação Brasileira de Startups de Saúde e HealthTechs (ABSS), 2025).

5.1.1. Definição de Agente Inteligente

Um agente inteligente pode ser definido como uma entidade computacional capaz de perceber o ambiente em que está inserida, processar essas informações e agir de forma autônoma para atingir determinados objetivos. Essa definição envolve três componentes fundamentais: percepção, decisão e ação (RUSSELL; NORVIG; INTELLIGENCE, 1995; WOOLDRIDGE, 2009). No contexto da saúde, a percepção pode envolver a coleta de dados clínicos, como sinais vitais, exames laboratoriais e histórico do paciente. A etapa de decisão está relacionada à análise desses dados, frequentemente utilizando algoritmos de aprendizado de máquina ou modelos probabilísticos.

Por fim, a ação pode se manifestar na forma de recomendações clínicas, alertas ou até mesmo intervenções automatizadas (REZENDE, 2003; GOODFELLOW et al., 2016). Diferentemente de sistemas tradicionais, agentes inteligentes podem operar em ambientes dinâmicos e incertos, ajustando seu comportamento com base em novas informações. Essa característica é particularmente relevante na área da saúde, onde decisões precisam ser tomadas com base em dados incompletos e em constante atualização.

O Hospital Israelita Albert Einstein[1], por exemplo, desenvolveu o *HStory*, uma plataforma que utiliza um agente de IA generativa para consolidar dados de pacientes distribuídos em diferentes sistemas em um único painel centralizado, sintetizando automa-

[1]https://www.einstein.br/n

 ©2026 SBC – Soc. Bras. de Computação

ticamente o histórico clínico e devolvendo ao médico mais tempo para o cuidado direto. Trata-se de um agente que percebe dados dispersos no ambiente hospitalar exames, internações, prescrições e os transforma em informação acionável, sem depender de intervenção humana em cada etapa do processo.

5.1.2. Evolução Histórica da IA na Saúde

A aplicação da IA na saúde não é recente. Desde a década de 1970 sistemas especialistas já eram utilizados para auxiliar no diagnóstico médico. Um exemplo clássico é o MYCIN[2], desenvoolvido pela universidade de stanford, que utilizava regras baseadas no conhecimento de especialistas para identificar infecções bacterianas e recomendar tratamentos (SHORTLIFFE, 2012). O MYCIN demonstrou que era possível codificar o raciocínio clínico em regras lógicas, mas também revelou as limitações dessa abordagem sistemas rígidos que não aprendiam com novos dados e dependiam de atualização manual constante.

Com o avanço do poder computacional e o surgimento de grandes bases de dados, a IA evoluiu para incorporar técnicas de aprendizado de máquina, permitindo que sistemas aprendessem padrões diretamente dos dados. Essa mudança marcou a transição de sistemas baseados em regras para sistemas baseados em dados. Redes neurais artificiais, Support Vector Machines e métodos de ensemble passaram a dominar a pesquisa, alcançando resultados expressivos em tarefas como classificação de exames de imagem e predição de risco clínico.

Mais recentemente, o desenvolvimento de modelos de linguagem avançados trouxe uma nova dimensão para a IA na saúde. Esses modelos são capazes de compreender linguagem natural, interpretar textos médicos e interagir com usuários de forma mais intuitiva. Como resultado, surgiram agentes mais sofisticados, capazes de integrar múltiplas fontes de informação e oferecer suporte mais abrangente. Além disso, a crescente digitalização dos serviços de saúde, incluindo prontuários eletrônicos e dispositivos vestíveis, ampliou significativamente o volume de dados disponíveis. Isso possibilitou o desenvolvimento de agentes que não apenas reagem a eventos, mas também antecipam situações, contribuindo para uma abordagem mais preventiva no contexto clínico.

Portanto, a trajetória da IA na saúde pode ser dividida em grandes fases. Nos anos 1970, sistemas especialistas como MYCIN e INTERNIST-1 aplicavam lógica baseada em regras(SHORTLIFFE, 2012; MILLER; JR; MYERS, 1985). Nas décadas de 1980 e 1990, redes neurais artificiais começaram a ser aplicadas à análise de imagens médicas e sinais biomédicos(LIPPMANN, 1988). Nos anos 2000, o aprendizado de máquina e a mineração de dados em grandes bases clínicas dominaram a pesquisa(HAN; KAMBER; PEI, 2011). Nos anos 2010, o deep learning revolucionou o diagnóstico por imagem e o IBM Watson Health trouxe a IA para o debate clínico mainstream(LECUN; BENGIO; HINTON, 2015). Em 2017, a arquitetura Transformer mudou definitivamente o paradigma dos modelos de (VASWANI et al., 2017). Nos anos 2020, LLMs, agentes autônomos, RAG e o MCP consolidam uma nova geração de assistentes clínicos (LEWIS et al., 2020; ANTHROPIC, 2024).

[2]<https://www.historytools.org/inventions/mycin-expert-system-the-first-ai-medical-diagnosis>

 ©2026 SBC – Soc. Bras. de Computação

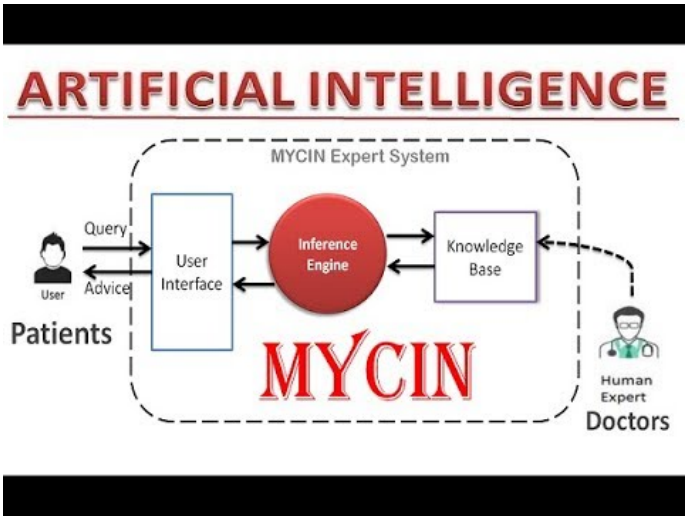


Figura 5.1. Aquitetura do MYCIN

5.1.3. Chatbot vs. Agente Autônomo

Embora os termos *chatbot* e agente autônomo sejam frequentemente utilizados de forma intercambiável no discurso popular, eles descrevem arquiteturas e capacidades fundamentalmente distintas. Compreender essa diferença é essencial para avaliar o potencial e as limitações de cada abordagem no contexto da saúde digital (Figura 5.9). Os chatbots tradicionais são sistemas de diálogo baseados em regras predefinidas ou em modelos estatísticos simples. Eles operam dentro de fluxos conversacionais estruturados, respondendo a intenções detectadas em mensagens dos usuários com respostas pré-cadastradas ou geradas a partir de *templates*. No contexto da saúde, chatbots desse tipo têm sido utilizados para triagem sintomática, agendamento de consultas e fornecimento de informações básicas de saúde pública.

Agentes autônomos, por sua vez, operam em um nível de sofisticação radicalmente superior. Eles são capazes de planejar sequências de ações, utilizar ferramentas externas, manter memória de interações anteriores e adaptar seu comportamento com base no contexto. Em vez de seguir fluxos predefinidos, um agente autônomo raciocina sobre o problema, decide quais recursos consultar e executa ações complexas com mínima intervenção humana. Na saúde, essa distinção tem implicações diretas enquanto um chatbot pode informar o horário de funcionamento de uma clínica, um agente autônomo pode analisar o histórico clínico de um paciente, consultar diretrizes atualizadas, identificar contraindicações e sugerir um plano de cuidado fundamentado em evidências.

Essa diferença se manifesta em diversas dimensões práticas. Em termos de arquitetura, chatbots seguem fluxos predefinidos com regras fixas, enquanto agentes operam por raciocínio dinâmico e planejamento adaptativo. Quanto à memória, chatbots geralmente não retêm informações entre sessões, ao passo que agentes mantêm memória de curto e longo prazo. No que diz respeito ao uso de ferramentas externas, chatbots raramente se integram a sistemas externos, enquanto agentes acessam APIs, bases de dados e prontuários de forma autônoma. Na prática clínica, essa distinção se traduz em capacidades radicalmente distintas um chatbot realiza triagem básica e responde a perguntas frequentes

 ©2026 SBC – Soc. Bras. de Computação

um agente como Ada Health, DAX Copilot ou HStory conduz raciocínio diagnóstico, personaliza recomendações e documenta automaticamente o atendimento.

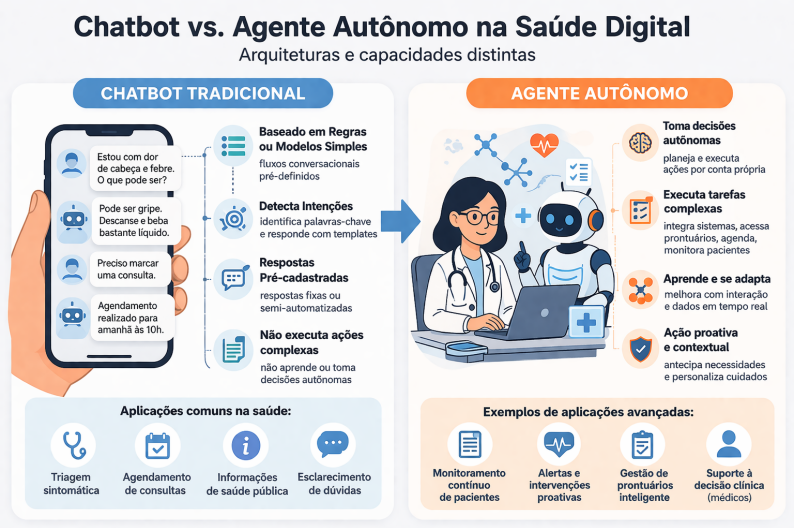


Figura 5.2. Chatbot x Agente

5.2. Como os Agentes de IA Funcionam

Compreender o funcionamento interno de um agente de IA é fundamental para avaliar seu potencial e suas limitações. Diferentemente de sistemas convencionais, que executam instruções predefinidas em sequências fixas, os agentes inteligentes modernos operam por meio de arquiteturas dinâmicas, capazes de perceber o ambiente, raciocinar sobre ele e agir de forma autônoma e adaptativa. Esta seção percorre os três pilares que sustentam esse funcionamento a arquitetura clássica de percepção – planejamento – ação, os modelos de linguagem de larga escala que viabilizam o raciocínio em linguagem natural, e as técnicas de recuperação de informação que permitem ao agente acessar e integrar conhecimento clínico especializado em tempo real.

5.2.1. Arquitetura Percepção – Planejamento – Ação

A compreensão do funcionamento interno de agentes inteligentes é fundamental para o desenvolvimento de sistemas autônomos robustos e confiáveis. Entre os diversos modelos propostos na literatura, a arquitetura baseada no ciclo percepção – planejamento – ação destaca-se como uma das mais clássicas e amplamente utilizadas (MÜLLER, 1999). Esse modelo descreve o comportamento de um agente como um processo contínuo de interação com o ambiente, no qual ele percebe informações, toma decisões e executa ações com o objetivo de atingir determinadas metas.

Esse paradigma tem origem em estudos da área de Inteligência Artificial e sistemas autônomos, sendo discutido em obras clássicas como as de Stuart Russell e Peter Norvig (RUSSELL; NORVIG; INTELLIGENCE, 1995), que estruturaram formalmente o conceito de agentes inteligentes como entidades que percebem o ambiente por meio de sensores e agem sobre ele por meio de atuadores.

 ©2026 SBC – Soc. Bras. de Computação

5.2.2. Percepção

A primeira etapa do ciclo corresponde à percepção, que envolve a coleta e interpretação de dados provenientes do ambiente. Esses dados podem assumir diversas formas, como texto, imagens, sinais biomédicos, registros clínicos ou interações do usuário. Em sistemas computacionais, essa etapa é geralmente implementada por meio de sensores lógicos (APIs, entradas de usuário, arquivos digitais) ou sensores físicos (dispositivos IoT, equipamentos médicos, etc.).

No contexto da saúde digital, por exemplo, a percepção pode envolver a leitura de prontuários eletrônicos, resultados de exames laboratoriais ou sinais vitais monitorados em tempo real. O agente precisa não apenas coletar esses dados, mas também interpretá-los corretamente, o que frequentemente exige técnicas de processamento de linguagem natural, visão computacional ou análise de séries temporais. Além disso, a percepção pode incluir etapas de pré-processamento, como limpeza de dados, normalização e extração de características relevantes. Essa fase é crítica, pois a qualidade das decisões do agente depende diretamente da qualidade das informações percebidas.

5.2.3. Planejamento

Após perceber o ambiente, o agente entra na fase de planejamento, na qual decide quais ações devem ser executadas para atingir seus objetivos. Essa etapa envolve raciocínio, tomada de decisão e, em muitos casos, previsão de cenários futuros. O planejamento pode variar em complexidade, desde regras simples (por exemplo, sistemas baseados em "se - então"), até algoritmos mais sofisticados, como busca heurística, algoritmos de otimização e aprendizado de máquina. Em arquiteturas mais avançadas, o agente pode manter um modelo interno do ambiente, permitindo simular diferentes estratégias antes de agir.

No contexto de agentes modernos baseados em IA, o planejamento também pode envolver o uso de modelos probabilísticos e técnicas de inferência, permitindo lidar com incertezas e informações incompletas. Em aplicações na saúde, por exemplo, o agente pode avaliar diferentes hipóteses diagnósticas com base nos dados disponíveis e selecionar a mais provável. Outro aspecto importante do planejamento é a definição de objetivos e restrições. Em ambientes críticos, como sistemas clínicos, as decisões precisam respeitar diretrizes médicas, protocolos de segurança e aspectos éticos, o que torna o planejamento ainda mais desafiador.

5.2.4. Ação

A terceira etapa do ciclo é a ação, na qual o agente executa as decisões tomadas durante o planejamento. As ações podem variar desde a geração de uma resposta textual até a execução de comandos em sistemas externos, como atualização de registros, envio de alertas ou controle de dispositivos. Em sistemas baseados em software, as ações geralmente são realizadas por meio de APIs, chamadas de sistema ou interfaces com outros serviços. Já em sistemas físicos, como robôs ou dispositivos médicos, as ações podem envolver atuadores que interagem diretamente com o ambiente.

Na área da saúde, exemplos de ações incluem a recomendação de tratamentos, o envio de alertas para profissionais de saúde ou a priorização de atendimentos com base na gravidade dos pacientes. É importante destacar que, nesses contextos, a execução

 ©2026 SBC – Soc. Bras. de Computação

de ações deve ser cuidadosamente controlada, muitas vezes exigindo validação humana (*human-in-the-loop*) para garantir segurança e confiabilidade.

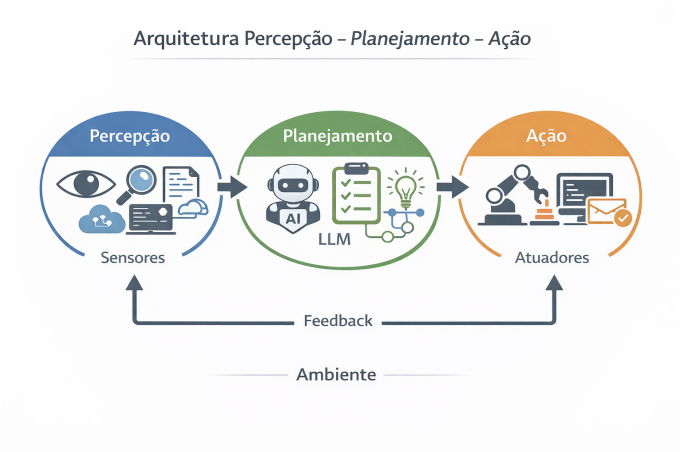


Figura 5.3. Arquitetura PPA

5.3. Memória dos Agentes

Uma dimensão frequentemente subestimada na arquitetura de agentes é a memória. Agentes de IA modernos operam com dois tipos principais de memória, cada um com papéis distintos no suporte à decisão clínica.

A memória de curto prazo corresponde ao contexto da sessão atual, incluindo o histórico de mensagens trocadas, os documentos já consultados e as decisões tomadas durante aquela interação específica. Trata-se de uma memória volátil, pois as informações normalmente são descartadas ao término da sessão. Em um agente clínico, esse tipo de memória permite que o sistema lembre, por exemplo, que o paciente mencionou alergia à penicilina no início da conversa ao formular uma recomendação terapêutica vários turnos depois. A capacidade da janela de contexto, geralmente medida em *tokens*, determina o quanto o agente consegue manter simultaneamente em processamento.

Já a memória de longo prazo refere-se a informações persistidas entre sessões, normalmente armazenadas em bancos de dados vetoriais ou relacionais. Esse mecanismo permite que o agente recupere interações anteriores com o mesmo paciente, acumule preferências institucionais e mantenha um histórico clínico continuamente consultável. Além disso, um terceiro tipo emergente vem ganhando relevância, a chamada memória episódica. Esse modelo permite ao agente recuperar interações semanticamente relevantes do passado, mesmo que não sejam as mais recentes. Em ambientes hospitalares, essa capacidade é particularmente importante para garantir continuidade do cuidado, sobretudo em pacientes crônicos que possuem múltiplas internações e tratamentos distribuídos ao longo do tempo.

5.4. Modelos de Linguagem de Larga Escala (LLMs)

Os Modelos de Linguagem de Larga Escala (*Large Language Models* – LLMs) representam a principal tecnologia responsável pela nova geração de agentes inteligentes. Esses

 ©2026 SBC – Soc. Bras. de Computação

sistemas utilizam técnicas de aprendizado profundo treinadas em enormes volumes de texto, permitindo compreender e gerar linguagem natural com elevado grau de sofisticação. Entre os exemplos mais conhecidos estão modelos desenvolvidos por organizações como a OpenAI[3] e o Google AI[4], que demonstraram avanços significativos na interação entre humanos e máquinas. Para compreender o papel desses modelos em agentes aplicados à saúde, é necessário entender sua arquitetura subjacente e o processo pelo qual adquirem conhecimento.

A arquitetura *Transformer*, proposta por Vaswani et al. (VASWANI et al., 2017), constitui a base dos modelos de linguagem modernos. Seu principal diferencial é o mecanismo de atenção, que permite ao modelo identificar quais partes de um texto são mais relevantes em determinado contexto. Esse mecanismo possibilita capturar relações semânticas entre palavras mesmo quando estão distantes umas das outras, superando limitações observadas em arquiteturas anteriores, como redes neurais recorrentes. Na prática, essa característica é particularmente útil em aplicações médicas. Ao analisar um prontuário clínico extenso, por exemplo, o modelo consegue relacionar informações mencionadas anteriormente como uma alergia à penicilina com decisões terapêuticas tomadas posteriormente durante a consulta.

O treinamento dos LLMs geralmente ocorre em duas etapas principais. Na primeira, denominada pré-treinamento, o modelo aprende padrões linguísticos a partir da leitura de grandes volumes de textos, incluindo livros, artigos científicos e conteúdos disponíveis na internet. Em seguida, ocorre o ajuste fino (*fine-tuning*) ou a aplicação de técnicas como o RLHF (*Reinforcement Learning from Human Feedback*), nas quais o modelo é refinado para responder de maneira mais útil, segura e alinhada às instruções humanas, inclusive em domínios específicos como a medicina.

O uso de LLMs na área médica avançou rapidamente nos últimos anos. Um exemplo relevante é o Med-PaLM 2[5], desenvolvido pelo Google, que alcançou desempenho comparável ao de médicos em avaliações relacionadas ao *United States Medical Licensing Examination* (USMLE). Outro exemplo importante é o BioMedLM[6], desenvolvido para aplicações biomédicas específicas, além do ClinicalBERT[7], treinado com prontuários clínicos e literatura médica. Esses modelos especializados apresentam desempenho significativo em tarefas como extração de informações clínicas, apoio ao diagnóstico e triagem de pacientes.

A principal vantagem desses modelos especializados é a capacidade de compreender terminologias médicas, abreviações clínicas e padrões comuns encontrados em prontuários e exames. Dessa forma, conseguem operar com maior precisão em ambientes clínicos quando comparados a modelos generalistas. Entretanto, ainda existem limitações importantes. Como dependem dos dados utilizados durante o treinamento, esses sistemas podem não possuir conhecimento atualizado sobre eventos recentes ou novas diretrizes médicas. Além disso, podem produzir respostas incorretas com alto grau aparente de confi-

[3]<https://openai.com>
[4]<https://ai.google>
[5]<https://sites.research.google/med-palm/>
[6]<https://crfm.stanford.edu/2022/12/15/biomedlm.html>
[7]<https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT>

 ©2026 SBC – Soc. Bras. de Computação

ança, fenômeno conhecido como "alucinação". Por esse motivo, recomenda-se que esses sistemas sejam utilizados em conjunto com fontes confiáveis de informação e mecanismos adicionais de verificação.

Nos agentes inteligentes, os LLMs funcionam como o núcleo de raciocínio do sistema. Eles recebem simultaneamente informações provenientes do ambiente, histórico de conversas, resultados de buscas externas e instruções operacionais, produzindo respostas em linguagem natural ou ações estruturadas, como chamadas de ferramentas externas. Essa capacidade de combinar raciocínio contextual com uso de ferramentas torna os agentes particularmente úteis em cenários complexos, como o ambiente hospitalar.

Um exemplo prático dessa aplicação é o Nabla Copilot[8]. Durante consultas médicas, o sistema processa a transcrição da conversa, consulta informações do prontuário do paciente e utiliza diretrizes médicas para gerar automaticamente um rascunho estruturado da consulta em tempo real. Ao final do processo, o profissional de saúde revisa e valida as informações antes do registro definitivo, mantendo a supervisão humana sobre a decisão clínica. Essa abordagem evidencia como a colaboração entre profissionais e sistemas de IA pode aumentar a eficiência operacional sem comprometer a responsabilidade médica e a qualidade do atendimento.

5.5. Retrieval-Augmented Generation (RAG)

Uma das principais limitações dos LLMs, especialmente em áreas críticas como a saúde, é que eles não têm acesso a informações atualizadas automaticamente. Eles aprendem com dados até um certo momento e, depois disso, não sabem de novas diretrizes médicas, estudos recentes, novos medicamentos ou até informações específicas de um paciente que estão em sistemas hospitalares. Para resolver esse problema, surgiu uma abordagem chamada *Retrieval-Augmented Generation (RAG)*. De forma simples, o RAG combina duas coisas: a capacidade do modelo de gerar respostas com a habilidade de buscar informações atualizadas em fontes externas. Em vez de depender só do que aprendeu no passado, o sistema procura dados relevantes em tempo real como documentos, bases médicas ou registros e usa essas informações para construir respostas mais corretas e atualizadas.

5.5.1. Como o RAG funciona: o pipeline de recuperação

O funcionamento do Retrieval-Augmented Generation (RAG) pode ser entendido de forma simples como um processo em etapas. Primeiro, os documentos importantes como diretrizes médicas, artigos científicos e protocolos são organizados e transformados em uma espécie de "linguagem numérica" que representa o significado dos textos. Isso permite que o sistema entenda melhor o conteúdo, e tudo fica armazenado em um banco de dados preparado para esse tipo de busca. Depois, quando alguém faz uma pergunta, o sistema transforma essa pergunta nesse mesmo formato e procura, dentro da base, os trechos mais parecidos em termos de significado. Ou seja, ele não busca só por palavras iguais, mas por conteúdo relevante. Em seguida, esses trechos encontrados são enviados junto com a pergunta para o modelo. Assim, o LLM não precisa depender apenas do que aprendeu no passado ele passa a usar informações atualizadas naquele momento. Por fim, o modelo gera uma resposta com base nesses documentos. Em muitos casos, ele pode até indicar

[8]<https://www.nabla.com/copilot>

 ©2026 SBC – Soc. Bras. de Computação

de onde tirou as informações, o que é muito importante em áreas como a saúde, onde é essencial saber a origem dos dados e garantir mais segurança nas decisões.

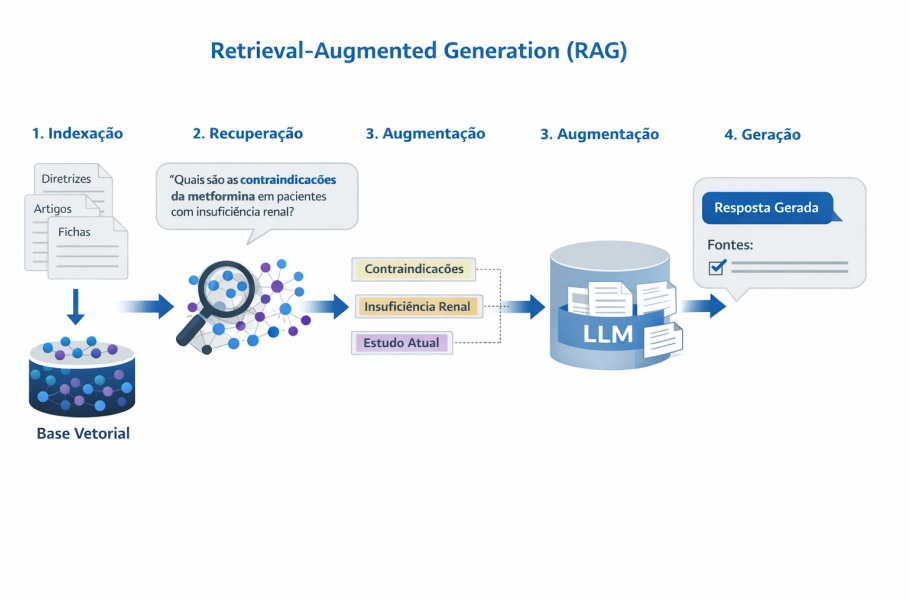


Figura 5.4. Aquitetura Rag

5.5.2. Tipos de Recuperação e Recuperação Híbrida

Existem dois paradigmas principais de recuperação de informação em sistemas baseados em RAG. O primeiro é a busca densa (*dense retrieval*), fundamentada em embeddings vetoriais e similaridade semântica. Nesse modelo, documentos e consultas são representados como vetores em um espaço multidimensional, permitindo recuperar conteúdos semanticamente relacionados mesmo quando não há correspondência literal entre os termos utilizados. O segundo paradigma é a busca esparsa (*sparse retrieval*), baseada em métodos tradicionais como TF-IDF e BM25, nos quais a recuperação ocorre principalmente por correspondência lexical entre palavras presentes na consulta e nos documentos.

Cada abordagem apresenta vantagens e limitações específicas. A busca densa tende a capturar melhor a intenção semântica da consulta, sendo particularmente útil quando diferentes termos podem representar conceitos semelhantes. Já a busca esparsa oferece elevada precisão em cenários nos quais termos técnicos exatos são fundamentais, como nomes de medicamentos, códigos CID, resultados laboratoriais ou nomenclaturas de procedimentos médicos.

Para superar as limitações individuais de cada abordagem, muitos sistemas modernos utilizam recuperação híbrida (*hybrid retrieval*), combinando simultaneamente busca densa e esparsa. Após a etapa inicial de recuperação, técnicas de *re-ranking* podem ser aplicadas para reorganizar os documentos encontrados conforme critérios adicionais de relevância semântica e contextual antes que o conteúdo seja encaminhado ao modelo de linguagem. Essa estratégia melhora significativamente a qualidade das respostas geradas, especialmente em ambientes clínicos, nos quais precisão terminológica e compreensão contextual são igualmente importantes.

 ©2026 SBC – Soc. Bras. de Computação

5.5.3. Vantagens do RAG no contexto clínico

O uso do Retrieval-Augmented Generation na área da saúde traz varios benefícios que vão além da simples atualização do conhecimento. Em primeiro lugar, essa técnica reduz significativamente o risco de alucinações: quando o modelo é explicitamente instruído a responder com base apenas nos documentos recuperados, a tendência de gerar informações não fundamentadas diminui de forma substancial.

Em segundo lugar, o RAG permite a personalização do agente para contextos institucionais específicos: um hospital pode indexar seus próprios protocolos, formulários e diretrizes locais, garantindo que o agente opere com base nas práticas da instituição e não em recomendações genéricas.

Em terceiro lugar, o RAG viabiliza a integração de dados do paciente em tempo real. Ao indexar os registros clínicos de um paciente específico incluindo histórico de exames, diagnósticos anteriores, alergias e medicamentos em uso, o agente pode recuperar essas informações contextualmente relevantes e incorporá-las ao raciocínio clínico. Esse modelo é o que sustenta, por exemplo, plataformas como o Suki AI, assistente de voz para médicos amplamente adotado nos Estados Unidos, que recupera informações do prontuário do paciente para contextualizar automaticamente as notas clínicas geradas durante a consulta.

5.6. Uso de Ferramentas Externas e o Model Context Protocol (MCP)

Agentes de IA modernos não operam de forma isolada. Para executar tarefas complexas, precisam interagir continuamente com sistemas externos, realizando operações como consultar bases de dados clínicas, acessar APIs laboratoriais, verificar disponibilidade de leitos, agendar exames e recuperar informações de prontuários eletrônicos. Essa capacidade de utilizar ferramentas externas frequentemente denominada *tool use* ou *function calling* constitui uma das principais características que diferenciam agentes autônomos de simples modelos geradores de texto.

Historicamente, integrações entre sistemas eram desenvolvidas de maneira *ad hoc*, utilizando código customizado, altamente acoplado e de difícil manutenção. Em ambientes hospitalares, essa limitação torna-se ainda mais evidente devido à coexistência de múltiplos sistemas heterogêneos, como HIS (*Hospital Information Systems*), LIS (*Laboratory Information Systems*), RIS (*Radiology Information Systems*) e plataformas de farmácia hospitalar. Além disso, esses sistemas frequentemente utilizam padrões distintos de comunicação, incluindo HL7 v2[9], FHIR[10] e protocolos proprietários.

Essa fragmentação tecnológica representa um obstáculo significativo para a adoção de agentes inteligentes em larga escala na saúde, pois dificulta interoperabilidade, padronização e manutenção das integrações. Consequentemente, grande parte do esforço de desenvolvimento desses agentes acaba sendo direcionada não apenas à inteligência do modelo, mas também à criação de mecanismos robustos de comunicação entre diferentes sistemas clínicos e administrativos.

[9] <https://www.hl7.org/implement/standards/product_brief.cfm?product_id=185>
[10] <https://www.hl7.org/fhir/>

 ©2026 SBC – Soc. Bras. de Computação

5.6.1. O Que é o MCP

O *Model Context Protocol* (MCP) é um protocolo aberto lançado pela Anthropic em 2024 que padroniza a forma como LLMs se comunicam com ferramentas e fontes de dados externas. Frequentemente descrito como um "USB para agentes de IA", o MCP define uma interface comum: qualquer ferramenta ou sistema que implemente o protocolo pode ser conectado a qualquer agente compatível, sem necessidade de integrações customizadas. Desde seu lançamento, o protocolo foi adotado por dezenas de plataformas, incluindo IDEs de desenvolvimento, ferramentas de produtividade e, progressivamente, sistemas de informação em saúde.

A arquitetura do MCP é composta por três elementos principais. O *MCP Host* corresponde à aplicação do agente, responsável por orquestrar as interações. O *MCP Client* gerencia as conexões entre o host e os servidores. Já o *MCP Server* expõe ferramentas e fontes de dados específicas, como servidores responsáveis pelo acesso a prontuários eletrônicos, consulta a bases farmacológicas ou verificação da disponibilidade de leitos hospitalares.

5.6.2. Relevância Clínica do MCP

No contexto hospitalar, o MCP possui implicações práticas imediatas. Um agente de apoio à decisão clínica pode, por meio de servidores MCP, consultar resultados de exames em tempo real, verificar interações medicamentosas em bases farmacológicas como o DrugBank[11], checar disponibilidade de especialistas para teleconsultas e registrar evoluções diretamente no prontuário eletrônico. Tudo isso ocorre utilizando um protocolo padronizado de comunicação, reduzindo significativamente a necessidade de integrações proprietárias específicas para cada sistema hospitalar.

Além dos ganhos operacionais, o MCP oferece vantagens relevantes sob a perspectiva regulatória. Ao estabelecer uma camada explícita e padronizada de comunicação entre agentes inteligentes e sistemas externos, o protocolo facilita auditoria, rastreabilidade e monitoramento das ações executadas pelo agente. Em um ambiente altamente regulado como o setor de saúde, no qual rastreabilidade representa simultaneamente uma exigência ética e legal especialmente diante da Lei Geral de Proteção de Dados (LGPD)[12] e das regulamentações da ANVISA[13] essa característica torna-se particularmente valiosa.

Apesar de seu potencial, o MCP ainda é considerado um protocolo relativamente recente. O ecossistema de servidores especializados para sistemas de informação em saúde permanece em rápida evolução, com iniciativas voltadas à integração de prontuários eletrônicos, plataformas laboratoriais, sistemas de imagem médica e serviços de suporte clínico baseados em IA. Além disso, desafios importantes ainda precisam ser superados para que o protocolo alcance ampla adoção em ambientes hospitalares reais. Entre esses desafios destacam-se questões relacionadas à padronização semântica dos dados clínicos, interoperabilidade entre sistemas legados, controle de permissões de acesso e garantia de privacidade das informações sensíveis dos pacientes. Em muitos hospitais, parte significativa da infraestrutura ainda depende de softwares antigos e altamente personalizados, o que

[11] <https://go.drugbank.com>

[12] <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>

[13] <https://www.gov.br/anvisa/pt-br>

 ©2026 SBC – Soc. Bras. de Computação

dificulta a adoção de arquiteturas modernas baseadas em agentes inteligentes e protocolos padronizados.

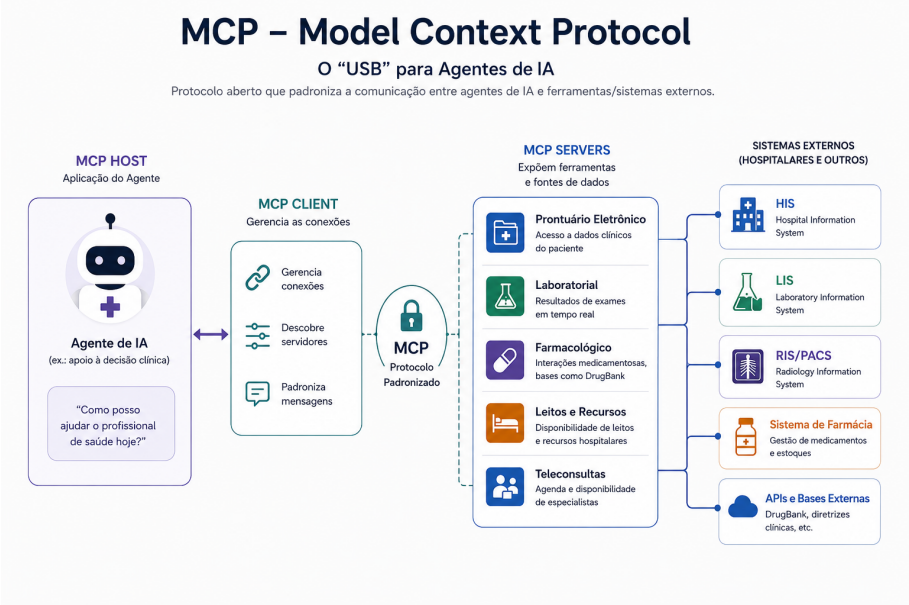


Figura 5.5. MCP

5.7. Sistemas Multi-Agentes

Em cenários clínicos de elevada complexidade, um único agente inteligente pode não ser suficiente para lidar adequadamente com todas as demandas operacionais e cognitivas do ambiente hospitalar. Nesse contexto, surgem os sistemas multiagentes (*Multi-Agent Systems* – MAS), nos quais múltiplos agentes especializados distribuem responsabilidades e colaboram para resolver problemas de maneira coordenada. Cada agente possui competências específicas e delimitadas, comunicando-se com os demais por meio de protocolos definidos e mecanismos estruturados de troca de informações.

Um exemplo aplicado à saúde pode ser observado em ambientes de pronto-socorro. Um agente de triagem seria responsável por classificar pacientes conforme critérios de gravidade e encaminhar os casos para agentes especializados. Pacientes com dor torácica poderiam ser direcionados a um agente cardiológico, enquanto quadros com déficit neurológico focal seriam encaminhados a um agente neurológico. Paralelamente, um agente de gestão hospitalar poderia monitorar disponibilidade de leitos e coordenar processos de internação quando necessário. Embora cada agente opere com relativa autonomia dentro de sua especialidade, o sistema global funciona de maneira integrada e colaborativa.

A coordenação desses ambientes multiagentes é viabilizada por *frameworks* especializados, como LangGraph[14], AutoGen[15] e CrewAI[16]. Essas plataformas oferecem infraestrutura para orquestração de agentes, definição de grafos de dependência, gerenciamento de fluxo de informações e controle das interações entre diferentes componentes do

[14]<https://www.langchain.com/langgraph>
[15]<https://microsoft.github.io/autogen/>
[16]<https://www.crewai.com>

 ©2026 SBC – Soc. Bras. de Computação

sistema. Em aplicações médicas, essa abordagem permite distribuir tarefas complexas entre múltiplos agentes especializados, aumentando escalabilidade, modularidade e capacidade de tomada de decisão em ambientes clínicos dinâmicos.

5.8. Integração com Bases de Conhecimento Clínico

A qualidade de um agente de IA na área da saúde depende muito das informações que ele consegue acessar. Quanto melhores e mais completas forem essas fontes, mais útil ele será na prática. Essas informações ficam organizadas em bases de conhecimento clínico. Elas funcionam como grandes repositórios que guardam o saber médico acumulado ao longo do tempo. Ali estão dados sobre diagnósticos, tratamentos, medicamentos e protocolos de atendimento. Tudo isso é estruturado de forma que os sistemas consigam entender e usar facilmente. Ou seja, não é só informação armazenada é informação pronta para ser consultada e aplicada. Quando essas bases são integradas ao funcionamento do agente, ele deixa de ser um sistema genérico e passa a atuar de forma mais especializada. Na prática, isso significa que o agente consegue interpretar melhor cada situação, tomar decisões mais informadas e oferecer respostas mais úteis no contexto clínico.

5.8.1. Ontologias e terminologias médicas padronizadas

A interoperabilidade em sistemas de saúde depende fortemente do uso de terminologias e ontologias médicas padronizadas, que funcionam como uma linguagem comum entre diferentes plataformas clínicas e administrativas. Entre os principais padrões utilizados destaca-se o SNOMED CT[17], uma ontologia clínica abrangente que organiza conceitos médicos e suas relações semânticas. O CID-11[18] é amplamente utilizado para padronização de diagnósticos, enquanto o LOINC[19] fornece nomenclaturas e códigos para exames laboratoriais, medições clínicas e observações médicas.

Quando agentes de IA incorporam essas terminologias em seus mecanismos de processamento, passam a compreender de maneira mais robusta a linguagem médica utilizada em prontuários, laudos e registros clínicos. Isso inclui o reconhecimento de sinônimos, abreviações e diferentes formas de representação de uma mesma condição clínica. Na prática, o sistema consegue identificar que expressões como “insuficiência cardíaca congestiva”, “ICC” ou outras variações semânticas correspondem ao mesmo conceito médico subjacente.

Essa capacidade de normalização semântica torna os agentes mais robustos, interoperáveis e confiáveis, especialmente em ambientes clínicos reais, nos quais a documentação médica frequentemente apresenta elevada variabilidade terminológica e diferenças na forma de registro das informações. Além disso, o uso dessas ontologias facilita integração entre sistemas heterogêneos, melhora a recuperação de informações clínicas e contribui para maior consistência em processos de apoio à decisão médica baseados em inteligência artificial.

[17] <https://www.snomed.org/snomed-ct>
[18] <https://icd.who.int/en>
[19] <https://loinc.org>

 ©2026 SBC – Soc. Bras. de Computação

5.8.2. Bases de literatura científica e diretrizes clínicas

Além do uso de ontologias médicas, agentes de IA aplicados à saúde tornam-se significativamente mais úteis quando conseguem acessar informações atualizadas provenientes da literatura científica biomédica. Um dos principais exemplos é o PubMed[20], uma das maiores bases de dados de literatura biomédica do mundo, mantida pela *National Library of Medicine*. Também existem plataformas especializadas em suporte à decisão clínica baseada em evidências, como o UpToDate[21] e o DynaMed[22], que organizam informações científicas na forma de recomendações clínicas estruturadas.

Quando agentes inteligentes se conectam a essas fontes utilizando técnicas como *Retrieval-Augmented Generation* (RAG), deixam de atuar apenas como sistemas geradores de respostas genéricas e passam a recuperar informações científicas confiáveis para fundamentar suas recomendações. Dessa forma, o agente pode não apenas responder uma pergunta clínica, mas também indicar artigos científicos, diretrizes médicas e níveis de evidência que sustentam determinada conduta terapêutica ou diagnóstica.

Um exemplo prático dessa abordagem é o Consensus[23], um mecanismo de busca baseado em IA voltado para literatura científica revisada por pares. O sistema permite que usuários realizem perguntas em linguagem natural e recebam respostas fundamentadas em artigos científicos, acompanhadas de resumos automáticos sobre o consenso existente na literatura acadêmica.

Na prática, esse tipo de integração auxilia médicos, pesquisadores e profissionais de saúde a economizar tempo durante processos de revisão bibliográfica e tomada de decisão clínica. Além disso, agentes capazes de recuperar evidências científicas atualizadas tendem a produzir recomendações mais transparentes, auditáveis e alinhadas às práticas da medicina baseada em evidências. No futuro, espera-se que sistemas desse tipo desempenhem papel cada vez mais relevante no suporte à decisão clínica, oferecendo assistência contextualizada e continuamente atualizada conforme a evolução da literatura biomédica.

5.8.3. Prontuários Eletrônicos e interoperabilidade via HL7 FHIR

A integração com Sistemas de Registro Eletrônico de Saúde (*Electronic Health Records* – EHR) ou Prontuários Eletrônicos do Paciente (PEP) representa um dos elementos centrais para viabilizar o uso prático de agentes de IA no ambiente clínico. Esses sistemas concentram informações essenciais sobre os pacientes, incluindo histórico médico, exames laboratoriais, diagnósticos, prescrições e evoluções clínicas. Para que diferentes plataformas consigam interoperar de maneira padronizada, utiliza-se amplamente o padrão HL7 FHIR (*Fast Healthcare Interoperability Resources*)[24], desenvolvido pela *Health Level Seven International*.

O padrão FHIR define uma estrutura organizada para representação e troca de dados clínicos, permitindo modelar recursos relacionados a pacientes, exames, diagnósticos,

[20] <https://pubmed.ncbi.nlm.nih.gov>
[21] <https://www.uptodate.com>
[22] <https://www.dynamed.com>
[23] <https://consensus.app>
[24] <https://www.hl7.org/fhir/>

 ©2026 SBC – Soc. Bras. de Computação

prescrições, procedimentos e observações médicas. Na prática, isso possibilita que agentes inteligentes acessem informações clínicas por meio de APIs padronizadas, funcionando como uma interface segura e interoperável entre sistemas hospitalares e aplicações baseadas em inteligência artificial. Além da leitura de dados, esses agentes também podem atualizar registros clínicos, registrar evoluções médicas e auxiliar fluxos operacionais de forma integrada ao ecossistema hospitalar.

Diversas soluções comerciais já utilizam esse modelo de integração. Ferramentas como o Nabla Copilot[25] e o Abridge[26] conectam-se diretamente a sistemas hospitalares para auxiliar processos de documentação clínica. O Abridge, por exemplo, já é utilizado em instituições como o UPMC (*University of Pittsburgh Medical Center*)[27], nos Estados Unidos. Esses agentes conseguem recuperar automaticamente informações do paciente, processar transcrições de consultas e gerar anotações clínicas estruturadas em tempo real, salvando os dados diretamente no prontuário eletrônico.

Esse tipo de automação reduz significativamente o volume de tarefas administrativas executadas manualmente pelos profissionais de saúde, diminuindo erros de documentação e aumentando eficiência operacional. Como consequência, médicos e equipes clínicas podem dedicar mais tempo ao cuidado direto do paciente, reduzindo sobrecarga burocrática e melhorando a qualidade da assistência prestada.

5.9. Exemplos Reais: Aplicações de Referência Mundial

Esta seção apresenta os principais exemplos reais de agentes e sistemas de inteligência artificial aplicados à saúde em escala global. Cada caso demonstra como os fundamentos teóricos discutidos anteriormente se materializam em produtos e plataformas que já impactam pacientes, profissionais de saúde e instituições hospitalares em diferentes partes do mundo.

5.9.1. Ada Health: Triagem Clínica Baseada em IA

A Ada Health[28] foi fundada em 2011 em Berlim, Alemanha, e tornou-se uma das plataformas de avaliação de sintomas por inteligência artificial mais reconhecidas mundialmente. Seu principal produto consiste em um aplicativo de saúde disponível em mais de 130 países, capaz de conduzir entrevistas clínicas adaptativas com usuários e gerar listas priorizadas de possíveis condições médicas com base nos sintomas relatados.

O mecanismo central da plataforma combina raciocínio probabilístico bayesiano com modelos de processamento de linguagem natural. Diferentemente de árvores de decisão estáticas, o sistema adapta dinamicamente as próximas perguntas conforme as respostas fornecidas pelo usuário, aproximando-se do raciocínio clínico utilizado por médicos durante consultas reais. A base de conhecimento da plataforma cobre mais de 3.000 condições médicas, incluindo doenças raras, sendo continuamente revisada e atualizada por equipes médicas especializadas.

[25] <https://www.nabla.com/copilot>
[26] <https://www.abridge.com>
[27] <https://www.upmc.com>
[28] <https://ada.com>

 ©2026 SBC – Soc. Bras. de Computação

Estudos publicados pela empresa e por pesquisadores independentes indicam que a Ada inclui o diagnóstico correto entre as três primeiras sugestões em mais de 90% dos casos avaliados, desempenho considerado comparável ao de médicos generalistas em cenários de triagem clínica. A plataforma também foi integrada a iniciativas de saúde pública em países africanos, como Ruanda e Tanzânia, com o objetivo de ampliar acesso à avaliação clínica em regiões com escassez de profissionais de saúde.

No Brasil, a empresa possui parcerias com operadoras de saúde e trabalha na adaptação cultural e linguística da plataforma ao contexto epidemiológico nacional, incluindo condições endêmicas como dengue, leishmaniose e doença de Chagas. Seu modelo de negócios baseado em integração B2B permite que hospitais e planos de saúde incorporem a tecnologia em fluxos digitais de triagem, reduzindo atendimentos desnecessários em pronto-socorros e direcionando pacientes para níveis adequados de cuidado.

5.9.2. Google DeepMind: AlphaFold e Gemini na Medicina

O Google DeepMind[29] produziu alguns dos avanços mais relevantes da inteligência artificial aplicada à saúde na última década, destacando-se especialmente o AlphaFold e os modelos multimodais Gemini voltados para aplicações biomédicas.

5.9.2.1. AlphaFold: Revolução na Biologia Estrutural

O AlphaFold[30] foi apresentado originalmente em 2020 e solucionou um dos maiores desafios da biologia molecular: a predição da estrutura tridimensional de proteínas a partir de sequências de aminoácidos. Antes dessa tecnologia, determinar experimentalmente a estrutura de proteínas podia demandar anos de trabalho e custos extremamente elevados.

O modelo utiliza arquiteturas derivadas de Transformers aplicadas ao processamento de sequências proteicas, permitindo prever estruturas com precisão próxima à experimental em poucos minutos. As implicações médicas são profundas, especialmente no contexto de doenças relacionadas a proteínas mal dobradas, como Alzheimer, Parkinson e diversos tipos de câncer.

Em parceria com o EMBL-EBI[31], o DeepMind disponibilizou publicamente um banco contendo mais de 200 milhões de estruturas proteicas previstas por IA. Empresas farmacêuticas como AstraZeneca, Bayer e Pfizer passaram a utilizar a tecnologia em processos de descoberta de medicamentos, enquanto a Isomorphic Labs[32] utiliza variantes do modelo para desenvolvimento de terapias baseadas em IA.

Em 2024, foi lançado o AlphaFold 3, expandindo as capacidades do sistema para modelagem de interações entre proteínas, DNA, RNA e pequenas moléculas terapêuticas.

[29] <https://deepmind.google>
[30] <https://alphafold.ebi.ac.uk>
[31] <https://www.ebi.ac.uk>
[32] <https://www.isomorphiclabs.com>

 ©2026 SBC – Soc. Bras. de Computação

5.9.2.2. Med-PaLM 2 e Gemini: LLMs para Aplicações Médicas

O Med-PaLM 2[33] foi desenvolvido pela Google Research especificamente para aplicações médicas. Em avaliações padronizadas como o USMLE (*United States Medical Licensing Examination*), o modelo alcançou desempenho comparável ao de médicos especialistas, representando um marco importante na aplicação de LLMs em domínios clínicos altamente especializados.

O modelo demonstrou capacidade de responder perguntas abertas sobre casos clínicos, interpretar exames laboratoriais e gerar explicações avaliadas por médicos como úteis ou muito úteis. Posteriormente, o Gemini[34] expandiu essas capacidades ao introduzir processamento multimodal integrado de texto, imagens radiológicas, lâminas histopatológicas e imagens dermatológicas.

Estudos preliminares demonstraram desempenho competitivo na interpretação de radiografias de tórax e classificação de lesões dermatológicas. Além disso, a integração com mecanismos de busca e recuperação de informações científicas via RAG permite acessar literatura médica atualizada em tempo real, reduzindo limitações relacionadas ao corte temporal de treinamento dos modelos.

5.9.3. Infermedica: Plataforma de Triagem Probabilística

A Infermedica[35], fundada na Polônia em 2012, desenvolveu uma das plataformas de triagem clínica baseada em IA mais amplamente validadas internacionalmente. Diferentemente de sistemas baseados em árvores de decisão fixas, a plataforma utiliza modelos probabilísticos bayesianos capazes de avaliar centenas de condições clínicas, milhares de sintomas e múltiplos fatores de risco em dezenas de idiomas.

O paciente descreve seus sintomas iniciais e o sistema conduz entrevistas dinâmicas adaptativas em tempo real. Ao final, os casos são classificados em diferentes níveis de urgência clínica, incluindo emergência, urgência, semiurgência, não urgência e autocuidado. Estudos reportados pela empresa indicam elevada segurança clínica nas recomendações emitidas e redução significativa no tempo médio de triagem.

5.9.4. DAX Copilot (Nuance/Microsoft): Documentação Clínica Automatizada

O DAX Copilot[36] representa uma das aplicações mais avançadas de IA para documentação clínica automatizada. Desenvolvido originalmente pela Nuance Communications, posteriormente adquirida pela Microsoft em 2022, o sistema utiliza tecnologia de *ambient listening* para capturar automaticamente conversas entre médicos e pacientes durante consultas clínicas.

Após o término da consulta, o sistema gera automaticamente notas estruturadas no formato SOAP (*Subjective, Objective, Assessment, Plan*), integrando-as diretamente ao prontuário eletrônico. O sistema possui integração com plataformas como Epic EHR[37],

[33] <https://sites.research.google/med-palm/>
[34] <https://deepmind.google/technologies/gemini/>
[35] <https://infermedica.com>
[36] <https://www.microsoft.com/en-us/health-solutions/dragon-copilot>
[37] <https://www.epic.com>

 ©2026 SBC – Soc. Bras. de Computação

amplamente utilizadas em hospitais norte-americanos.

Resultados publicados indicam redução significativa no tempo gasto com documentação médica, economia de vários minutos por consulta e diminuição importante nos índices de burnout associados à sobrecarga administrativa. Em versões mais recentes, o sistema passou a incluir funcionalidades adicionais, como sugestão automática de códigos de faturamento e identificação de lacunas em cuidados preventivos.

5.9.5. Tempus AI: Medicina de Precisão em Oncologia

A Tempus AI[38], fundada em Chicago em 2015, opera uma das maiores plataformas de medicina de precisão voltadas à oncologia. A empresa integra milhões de registros clínicos com dados de sequenciamento genômico avançado, incluindo exoma completo e RNA-seq, criando bases de dados utilizadas para desenvolvimento de modelos preditivos e suporte terapêutico personalizado.

A plataforma identifica mutações tumorais relevantes, sugere terapias-alvo aprovadas pela FDA associadas às alterações genéticas encontradas e identifica ensaios clínicos compatíveis com o perfil molecular do paciente. Esse modelo auxilia oncologistas na identificação de opções terapêuticas especialmente em cânceres raros ou casos complexos.

Além disso, a empresa desenvolve modelos capazes de prever resposta terapêutica, risco de progressão tumoral e probabilidade de recorrência da doença, permitindo estratégias mais individualizadas de acompanhamento clínico.

5.9.6. Abridge: Transcrição e Sumarização de Consultas

A Abridge[39] é uma startup norte-americana fundada em 2018 voltada à transcrição e sumarização automática de consultas médicas utilizando inteligência artificial. O sistema captura conversas entre médicos e pacientes e gera automaticamente resumos estruturados contendo diagnósticos discutidos, medicamentos prescritos, exames solicitados e planos terapêuticos.

A plataforma é utilizada pelo UPMC (*University of Pittsburgh Medical Center*), um dos maiores sistemas hospitalares dos Estados Unidos. Estudos preliminares indicam redução significativa da carga de documentação pós-consulta e aumento da satisfação dos profissionais com a qualidade das notas clínicas produzidas automaticamente.

O sistema foi desenvolvido especificamente para lidar com a elevada variabilidade linguística observada em consultas reais, incluindo terminologia médica, abreviações, interrupções e mudanças frequentes de contexto.

5.9.7. Hospital Israelita Albert Einstein: HStory

No contexto brasileiro, o Hospital Israelita Albert Einstein[40] desenvolveu o HStory, uma plataforma baseada em IA generativa voltada à consolidação inteligente de dados clínicos distribuídos em múltiplos sistemas hospitalares.

[38]<https://www.tempus.com>
[39]<https://www.abridge.com>
[40]<https://www.einstein.br>

 ©2026 SBC – Soc. Bras. de Computação

A solução foi criada para enfrentar um problema recorrente em hospitais de grande porte: a fragmentação das informações clínicas ao longo de diferentes atendimentos, especialidades e sistemas de informação. O HStory utiliza agentes inteligentes para reunir, estruturar e sintetizar automaticamente essas informações em um painel clínico centralizado acessível durante o atendimento médico.

Dessa forma, em vez de navegar manualmente por múltiplos registros fragmentados, o profissional recebe uma narrativa clínica consolidada contendo histórico de internações, diagnósticos prévios, alergias, medicamentos em uso, exames relevantes e procedimentos anteriores. A iniciativa é considerada uma das aplicações mais avançadas de IA generativa em ambiente hospitalar na América Latina.

5.10. Benefícios dos Agentes de IA na Saúde

A incorporação de agentes inteligentes em ambientes clínicos e hospitalares representa uma das transformações mais profundas vivenciadas pelo setor de saúde nas últimas décadas. Impulsionados pelos avanços em aprendizado de máquina, processamento de linguagem natural, processamento de imagens e sistemas especialistas, esses agentes têm demonstrado capacidade de atuar em múltiplas frentes operacionais, desde a burocracia administrativa até o suporte direto ao raciocínio diagnóstico. Esta seção apresenta os principais benefícios documentados na literatura científica, organizados em cinco eixos temáticos como automação de processos clínicos, apoio à decisão médica, redução de sobrecarga administrativa, acesso à literatura científica e personalização no atendimento ao paciente.

5.10.1. Automação de Processos Clínicos

A automação de processos clínicos constitui um dos eixos de maior impacto imediato da inteligência artificial na saúde. Processos que historicamente demandavam horas de trabalho humano como o agendamento de consultas, a triagem de pacientes, o monitoramento de sinais vitais e a reconciliação de medicamentos passam a ser executados por agentes automáticos com maior velocidade, menor taxa de erro e disponibilidade contínua ao longo de todas as horas do dia (TOPOL, 2019).

Em ambientes de pronto-socorro, por exemplo, agentes inteligentes são capazes de realizar a triagem inicial dos pacientes com base em sintomas relatados, histórico clínico e dados de sinais vitais coletados. Estudos demonstram que sistemas de triagem baseados em IA apresentam acurácia comparável ou superior à triagem humana convencional em categorias de risco intermediário, além de reduzir significativamente o tempo de espera (ALOMARI et al., 2025). Esta capacidade não substitui o julgamento médico, mas otimiza o fluxo de atendimento, redirecionando casos críticos mais rapidamente para o cuidado especializado.

Outra área de aplicação relevante é o monitoramento contínuo de pacientes internados. Sensores conectados a plataformas de IA permitem o acompanhamento em tempo real de parâmetros como frequência cardíaca, saturação de oxigênio e pressão arterial, com geração automática de alertas quando valores se afastam de limiares pré-estabelecidos. Sistemas como o de detecção de sepse baseado em IA têm demonstrado redução de mortalidade em ambientes de UTI, ao permitir intervenções antes que o quadro se torne crítico

 ©2026 SBC – Soc. Bras. de Computação

(YUAN et al., 2020).

O processo de verificação e atualização da lista de medicamentos de um paciente ao longo das transições de cuidado é uma etapa essencial para a segurança clínica e tem se beneficiado significativamente da automação. Isso é especialmente importante quando se considera que erros de medicação estão entre as causas mais frequentes de eventos adversos em ambientes hospitalares. Nesse cenário, agentes de inteligência artificial integrados a prontuários eletrônicos têm se mostrado ferramentas estratégicas. Esses sistemas conseguem analisar prescrições em tempo real, identificando incompatibilidades, duplicidades e possíveis erros de dosagem antes mesmo que o medicamento seja administrado ao paciente.

Estudos sobre sistemas de suporte à decisão clínica demonstram que essas tecnologias contribuem de forma significativa para a redução de erros de medicação e para o aumento da segurança do paciente (SUTTON et al., 2020).

5.10.2. Apoio à Decisão Médica

O apoio à decisão médica representa talvez o benefício mais complexo e transformador da IA clínica. Diferentemente da automação de tarefas rotineiras, os sistemas de suporte à decisão (Clinical Decision Support Systems — CDSS) atuam diretamente na esfera do raciocínio diagnóstico e terapêutico, análise de exames e formulação de hipóteses clínicas. Seu objetivo central não é substituir o médico, mas ampliar a capacidade cognitiva do profissional diante de cenários de alta complexidade e volume massivo de dados (OBERMEYER; EMANUEL, 2016). Modelos baseados em redes neurais profundas (deep learning) vêm apresentando resultados expressivos na análise de imagens médicas. Na dermatologia, por exemplo, sistemas de inteligência artificial já conseguem identificar melanomas malignos com sensibilidade comparável à de dermatologistas experientes, a partir de imagens dermatoscópicas (ESTEVA et al., 2017).

Na radiologia, modelos voltados à detecção de nódulos pulmonares em tomografias computadorizadas alcançam altas taxas de verdadeiro-positivo, frequentemente superiores a 94%, mantendo baixos índices de falsos positivos e contribuindo para o diagnóstico precoce do câncer de pulmão. De forma semelhante, na oftalmologia, algoritmos aplicados à análise de imagens de fundo de olho têm demonstrado elevada precisão na classificação de doenças como retinopatia diabética e degeneração macular, em alguns casos superando o desempenho de especialistas (GULSHAN et al., 2016).

Para além da análise de imagens, os sistemas de IA também se destacam pela capacidade de integrar diferentes tipos de dados clínicos, como informações laboratoriais, genômicas, farmacológicas e históricas do paciente, a fim de apoiar decisões terapêuticas mais personalizadas. No contexto da oncologia de precisão, plataformas como o *Watson for Oncology*[41], desenvolvido pela IBM, assim como sistemas acadêmicos criados em centros de pesquisa, têm demonstrado alta concordância com diretrizes clínicas consolidadas, chegando a recomendar tratamentos consistentes em até 96% dos casos analisados (SOMASHEKHAR et al., 2018).

Esse desempenho está diretamente relacionado à capacidade desses sistemas de

[41] https://www.ibm.com/mysupport/s/topic/0TO500000002PWlGAM/watson-for-oncology?language=en_US

 ©2026 SBC – Soc. Bras. de Computação

analisar simultaneamente um grande volume de variáveis clínicas algo que, na prática, ultrapassa os limites do raciocínio humano sem suporte computacional. Outro ponto relevante diz respeito à redução de vieses cognitivos presentes no processo de decisão clínica. Tendências como o viés de disponibilidade em que diagnósticos recentes influenciam desproporcionalmente novas avaliações ou o fechamento prematuro do diagnóstico podem comprometer a qualidade da análise médica. Nesse sentido, sistemas de apoio à decisão clínica contribuem ao sugerir hipóteses alternativas baseadas em padrões estatísticos e dados populacionais, incentivando uma abordagem mais abrangente e menos suscetível a erros sistemáticos (GRABER, 2013).

5.10.3. Redução de Sobrecarga Administrativa

A sobrecarga administrativa representa um dos principais fatores de esgotamento profissional entre médicos e enfermeiros em todo o mundo. Estima-se que profissionais de saúde nos Estados Unidos dedicam entre 35% e 50% de sua jornada de trabalho a tarefas burocráticas, como documentação em prontuários, preenchimento de formulários de autorização, codificação de procedimentos e correspondência administrativa tempo que poderia ser direcionado ao cuidado direto ao paciente (SINSKY et al., 2016).

Agentes de processamento de linguagem natural (PLN) oferecem uma solução direta para esse problema, sendo capazes de transcrever consultas médicas em tempo real, estruturar automaticamente o prontuário eletrônico e codificar procedimentos nos padrões internacionais como CID-10 e SNOMED-CT. Ferramentas de documentação ambiental, como Nuance DAX e Suki AI, têm sido adotadas em sistemas de saúde norte-americanos e europeus, com evidências de redução significativa no tempo de documentação clínica e na carga de trabalho dos profissionais. Estudos apontam diminuições na ordem de 20% a 30% no tempo gasto com registros, além da redução de horas extras dedicadas à finalização de prontuários (HABERLE et al., 2024; RAZAGHI et al., 2026).

No contexto brasileiro, o Sistema Único de Saúde (SUS) enfrenta desafios adicionais relacionados ao volume de processos administrativos em instituições com recursos limitados. A implementação de agentes automáticos para triagem de solicitações de exames, autorização de internações e emissão de laudos tem potencial de agilizar significativamente o fluxo assistêencial, reduzindo filas e o tempo de espera para procedimentos eletivos (Brasil, Ministério da Saúde, 2022). As principais áreas de redução de carga administrativa por meio de agentes de IA incluem:

- Transcrição automática de consultas e geração de prontuários estruturados;
- Codificação automática de diagnósticos e procedimentos (CID-10, TUSS, SNOMED-CT);
- Processamento e roteamento inteligente de solicitações de autorização de planos de saúde;
- Geração automática de relatórios e sumários de alta hospitalar;
- Agendamento e reescalonamento inteligente de consultas e exames.

 ©2026 SBC – Soc. Bras. de Computação

O impacto da redução administrativa vai além da eficiência operacional estudos apontam correção direta entre menor carga burocrática e menor taxa de burnout entre profissionais de saúde, com reflexo na qualidade do cuidado prestado e na satisfação dos pacientes (LINZER et al., 2015).

5.10.4. Acesso Rápido à Literatura Científica

O volume de publicações científicas na área médica cresce de forma exponencial. Em 2023, a base de dados PubMed indexou mais de 36 milhões de artigos, com cerca de 1,3 milhão de novos trabalhos adicionados anualmente. Diante desse cenário, torna-se humanamente impossível para qualquer profissional da saúde manter-se atualizado em todas as subáreas relevantes para sua prática, criando uma lacuna entre o conhecimento científico disponível e a prática clínica cotidiana (BASTIAN; GLASZIOU; CHALMERS, 2010).

Agentes de IA baseados em modelos de linguagem de grande escala (LLMs) oferecem uma resposta concreta a esse desafio. Sistemas como o PubMedBERT[42], BioGPT[43] e ferramentas integradas a plataformas clínicas são capazes de realizar buscas na literatura, identificar evidências relevantes para uma pergunta clínica específica e sintetizar as principais conclusões de múltiplos estudos em linguagem natural, respeitando os princípios da medicina baseada em evidências (SINGHAL et al., 2023).

Uma das aplicações mais promissoras é a geração automática de revisões sistemáticas assistidas, nas quais o agente realiza a triagem de artigos, extrai dados padronizados de cada estudo e organiza os resultados em formatos compatíveis com as diretrizes PRISMA. Embora a validação humana continue indispensável, o auxílio automatizado reduz em semanas o tempo necessário para a conclusão de uma revisão abrangente (MARSHALL; WALLACE, 2019). No ambiente clínico, a integração de agentes de busca literária ao prontuário eletrônico permite que, no momento do atendimento, o profissional receba sugestões contextualizadas de diretrizes clínicas, meta-análises e protocolos baseados em evidências relacionados ao caso em tela.

5.10.5. Personalização no Atendimento ao Paciente

A medicina personalizada também denominada medicina de precisão parte do princípio de que cada paciente é único em sua constituição genômica, história clínica, estilo de vida e contexto social. Agentes de IA viabilizam, em escala, a aplicação desse paradigma, integrando dados heterogêneos de múltiplas fontes para construir perfis individuais e gerar recomendações altamente adaptadas a cada cenário particular (COLLINS; VARMUS, 2015). No âmbito da farmacogenômica, agentes de IA analisam o perfil genético do paciente para prever respostas a medicamentos específicos, identificar riscos de reações adversas e recomendar ajustes de dosagem individualizados. Estudos em oncologia demonstram que o tratamento guiado por análise genômica assistida por IA resulta em taxas de resposta superior à quimioterapia convencional não-dirigida, com menor toxicidade e melhor qualidade de vida (KRZYSZCZYK et al., 2018).

A personalização também envolve a forma de comunicação e o acompanhamento ao longo do tempo. Assistentes virtuais clínicos têm sido usados para monitorar pacientes

[42]

[43]https://the-decoder.com/biogpt-is-a-microsoft-language-model-trained-for-biomedical-tasks/

 ©2026 SBC – Soc. Bras. de Computação

com doenças crônicas fora do hospital, realizando contatos periódicos, identificando sinais iniciais de piora e incentivando o uso correto dos medicamentos por meio de lembretes e orientações adaptadas a cada paciente. Estudos com pessoas com diabetes e hipertensão mostram que o uso de chatbots clínicos pode melhorar a adesão ao tratamento e os resultados de saúde após seis meses (SHAN; SARKAR; MARTIN, 2019). Essa abordagem também aparece na adaptação da linguagem e do canal de comunicação às preferências do paciente.

Sistemas de IA conseguem identificar o nível de letramento em saúde do usuário e ajustar automaticamente a complexidade das informações, o que facilita a compreensão e aumenta o engajamento. Em populações com baixo letramento funcional comum em contextos de baixa renda essa capacidade pode contribuir diretamente para reduzir desigualdades no acesso à informação em saúde (MK, 2007). Por fim, essa abordagem orientada por agentes também se estende ao campo da prevenção. Com base em padrões de comportamento, dados de dispositivos vestíveis e histórico clínico, sistemas de IA podem elaborar planos preventivos mais ajustados a cada indivíduo, identificando fatores de risco e sugerindo mudanças de hábitos antes do surgimento de doenças crônicas. Essa mudança de foco do tratamento para a prevenção tem potencial para reduzir significativamente os custos em saúde no longo prazo, além de favorecer uma vida mais longa e com melhor qualidade (BHAVNANI; NARULA; SENGUPTA, 2016).

5.10.6. Bases de dados farmacológicos e de interações medicamentosas

Outro ponto muito importante nessa integração com bases de conhecimento é o acesso a informações sobre medicamentos. Existem bases como DrugBank, RxNorm e Micromedex que reúnem dados detalhados sobre milhares de remédios, incluindo para que servem, quando não devem ser usados e como interagem entre si.

Quando agentes de IA têm acesso a essas bases, eles conseguem atuar diretamente no momento da prescrição. Por exemplo, ao analisar os medicamentos que um paciente já utiliza, o agente pode identificar automaticamente possíveis interações perigosas e alertar o profissional de saúde antes que o problema aconteça. Isso complementa o que vimos antes: o agente não só utiliza evidências científicas (via RAG) e dados do paciente (via EHR), mas também aplica conhecimento farmacológico específico para aumentar a segurança do tratamento. Na prática, esse tipo de solução já está sendo utilizado.

A Santa Casa de Belo Horizonte implementou um sistema em que um agente de IA acompanha continuamente as prescrições médicas. Ele cruza essas informações com bases farmacológicas e gera alertas organizados por nível de risco. O resultado é um cuidado mais proativo: em vez de identificar problemas depois que eles acontecem, o sistema ajuda a prevenir eventos adversos, destacando casos críticos com antecedência — algo que, no modelo manual, poderia passar despercebido ou ser detectado tarde demais.

5.11. Desafios e Limitações

Apesar do potencial transformador demonstrado nas seções anteriores, a adoção de agentes de IA na saúde enfrenta desafios técnicos, éticos, regulatórios e operacionais significativos. A compreensão dessas limitações é essencial para garantir desenvolvimento responsável, segurança clínica e implementação sustentável dessas tecnologias em ambientes hospitalares

 ©2026 SBC – Soc. Bras. de Computação

e assistenciais.

5.11.1. Riscos de Alucinação em Modelos Generativos

O fenômeno das chamadas "alucinações" situações nas quais modelos generativos produzem informações factualmente incorretas apresentadas com aparente confiança representa um dos riscos mais críticos da aplicação de LLMs em saúde. Um modelo pode sugerir dosagens incorretas de medicamentos, inventar referências bibliográficas inexistentes ou recomendar condutas contraindicadas utilizando linguagem altamente convincente. Em ambientes clínicos, nos quais decisões equivocadas podem gerar danos diretos aos pacientes, essa limitação exige mecanismos robustos de mitigação. Entre as estratégias mais utilizadas destacam-se arquiteturas baseadas em *Retrieval-Augmented Generation* (RAG) com fontes controladas, camadas adicionais de validação, implementação de *guardrails* automatizados e supervisão humana obrigatória em decisões clínicas de maior impacto.

Estudos recentes demonstram que mesmo modelos considerados estado da arte, como GPT-4 e Med-PaLM 2, apresentam taxas mensuráveis de alucinação em tarefas médicas específicas. Nesse contexto, abordagens híbridas de mitigação têm se mostrado mais eficazes, incluindo recuperação de informações em bases verificadas, utilização de parâmetros de geração mais conservadores, instruções explícitas para indicação de incerteza e avaliação sistemática utilizando frameworks especializados como RAGAS[44] e TruLens[45].

5.11.2. Segurança de Dados e Conformidade com a LGPD

Dados de saúde figuram entre as informações mais sensíveis existentes. A integração de agentes inteligentes com prontuários eletrônicos, sistemas laboratoriais e bases genômicas levanta questões críticas relacionadas ao armazenamento, transmissão e processamento seguro dessas informações.

No Brasil, a Lei Geral de Proteção de Dados (LGPD)[46] classifica dados de saúde como dados pessoais sensíveis, impondo requisitos rigorosos para tratamento, armazenamento e compartilhamento dessas informações. Entre esses requisitos destacam-se necessidade de base legal adequada, princípios de minimização de dados, transparência no tratamento e garantia de segurança da informação. Vazamentos de dados médicos possuem consequências particularmente graves, podendo expor condições clínicas sensíveis, comprometer privacidade de pacientes e gerar impactos sociais, financeiros e psicológicos relevantes.

Além disso, órgãos reguladores como a ANVISA[47] e o Conselho Federal de Medicina (CFM)[48] têm avançado na definição de diretrizes específicas para sistemas de IA com aplicação clínica, incluindo requisitos relacionados à validação clínica, rastreabilidade de decisões e responsabilização institucional. Nesse cenário, conformidade regulatória deixa de ser apenas uma exigência jurídica e passa a representar elemento central para

[44] <https://github.com/explodinggradients/ragas>
[45] <https://www.trulens.org>
[46] <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm>
[47] <https://www.gov.br/anvisa/pt-br>
[48] <https://portal.cfm.org.br>

 ©2026 SBC – Soc. Bras. de Computação

confiança, segurança e adoção sustentável dessas tecnologias.

5.11.3. Viés Algorítmico e Equidade em Saúde

Modelos de inteligência artificial aprendem padrões a partir de dados históricos, os quais frequentemente refletem desigualdades estruturais presentes nos sistemas de saúde. Como consequência, algoritmos treinados predominantemente com determinadas populações podem apresentar desempenho inferior quando aplicados a grupos sub-representados no conjunto de treinamento.

Estudos demonstram que sistemas de predição de risco clínico podem apresentar diferenças significativas de desempenho relacionadas a raça, gênero, faixa etária e condição socioeconômica. Um caso amplamente discutido na literatura mostrou que um algoritmo utilizado nos Estados Unidos atribuía sistematicamente menor prioridade de cuidado a pacientes negros em comparação a pacientes brancos com gravidade clínica semelhante, perpetuando desigualdades já existentes no sistema de saúde.

No contexto brasileiro, essa questão torna-se particularmente relevante devido à elevada heterogeneidade populacional e às profundas desigualdades regionais e socioeconômicas. Dessa forma, o desenvolvimento de datasets representativos da população brasileira, aliado à validação específica em grupos vulneráveis, constitui requisito fundamental para adoção ética e segura de IA em saúde pública.

5.11.4. Limites da Automação e Responsabilidade Profissional

A crescente automação de processos clínicos levanta questões fundamentais relacionadas à responsabilidade profissional e ao papel do julgamento humano na tomada de decisão médica. Segundo diretrizes do Conselho Federal de Medicina, a responsabilidade final pela conduta clínica permanece sempre sob responsabilidade do profissional médico, independentemente do suporte tecnológico utilizado. Isso implica que agentes inteligentes devem ser concebidos como ferramentas de apoio à decisão clínica, e não como substitutos do raciocínio médico. Além disso, profissionais de saúde precisam ser adequadamente treinados para compreender tanto capacidades quanto limitações dos sistemas baseados em IA.

Outro risco relevante é o fenômeno conhecido como "automação excessiva", no qual profissionais passam a aceitar recomendações automatizadas sem avaliação crítica adequada. Estudos relacionados à chamada *alert fatigue* demonstram que excesso de notificações e alertas pode levar usuários a ignorarem sistematicamente mensagens do sistema inclusive aquelas realmente importantes. Para mitigar esses problemas, tornam-se essenciais estratégias como design cuidadoso de interfaces, priorização inteligente de alertas, mecanismos de explicabilidade e construção de cultura institucional voltada ao uso crítico e supervisionado das ferramentas de IA.

5.11.5. Barreiras de Adoção e Infraestrutura

A adoção prática de agentes inteligentes na saúde também enfrenta barreiras estruturais significativas, especialmente em ambientes com recursos limitados. Sistemas baseados em IA dependem fortemente da qualidade, completude e padronização dos dados clínicos disponíveis. Entretanto, muitos hospitais ainda operam com prontuários parcialmente

 ©2026 SBC – Soc. Bras. de Computação

digitalizados ou sistemas fragmentados sem interoperabilidade adequada.

Além disso, a infraestrutura computacional necessária para treinamento e execução de modelos de linguagem de larga escala pode representar custo elevado para instituições de saúde. A utilização de GPUs especializadas, armazenamento seguro de grandes volumes de dados e integração contínua com sistemas hospitalares demandam investimentos financeiros e técnicos substanciais.

Nesse contexto, arquiteturas híbridas têm emergido como alternativa pragmática. Essas abordagens combinam processamento local (*on-premises*) para dados mais sensíveis com utilização de serviços em nuvem para tarefas de maior demanda computacional, buscando equilibrar desempenho, custo operacional e privacidade das informações clínicas.

5.12. Implementação Estudo de Caso

Esta seção apresenta um estudo de caso clínico hipotético com o objetivo de ilustrar, de forma concreta e detalhada, como um sistema multi-agente de IA operaria na prática em um contexto de urgência e emergência. O cenário é fictício, mas clinicamente plausível, construído com base em protocolos reais de atendimento de urgência.

5.12.1. Visão Geral da Arquitetura do Sistema

O sistema proposto é composto por três agentes especializados que operam de forma coordenada, cada um com escopo e responsabilidades bem definidos. O Agente de Triagem é responsável pela recepção da queixa inicial, classificação de gravidade e encaminhamento interno. O Agente Especialista realiza o raciocínio diagnóstico aprofundado, recupera diretrizes clínicas relevantes via RAG e formula hipóteses e condutas fundamentadas em evidências. O Agente de Documentação captura o raciocínio clínico produzido ao longo da interação e gera automaticamente o registro estruturado no prontuário eletrônico.

A orquestração entre os agentes segue um grafo de decisão implementado em LangGraph. O Agente de Triagem atua como ponto de entrada, avalia o nível de urgência e determina se o caso deve ser resolvido localmente, encaminhado ao Agente Especialista ou escalado imediatamente para atendimento presencial de emergência. O Agente de Documentação opera em paralelo, monitorando toda a troca de informações e consolidando o registro ao final do atendimento. O médico mantém controle total em cada etapa, e o sistema nunca executa ações clínicas sem que haja confirmação ou ao menos ciência do profissional responsável.

5.12.2. Contexto do Cenário

O cenário envolve um clínico geral que atende em consultório particular em uma cidade de médio porte no interior do Piauí. O consultório dispõe de recursos diagnósticos básicos, incluindo oxímetro, aparelho de pressão, glicosímetro e eletrocardiógrafo portátil, além de acesso limitado a especialistas presenciais. O médico utiliza um sistema de prontuário eletrônico integrado à plataforma multiagente, acessível por interface web durante a consulta.

O paciente é um homem de 61 anos, hipertenso em uso de losartana 50 mg por dia e metformina 850 mg duas vezes ao dia para diabetes tipo 2, tabagista de 30 maços ano,

 ©2026 SBC – Soc. Bras. de Computação

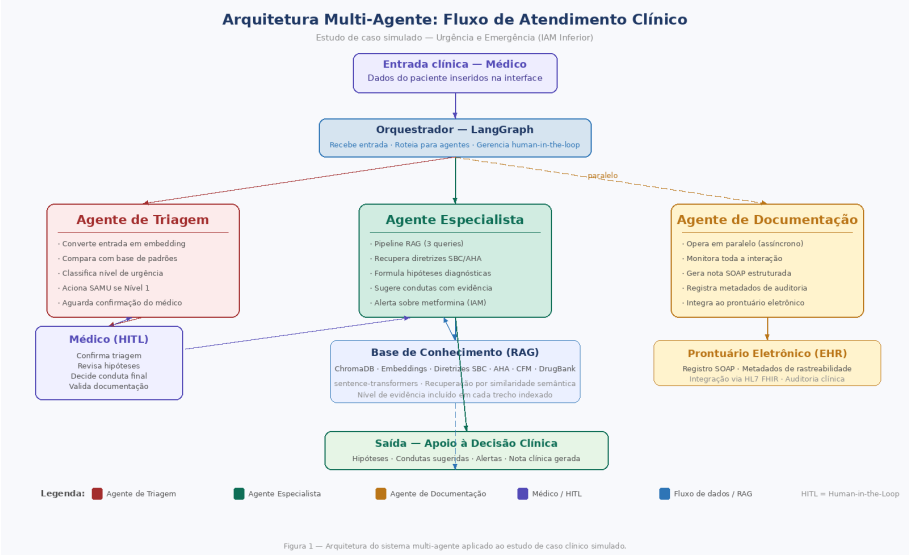


Figura 5.6. Diagrama Multi-agente

que chega ao consultório acompanhado da esposa às 10h47 com queixa de dor no peito de início súbito há aproximadamente 40 minutos, irradiando para o braço esquerdo, associada a sudorese fria e discreta falta de ar. O médico insere essas informações na interface do sistema e aciona o pipeline multiagente.

5.12.3. Etapa 1 Agente de Triagem

Ao receber a descrição clínica, o Agente de Triagem converte o texto em embedding e o compara com sua base de padrões de triagem indexados. Em menos de dois segundos, o agente identifica a combinação de dor torácica irradiada, sudorese e dispneia em paciente com perfil de risco cardiovascular elevado como um padrão de alta probabilidade para síndrome coronariana aguda (SCA). Com base nos protocolos do Manchester Triage System e nas diretrizes da Sociedade Brasileira de Cardiologia, o agente classifica o caso como Nível 1 Emergência Cardiovascular Provável, recomenda o acionamento imediato do SAMU (192) e solicita confirmação do médico antes de acionar o Agente Especialista. Simultaneamente, o sistema exibe um lembrete automatizado para checagem imediata de saturação de oxigênio, pressão arterial e realização do ECG portátil procedimentos iniciais recomendados enquanto o raciocínio diagnóstico aprofundado é processado em paralelo.

Além da classificação inicial, o agente também realiza uma análise contextual do histórico clínico previamente registrado no prontuário eletrônico. Informações como hipertensão arterial, diabetes mellitus, tabagismo, histórico familiar de infarto agudo do miocárdio e uso contínuo de medicamentos cardiovasculares são automaticamente incorporadas ao processo de priorização clínica. Essa etapa permite aumentar a sensibilidade da triagem e reduzir o risco de subestimação de casos críticos.

5.12.4. Etapa 2 Agente Especialista

Com a confirmação do médico, o Agente Especialista recebe o caso completo incluindo a classificação do Agente de Triagem, o histórico do paciente e os dados inseridos e

 ©2026 SBC – Soc. Bras. de Computação



Figura 5.7. Agente de Triagem

inicia o pipeline RAG. O agente formula três queries semânticas para recuperação de documentos a primeira buscando diretrizes de manejo de SCA em atenção primária e consultório, a segunda focando em contraindicações e cuidados com antiagregação em diabéticos hipertensos e a terceira recuperando protocolos de estabilização pré-hospitalar para encaminhamento de emergência. O sistema recupera trechos relevantes da Diretriz Brasileira de Síndromes Coronarianas Agudas (SBC, 2021), do Protocolo de Suporte Avançado de Vida Cardiovascular da American Heart Association adaptado para cenários de baixa complexidade, e das orientações do CFM sobre responsabilidades do médico assistente em situações de emergência.

Com base nesses documentos todos com nível de evidência A, o Agente Especialista formula três hipóteses diagnósticas em ordem decrescente de probabilidade IAMCSST como hipótese principal, Angina Instável / IAMSST como segunda hipótese, e Dissecção Aórtica como diagnóstico diferencial a ser excluído. O agente sugere condutas de suporte enquanto o SAMU é acionado: AAS 200–300 mg VO mastigado (na ausência de contraindicação), nitroglicerina sublingual 0,5 mg se pressão arterial sistólica acima de 90 mmHg, oxigênio suplementar se SpO abaixo de 94%, acesso venoso periférico,

 ©2026 SBC – Soc. Bras. de Computação

posicionamento semissentado e monitorização contínua. O agente emite ainda um alerta crítico a metformina em uso pelo paciente deve ser suspensa imediatamente diante da suspeita de IAM, dado o risco de acidose lática em contexto de hipoperfusão tecidual informação que poderia ser facilmente negligenciada na urgência do atendimento.

O médico revisa as sugestões, decide administrar AAS 300 mg mastigado e inicia o ECG portátil. O traçado evidencia supradesnivelamento de ST em DII, DIII e aVF, compatível com IAM inferior. O médico registra esse achado na interface e aciona o SAMU. O Agente Especialista, ao receber a atualização do ECG, complementa seu raciocínio identificando o padrão como sugestivo de oclusão da artéria coronária direita e recomenda antecipar o relato ao serviço de hemodinâmica referenciado o sistema recupera automaticamente o protocolo de ativação da sala de hemodinâmica do hospital de referência regional e o exibe ao médico.

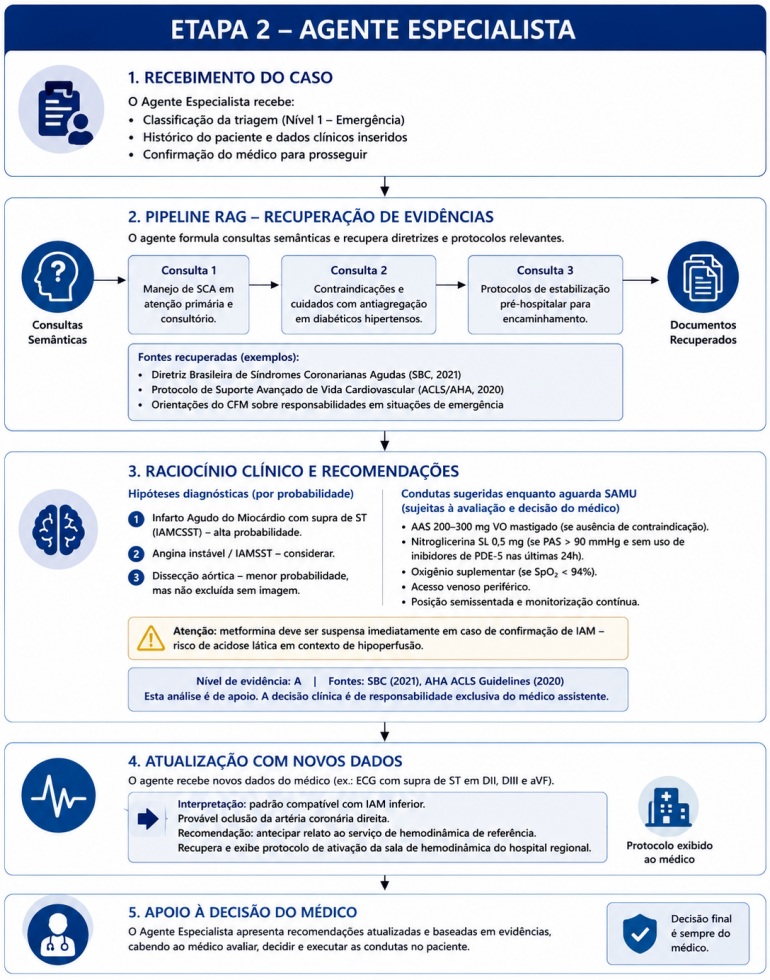


Figura 5.8. Agente Especialista

5.12.5. Etapa 3 Agente de Documentação

Enquanto o médico conduz o atendimento, o Agente de Documentação opera em segundo plano monitorando toda a interação. Ao término do atendimento, com a chegada do SAMU

 ©2026 SBC – Soc. Bras. de Computação

e transferência do paciente, o agente consolida automaticamente um registro clínico estruturado no prontuário eletrônico, organizado no formato SOAP. O registro inclui informações subjetivas relacionadas à queixa e à história clínica, dados objetivos como sinais vitais e achados do ECG, avaliação contendo hipóteses diagnósticas e raciocínio clínico, além do plano terapêutico com medicamentos administrados, encaminhamentos realizados, serviço acionado e horários registrados.

O registro gerado também incorpora metadados de rastreabilidade, incluindo quais documentos foram recuperados pelo pipeline RAG, quais sugestões foram aceitas ou descartadas pelo médico e o horário correspondente a cada etapa do atendimento. Esse nível de documentação automatizada contribui para a continuidade do cuidado no serviço receptor, oferece suporte à auditoria clínica e produz evidências objetivas do raciocínio clínico empregado em eventuais análises jurídicas ou administrativas.

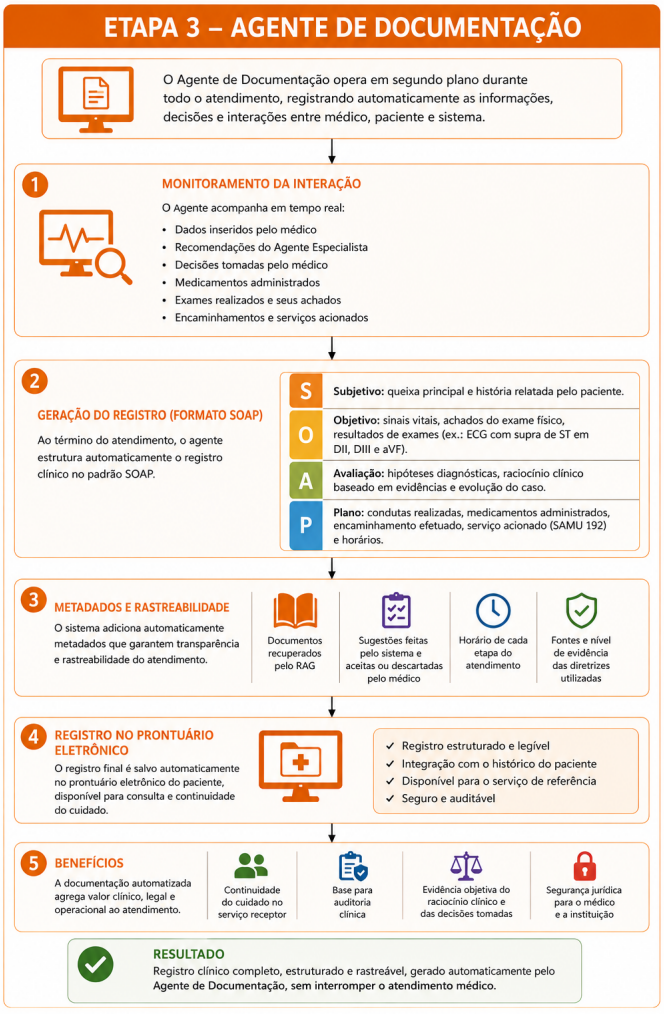


Figura 5.9. Agente de Documentação

5.12.6. Análise do Fluxo Multi-agente

O caso ilustra as vantagens da arquitetura multiagente em relação a soluções monolíticas. A especialização de responsabilidades entre triagem, raciocínio diagnóstico e documentação

 ©2026 SBC – Soc. Bras. de Computação

permite que cada componente seja otimizado, atualizado e auditado de forma independente. Além disso, a paralelização entre o Agente Especialista e o Agente de Documentação reduz significativamente a latência operacional que existiria caso ambas as funções fossem executadas sequencialmente pelo mesmo sistema.

O princípio de *human in the loop* é preservado em todas as transições críticas do fluxo. O Agente de Triagem aguarda confirmação antes de acionar o Agente Especialista. As condutas sugeridas pelo Agente Especialista são apresentadas explicitamente como recomendações de apoio à decisão clínica. O Agente de Documentação registra não apenas as recomendações fornecidas pelo sistema, mas também as decisões efetivamente adotadas pelo profissional responsável. Essa distinção possui relevância ética, regulatória e jurídica.

Do ponto de vista do desempenho, o tempo total de processamento entre a inserção inicial dos dados e a apresentação do raciocínio clínico pelo Agente Especialista é estimado em menos de oito segundos em condições normais de conectividade. A documentação automatizada ocorre de forma assíncrona e não interfere diretamente na dinâmica do atendimento médico.

5.12.7. Limitações e Considerações do Cenário

É importante reconhecer as limitações inerentes ao estudo de caso apresentado. Por se tratar de um cenário simulado, os tempos de resposta, a qualidade dos documentos recuperados e o comportamento do sistema em situações limite, como perguntas fora do escopo da base de conhecimento ou dados clínicos contraditórios, ainda não foram empiricamente validados. A validação clínica formal do sistema, envolvendo especialistas e ambientes reais de atendimento, constitui etapa indispensável antes de qualquer implantação prática.

Além disso, o cenário pressupõe conectividade estável e integração com prontuário eletrônico dotado de API compatível, condições que nem sempre estão disponíveis em consultórios do interior do Brasil. Dessa forma, o desenvolvimento de modos de operação degradada, capazes de manter funcionalidades essenciais mesmo em ambientes de baixa conectividade, representa um requisito técnico importante para as próximas etapas do projeto.

5.13. Conclusão

Os agentes de Inteligência Artificial representam uma das fronteiras mais promissoras da computação aplicada à saúde. Ao combinar a capacidade generativa dos grandes modelos de linguagem com técnicas de recuperação de informação, integração padronizada com sistemas externos por meio de protocolos como o MCP, e arquiteturas multiagentes capazes de distribuir responsabilidades especializadas, esses sistemas abrem possibilidades inéditas para o suporte à decisão clínica, a automação de processos e a personalização do cuidado.

Ao longo deste minicurso, percorremos o caminho que vai dos fundamentos conceituais a definição de agente inteligente, a evolução histórica da IA na saúde, a distinção entre chatbots e agentes autônomos até os desafios técnicos, éticos e regulatórios que condicionam uma adoção responsável dessas tecnologias. Exploramos em profundidade exemplos reais de referência global, como a Ada Health com sua triagem probabilística adaptativa, o Google DeepMind com o AlphaFold e o Med-PaLM 2, a Infermedica com

 ©2026 SBC – Soc. Bras. de Computação

sua plataforma multilíngue, o DAX Copilot com a automação de documentação clínica e a Tempus AI com a medicina de precisão em oncologia. No contexto brasileiro, o HStory do Hospital Israelita Albert Einstein demonstra que o país também produz inovação de alto nível nessa área.

A mensagem central que este material busca transmitir é de equilíbrio: entre o entusiasmo com o potencial transformador dessas tecnologias e a responsabilidade que seu uso em contextos de saúde demanda. Agentes de IA não substituem o julgamento clínico; eles o ampliam. Não eliminam a necessidade de profissionais qualificados; reduzem sua carga cognitiva e administrativa. E não resolvem por si sós as desigualdades do sistema de saúde, mas podem ser ferramentas poderosas na direção de um cuidado mais equitativo, acessível e baseado em evidências.

O futuro imediato da IA na saúde será moldado pela capacidade do setor de estabelecer frameworks de governança robustos, datasets representativos da diversidade da população, protocolos de validação clínica rigorosos e culturas institucionais que promovam o uso crítico, e não a dependência acrítica, dessas ferramentas. Profissionais de saúde, desenvolvedores de tecnologia, reguladores e pacientes precisam participar conjuntamente desse processo de construção. Este minicurso é uma contribuição para a formação dos primeiros aqueles que, equipados com compreensão técnica e consciência ética, serão os protagonistas dessa transformação.

Referências

ALOMARI, L. M. et al. Safety and accuracy of ai in triaging patients in the emergency department. *International Journal of Emergency Medicine*, Springer, v. 18, n. 1, p. 243, 2025. Citado na página page.2020.

ANTHROPIC. *Model Context Protocol Introduction*. 2024. <https://modelcontextprotocol.io/introduction>. Acesso em: 10 maio 2026. Citado na página page.33.

Associação Nacional de Hospitais Privados (Anahp). *A Inteligência Artificial já está sendo usada na saúde: descubra como isso te beneficia*. 2025. Publicado com base nos dados apresentados no Showcase de IA nos Hospitais Brasileiros. Disponível em: <https://www.anahp.com.br/saude-da-saude/a-inteligencia-artificial-ja-esta-sendo-usada-na-saude-descubra-como-isso-te-beneficia/>. Citado na página page.22.

BASTIAN, H.; GLASZIOU, P.; CHALMERS, I. Seventy-five trials and eleven systematic reviews a day: how will we ever keep up? *PLoS medicine*, Public Library of Science San Francisco, USA, v. 7, n. 9, p. e1000326, 2010. Citado na página page.2323.

BHAVNANI, S. P.; NARULA, J.; SENGUPTA, P. P. Mobile technology and the digitization of healthcare. *European heart journal*, v. 37, n. 18, p. 1428, 2016. Citado na página page.2424.

COLLINS, F. S.; VARMUS, H. A new initiative on precision medicine. *New England journal of medicine*, Mass Medical Soc, v. 372, n. 9, p. 793–795, 2015. Citado na página page.2323.

 ©2026 SBC – Soc. Bras. de Computação

Distrito; Associação Brasileira de Startups de Saúde e HealthTechs (ABSS). *HealthTech Recap 2024*. [S.l.], 2025. Relatório lançado em fevereiro de 2025 com dados do exercício de 2024. Disponível em: <https://materiais.distrito.me/report/healthtech-report-recap-2024>. Citado na página page.22.

ESTEVA, A. et al. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, Nature Publishing Group UK London, v. 542, n. 7639, p. 115–118, 2017. Citado na página page.2121.

ESTEVA, A. et al. A guide to deep learning in healthcare. *Nature medicine*, Nature Publishing Group US New York, v. 25, n. 1, p. 24–29, 2019. Citado na página page.22.

GOODFELLOW, I. et al. *Deep learning*. [S.l.]: MIT press Cambridge, 2016. v. 1. Citado na página page.22.

GRABER, M. L. The incidence of diagnostic error in medicine. *BMJ quality & safety*, BMJ Publishing Group Ltd, v. 22, n. Suppl 2, p. ii21–ii27, 2013. Citado na página page.2222.

GULSHAN, V. et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *jama*, American Medical Association, v. 316, n. 22, p. 2402–2410, 2016. Citado na página page.2121.

HABERLE, T. et al. The impact of nuance dax ambient listening ai documentation: a cohort study. *Journal of the American Medical Informatics Association*, Oxford University Press, v. 31, n. 4, p. 975–979, 2024. Citado na página page.2222.

HAN, J.; KAMBER, M.; PEI, J. *Data Mining: Concepts and Techniques*. 3. ed. Burlington, MA: Morgan Kaufmann, 2011. ISBN 9780123814791. Disponível em: <https://www.sciencedirect.com/book/9780123814791/data-mining-concepts-and-techniques>. Citado na página page.33.

KRZYSZCZYK, P. et al. The growing role of precision and personalized medicine for cancer treatment. *Technology*, World Scientific, v. 6, n. 03n04, p. 79–100, 2018. Citado na página page.2323.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015. Disponível em: <https://www.nature.com/articles/nature14539>. Citado na página page.33.

LEWIS, P. et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, v. 33, p. 9459–9474, 2020. Disponível em: <https://arxiv.org/abs/2005.11401>. Citado na página page.33.

LINZER, M. et al. A cluster randomized trial of interventions to improve work conditions and clinician burnout in primary care: results from the healthy work place (hwp) study. *Journal of general internal medicine*, Springer, v. 30, n. 8, p. 1105–1111, 2015. Citado na página page.2323.

 ©2026 SBC – Soc. Bras. de Computação

LIPPMANN, R. P. An introduction to computing with neural nets. *ACM SIGARCH Computer Architecture News*, ACM New York, NY, USA, v. 16, n. 1, p. 7–25, 1988. Citado na página page.33.

MARSHALL, I. J.; WALLACE, B. C. Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. *Systematic reviews*, Springer, v. 8, n. 1, p. 163, 2019. Citado na página page.2323.

MILLER, R. A.; JR, H. E. P.; MYERS, J. D. Internist-i, an experimental computer-based diagnostic consultant for general internal medicine. In: *Computer-assisted medical decision making*. [S.l.]: Springer, 1985. p. 139–158. Citado na página page.33.

MK, P.-O. The causal pathways linking health literacy to health outcomes. *Am J Health Behav.*, v. 31, n. 1, p. S19–S26, 2007. Citado na página page.2424.

MÜLLER, J. P. Architectures and applications of intelligent agents: A survey. *The Knowledge Engineering Review*, Cambridge University Press, v. 13, n. 4, p. 353–380, 1999. Citado na página page.55.

Núcleo de Informação e Coordenação do Ponto BR (NIC.br). *Pesquisa sobre o uso das tecnologias de informação e comunicação nos estabelecimentos de saúde brasileiros: TIC Saúde 2024*. São Paulo, 2024. Entrevistas realizadas entre fevereiro e agosto de 2024 com 2.057 gestores e 2.021 profissionais de saúde em todo o território nacional. Disponível em: <https://cetic.br/pt/tics/saude/2024/medicos/>. Citado na página page.22.

OBERMEYER, Z.; EMANUEL, E. J. Predicting the future—big data, machine learning, and clinical medicine. *The New England journal of medicine*, v. 375, n. 13, p. 1216, 2016. Citado na página page.2121.

RAZAGHI, M. et al. Transforming clinical documentation with ambient artificial intelligence (ai) scribes: a narrative review of technology, impact, and implementation. *Cardiovascular Diagnosis and Therapy*, LWW, v. 16, n. 1, p. 11, 2026. Citado na página page.2222.

REZENDE, S. O. *Sistemas inteligentes: fundamentos e aplicações*. [S.l.]: Editora Manole Ltda, 2003. Citado na página page.22.

RUSSELL, S.; NORVIG, P.; INTELLIGENCE, A. A modern approach. *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs*, v. 25, n. 27, p. 79–80, 1995. Citado 2 vezes nas páginas page.22 e page.55.

SHAN, R.; SARKAR, S.; MARTIN, S. S. Digital health technology and mobile devices for the management of diabetes mellitus: state of the art. *Diabetologia*, Springer, v. 62, n. 6, p. 877–887, 2019. Citado na página page.2424.

SHORTLIFFE, E. *Computer-based medical consultations: MYCIN*. [S.l.]: Elsevier, 2012. v. 2. Citado na página page.33.

SINGHAL, K. et al. Large language models encode clinical knowledge. *Nature*, Nature Publishing Group UK London, v. 620, n. 7972, p. 172–180, 2023. Citado na página page.2323.

 ©2026 SBC – Soc. Bras. de Computação

SINSKY, C. et al. Allocation of physician time in ambulatory practice: a time and motion study in 4 specialties. *Annals of internal medicine*, American College of Physicians, v. 165, n. 11, p. 753–760, 2016. Citado na página page.2222.

SOMASHEKHAR, S. P. et al. Watson for oncology and breast cancer treatment recommendations: agreement with an expert multidisciplinary tumor board. *Annals of Oncology*, Elsevier, v. 29, n. 2, p. 418–423, 2018. Citado na página page.2121.

SUTTON, R. T. et al. An overview of clinical decision support systems: benefits, risks, and strategies for success. *NPJ digital medicine*, Nature Publishing Group UK London, v. 3, n. 1, p. 17, 2020. Citado na página page.2121.

TOPOL, E. J. High-performance medicine: the convergence of human and artificial intelligence. *Nature medicine*, Nature Publishing Group US New York, v. 25, n. 1, p. 44–56, 2019. Citado 2 vezes nas páginas page.22 e page.2020.

VASWANI, A. et al. Attention is all you need. In: *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. v. 30, p. 5998–6008. Disponível em: <https://papers.nips.cc/paper/7181-attention-is-all-you-need>. Citado 2 vezes nas páginas page.33 e page.88.

WOOLDRIDGE, M. *An introduction to multiagent systems*. [S.l.]: John wiley & sons, 2009. Citado na página page.22.

YUAN, K.-C. et al. The development an artificial intelligence algorithm for early sepsis diagnosis in the intensive care unit. *International journal of medical informatics*, Elsevier, v. 141, p. 104176, 2020. Citado na página page.2121.

 ©2026 SBC – Soc. Bras. de Computação

Capítulo 6

Aprendizado de Máquina Aplicado à Descoberta de Fármacos

Karina dos Santos Machado, Adriano Velasque Werhli,
Frederico Kremer, Rafael Junqueira Borges

Abstract

Drug discovery is a long, complex, and expensive process, motivating the development of computational strategies capable of accelerating different stages of this pipeline. In this context, Machine Learning (ML) and Artificial Intelligence (AI) have played an important role in applications such as protein structure prediction, co-folding, virtual screening, molecular docking, and scoring function development. This chapter presents the main theoretical foundations of ML methods applied to rational drug discovery, covering concepts of structural bioinformatics, molecular data representation, classical supervised models, and modern deep learning-based approaches. In addition, collaborative open science initiatives such as CACHE and DREAM Challenge are discussed, highlighting their role in the development and evaluation of computational methods under realistic scenarios.

Resumo

A descoberta de fármacos é um processo longo, complexo e de alto custo, impulsionando o desenvolvimento de estratégias computacionais capazes de acelerar diferentes etapas desse pipeline. Nesse contexto, o Aprendizado de Máquina (AM) e a Inteligência Artificial (IA) têm desempenhado um papel cada vez mais relevante em aplicações como predição estrutural de proteínas e complexos, VS, docking molecular, desenvolvimento de FEs. Este capítulo apresenta os principais fundamentos teóricos de métodos de AM aplicados à descoberta racional de fármacos, abordando conceitos de bioinformática estrutural, representação de dados moleculares, modelos supervisionados clássicos e abordagens modernas baseadas em aprendizado profundo. Além disso, são discutidas iniciativas colaborativas de ciência aberta, como CACHE e DREAM Challenge, que têm impulsionado o desenvolvimento e avaliação de métodos computacionais em cenários realistas.

 ©2026 SBC – Soc. Bras. de Computação

6.1. Introdução

A necessidade de inovação na descoberta de novos fármacos é urgente, especialmente diante de problemas de saúde pública de escala global, como pandemias, resistência antimicrobiana, crescimento anual dos casos de câncer e doenças genéticas ainda sem tratamento conhecido [Murray et al. 2022, Bray et al. 2024, Kernohan and Boycott 2024]. Todos esses cenários destacam a necessidade urgente de desenvolver novas estratégias para a busca de fármacos. Apesar de todos esses problemas de saúde mundial, a proposta de um novo fármaco ainda é um processo longo e caro. Estima-se que sejam necessários 10-15 anos para que uma nova droga seja aprovada e um investimento médio é de aproximadamente 2,3 bilhões de dólares. Como alternativa para acelerar e reduzir custos desse processo, o uso de ferramentas computacionais tem se mostrado promissor.

Recentemente, avanços significativos foram alcançados pela Ciência da Computação - principalmente em **Inteligência Artificial** (IA) - e pela área de Desenho Racional de fármacos (*Rational Drug Design* - RDD). O RDD pode ser categorizado em dois tipos: planejamento de fármacos baseado na estrutura do ligante (*Ligand-Based Drug Design* - LBDD) e planejamento de fármacos baseado na estrutura do receptor (*Structure-Based Drug Design* - SBDD) [Lima et al. 2016]. A abordagem SBDD, o foco desse capítulo, é baseada no conhecimento da estrutura tridimensional do alvo biológico, usualmente uma proteína (receptor) e como ele interage com o(s) candidato(s) a inibidor (ligante).

Em 2024, os avanços de IA no SBDD foram notáveis, destacando-se os prêmios Nobel de Física para G. Hinton e J. Hopfiled pela proposta das Redes Neurais Artificiais e o prêmio Nobel de Química, para D. Baker pelo método computacional para projetar proteínas *de novo*, além de D. Hassabis e J. Jumper pela proposta do algoritmo de predição de estrutura de proteínas Alphafold2 [Xu 2024, Jumper et al. 2021]. Esses avanços têm permitido o desenvolvimento de métodos aplicados na indústria farmacêutica durante os primeiros estágios de descoberta de um novo fármaco [Kuntz 1992, Meng et al. 2011].

Nesse contexto, a **triagem virtual** (*Virtual Screening* - VS) é uma técnica computacional que permite avaliar e priorizar grandes bibliotecas de compostos para identificar potenciais candidatos a fármacos, geralmente utilizando ***docking*** (docagem) **molecular**. O *docking* molecular consiste em uma simulação em nível atômico, onde milhares de orientações e conformações do ligante no sítio de ligação do receptor são avaliadas e ranqueadas de acordo com a estabilidade do complexo estimada por uma **energia livre de ligação** (*Free Energy of Binding* - FEB) ou escore, predita/calculada por uma função de escore (FE) [Lybrand 1995, Crampon et al. 2022] .

Os desafios e avanços na área de VS com uso de **Aprendizado de Máquina** (AM) têm sido destacados em iniciativas como o **Challenge** [Li et al. 2024], uma competição que explora o uso de IA em VS para busca de novas moléculas promissoras para alvos de doenças complexas. Nas duas primeiras edições do **CACHE**, foram abordadas doença como Parkinson (#1) e Covid-19 (#2), os quais os participantes utilizaram estratégias de AM para selecionar compostos candidatos promissores. No Cache Challenge #2, os participantes tinham que encontrar moléculas promissoras para a proteína NSP13 do Sars-Cov-2. A equipe composta pelos proponentes deste minicurso foi a vencedora entre 22 times participantes. Durante o desafio que ocorreu de 2022 a 2024, foram selecionadas 46 moléculas promissoras determinadas após ex-

 ©2026 SBC – Soc. Bras. de Computação

tensivo conjunto de testes experimentais, sendo 8 dessas indicadas pelo nosso grupo [Herasymenko et al. 2025, CACHE Initiative 2024]. Nosso método combinou o uso de ferramentas *open-source* e métodos de AM, sendo a única equipe do Sul Global e a única liderada por uma mulher.

Há também o **DREAM Challenge**, uma iniciativa dentro do consórcio **MAINFRAME** (*MAchine learning Innovation Network For Research to Advance MEdicinal chemistry*) [Edwards et al. 2025] que promove competições abertas voltadas à aplicação de AM e IA na descoberta de fármacos. Essa plataforma segue os princípios da ciência aberta, oferecendo grandes conjuntos de dados experimentais e computacionais sobre proteínas, ligantes e resultados de ensaios biológicos para que equipes do mundo inteiro possam propor e comparar modelos preditivos. O objetivo é acelerar a inovação na área de descoberta de fármacos baseada em dados, estimulando a reprodutibilidade científica, a colaboração entre grupos acadêmicos e industriais, e a criação de soluções de IA transparentes e acessíveis. Além disso, o DREAM Challenge/MAINFRAME busca integrar dados heterogêneos de diferentes fontes para permitir o desenvolvimento de modelos robustos e generalizáveis, consolidando-se como um ambiente de referência para o uso ético e colaborativo de IA em bioinformática e saúde.

Além das competições, Xu *et al*. destacam uma lista de exemplos de descoberta de fármacos baseados em IA, já em estudo clínico, mostrando o sucesso desse tipo de estratégia [Xu 2024]. No trabalho publicado recentemente, Xu *et al*. relatam o desenvolvimento do rentosertib, um inibidor da quinase TNIK identificado por meio de algoritmos de IA generativa, desde o desenho molecular até a otimização estrutural e seleção de candidatos para ensaio clínico [Xu et al. 2025]. Uma das moléculas selecionadas neste trabalho tem apresentado resultados clínicos promissores em pacientes com fibrose pulmonar idiopática (IPF). Esse é um dos primeiros casos documentados em que um fármaco proposto a partir de diferentes algoritmos de IA avança até a fase 2 de testes clínicos, evidenciando que modelos generativos e *pipelines* baseados em AM podem produzir candidatos farmacologicamente eficazes e seguros.

Nos últimos anos, o avanço de arquiteturas de **aprendizado profundo**, **modelos generativos** e **modelos de linguagem de larga escala** (*Large Language Models* – LLMs) tem ampliado significativamente as possibilidades da descoberta de fármacos assistida por IA. Essas abordagens têm sido aplicadas em diferentes etapas do *pipeline* de desenvolvimento de fármacos, incluindo predição de estrutura de proteínas-alvo, geração de novas moléculas, VS, desenvolvimento de FEs e predição de propriedades farmacocinéticas e toxicológicas [Jumper et al. 2021, Abramson et al. 2024, Crampon et al. 2022, Xu 2024]. Além disso, a disponibilidade crescente de grandes bases de dados biológicos e químicos, associada ao aumento do poder computacional, tem permitido o treinamento de modelos cada vez mais robustos e generalizáveis. Apesar desses avanços, ainda existem desafios importantes relacionados à interpretabilidade dos modelos, viés nos conjuntos de dados, custo computacional e capacidade de generalização para sistemas biológicos complexos. Nesse contexto, compreender os fundamentos dessas abordagens e suas aplicações práticas torna-se essencial para a formação de pesquisadores capazes de desenvolver e aplicar soluções computacionais modernas na descoberta racional de fármacos.

 ©2026 SBC – Soc. Bras. de Computação

6.1.1. Objetivo

Apresentar, de forma teórica e prática, a aplicação de Aprendizado de Máquina na descoberta de fármacos, com enfoque na preparação de equipes e divulgação no Brasil de competições com iniciativa ciência aberta (*Open Science*) para esse fim.

6.1.2. Organização do Capítulo

Este capítulo está organizado da seguinte forma: inicialmente, a Seção *Referencial Teórico* apresenta conceitos fundamentais relacionados à bioinformática estrutural, dados biológicos, Aprendizado de Máquina, e iniciativas colaborativas como CACHE e DREAM Challenge. Posteriormente, a Seção *Estratégias Computacionais para a descoberta de fármacos* descreve os métodos de *docking* molecular, com destaque para os diferentes tipos de FEs e os principais métodos de Co-folding em uso atualmente. Em seguida, a Seção *Estudo de Caso: WDR91* apresenta de forma resumida um estudo de caso para o alvo WDR91, demonstrando o uso integrado de ferramentas computacionais e métodos de IA em um problema real de descoberta de fármacos. Por fim, a seção *Considerações finais* discute os desafios atuais e as perspectivas futuras para o uso de IA e AM na área.

6.2. Referencial Teórico

Esta subseção apresenta o referencial teórico para fundamentar as abordagens computacionais discutidas ao longo do capítulo, situando a descoberta de fármacos no contexto da bioinformática estrutural, da disponibilidade de dados moleculares e da aplicação de métodos de AM. Inicialmente, são introduzidos os conceitos centrais da bioinformática estrutural, enfatizando o papel da estrutura tridimensional na compreensão da função biológica. Em seguida, são descritos os principais tipos de dados utilizados na área, incluindo bases de dados de receptores, moléculas e interações experimentais, bem como os formatos e representações computacionais empregados para sua manipulação. A seção também aborda, de maneira introdutória, técnicas de AM relevantes para a descoberta de fármacos, desde a seleção de atributos e modelos supervisionados clássicos até abordagens modernas baseadas em aprendizado profundo, além de estratégias adequadas de validação. Por fim, são apresentadas iniciativas de avaliação colaborativa, como as competições CACHE e DREAM, que posiciona o uso dessas metodologias em cenários realistas e desafiadores.

6.2.1. Bioinformática estrutural

A bioinformática estrutural é uma área interdisciplinar que integra conceitos e métodos da Biologia Estrutural, da Química Computacional, da Física e da Computação. Esta área de estudos dedica-se à análise e modelagem computacional de estruturas tridimensionais de macromoléculas biológicas, partindo do princípio de que a função biológica está intimamente ligada à estrutura tridimensional (3D) das moléculas [Branden and Tooze 2012, Petsko and Ringe 2004]. A forma espacial assumida por proteínas e ácidos nucleicos determina propriedades fundamentais como especificidade molecular, afinidade de ligação e atividade catalítica, uma vez que processos biológicos tem origem diretamente da disposição dos átomos e resíduos no espaço. Alterações estruturais, sejam elas induzidas por mutações, interações intermoleculares ou efeitos dinâmicos, podem resultar em mudanças significativas de função, evidenciando a inseparabilidade

 ©2026 SBC – Soc. Bras. de Computação

entre estrutura tridimensional e função biológica em nível molecular.

Embora grande parte das aplicações em descoberta de fármacos concentre-se em proteínas globulares, a bioinformática estrutural abrange um escopo mais amplo. A área dedica-se também ao estudo estrutural de ácidos nucleicos, como DNA e RNA, incluindo ribozimas, ribossomos e complexos ribonucleoproteicos, bem como de complexos macromoleculares, membranas biológicas e sistemas biomoleculares heterogêneos [McCammon and Harvey 1987]. Diferentes representações estruturais podem ser empregadas, variando desde modelos atômicos completos, nível de átomos, até descrições *coarse-grained*[1], dependendo da escala espacial e temporal do fenômeno de interesse. Do ponto de vista metodológico, a bioinformática estrutural engloba um conjunto diversificado de técnicas computacionais, entre as quais se destacam a modelagem por homologia, a comparação e alinhamento estrutural, o *docking* molecular (veja na subseção 6.3.1), a Dinâmica Molecular clássica e acelerada, simulações de Monte Carlo, análises de flexibilidade e estudos de estabilidade estrutural. Essas abordagens permitem investigar não apenas estruturas estáticas, mas também aspectos dinâmicos e funcionais dos sistemas biológicos, consolidando a bioinformática estrutural como uma área central para a compreensão mecanicista de processos moleculares e para o desenvolvimento racional de fármacos.

A estrutura tridimensional das biomoléculas fornece a base para a identificação e caracterização de sítios funcionais, como bolsões de ligação, cavidades catalíticas e regiões de interação proteína-ligante. A descrição espacial detalhada dessas estruturas também possibilita a determinação dos tipos de interações atômicas que estabilizam os complexos biomoleculares, incluindo ligações de hidrogênio, interações hidrofóbicas, forças eletrostáticas e empilhamentos aromáticos, fatores determinantes tanto da afinidade quanto da seletividade molecular [Petsko and Ringe 2004]. Do ponto de vista computacional, esses modelos estruturais 3D constituem a entrada essencial para métodos amplamente empregados na bioinformática estrutural. Em particular, o *docking* molecular explora a complementaridade geométrica e energética entre ligantes e alvos biológicos, enquanto a Dinâmica Molecular investiga a relação entre conformação, flexibilidade e função ao longo do tempo, permitindo capturar efeitos que não são acessíveis por abordagens puramente estáticas. Além disso, a estrutura tridimensional é central em análises como comparação estrutural, modelagem por homologia, avaliação de mutações e estudos de estabilidade, consolidando-se como um elemento fundamental nos *pipelines* modernos de bioinformática estrutural e descoberta de fármacos.

Como exemplo representativo de organização estrutural, são apresentadas em maior detalhe as proteínas, que figuram entre as macromoléculas mais extensivamente estudadas na bioinformática estrutural, tanto pela diversidade de suas funções biológicas quanto pela ampla disponibilidade de dados estruturais tridimensionais. As proteínas podem ser descritas em diferentes níveis hierárquicos, tradicionalmente classificados como estrutura primária, secundária, terciária e quaternária [Branden and Tooze 2012]. Essa hierarquia estrutural está representada da esquerda para direita na Figura 6.1. A estrutura primária corresponde à sequência linear de aminoácidos, cuja composição e ordem

[1] resolução reduzida para conjuntos de átomos, possibilitando a captura de fenômenos em escalas maiores

 ©2026 SBC – Soc. Bras. de Computação

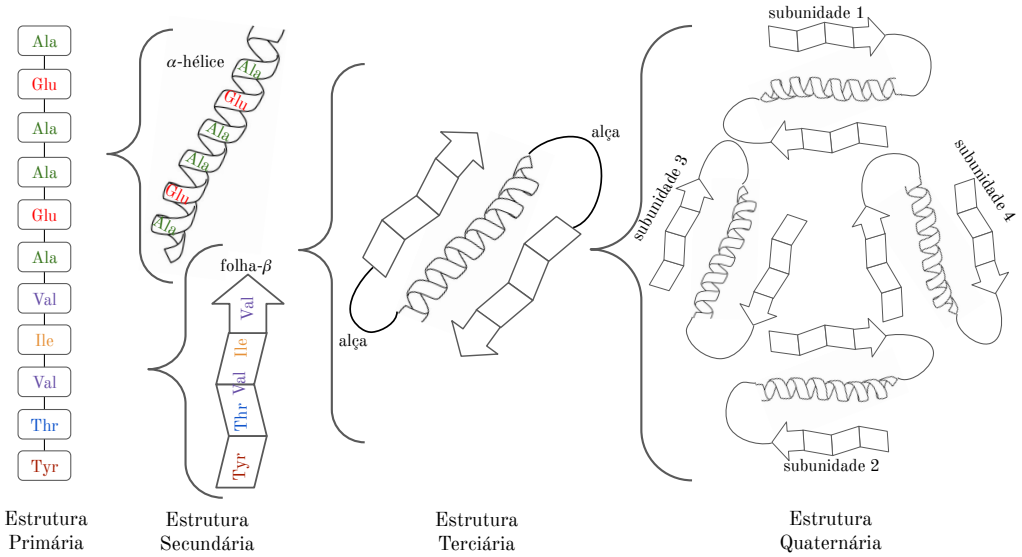


Figura 6.1: Organização estrutural das proteínas. Da esquerda para direita são apresentadas representações das estruturas primária, secundária, tericiária e quaternária. A estrutura primária consiste na sequência específica de aminoácidos. A cadeia peptídica resultante pode se dobrar em uma hélice-α (alfa-hélice) ou em uma folha-β que são tipos de estrutura secundária. Estes segmentos são incorporados à estrutura terciária da cadeia polipeptídica dobrada e conectados por alças. A estrutura quaternária é composta por subunidades de estruturas terciárias fomando complexos proteícos.

determinam as possibilidades conformacionais da molécula. A partir dessa sequência, interações locais entre resíduos dão origem às estruturas secundárias, como hélices-α e folhas-β, estabilizadas principalmente por ligações de hidrogênio. Por sua vez, a estrutura terciária emerge do arranjo tridimensional completo da cadeia polipeptídica, resultante da interação entre elementos secundários e de forças físico-químicas de maior alcance, como interações hidrofóbicas, eletrostáticas e ligações dissulfeto. Em muitas proteínas funcionalmente relevantes, a atividade biológica emerge não apenas da conformação de uma única cadeia, mas da associação específica entre múltiplas subunidades, caracterizando a estrutura quaternária. Esse nível estrutural descreve a organização espacial de complexos proteicos formados por duas ou mais cadeias polipeptídicas, cujas interações são essenciais para processos como regulação alostérica, estabilidade estrutural e reconhecimento molecular.

6.2.2. Dados

Esta subseção discutirá os diferentes tipos de dados utilizados em estudos de descoberta de fármacos assistida por computador. Serão abordados dados estruturais de proteínas (experimentais e modelados), dados químicos de pequenas moléculas, dados experimentais de interação proteína-ligante e dados derivados de triagens químicas de larga escala.

6.2.2.1. Bancos de dados de receptores

Fundado em 1971 como o primeiro arquivo digital de biologia de acesso aberto, o Protein Data Bank (PDB) é o repositório global primário de dados estruturais experi-

 ©2026 SBC – Soc. Bras. de Computação

mentais de macromoléculas biológicas. Atualmente mantido pelo consórcio internacional **Worldwide Protein Data Bank** (wwPDB) (https://www.wwpdb.org/), o repositório opera através de um sistema colaborativo onde pesquisadores de todo o mundo depositam suas descobertas empíricas, derivadas majoritariamente de cristalografia de raios X, Ressonância Magnética Nuclear (RMN) e criomicroscopia eletrônica. Em termos de funcionamento, cada submissão passa por um rigoroso processo de validação, anotação e padronização biocuratorial antes de ser disponibilizada publicamente, garantindo a integridade e reprodutibilidade científica.

O banco disponibiliza uma vasta gama de dados que vão além das coordenadas atômicas tridimensionais (eixos x, y, z). Os arquivos (disponibilizados primariamente no formato moderno *mmCIF* e no formato legado *PDB*) incluem estruturas primárias, secundárias, terciárias e complexos quaternários de proteínas, ácidos nucleicos (DNA/RNA) e complexos proteína-ligante fundamentais para o desenho de fármacos. Além da geometria da molécula, os dados contêm metadados experimentais detalhados, como fatores de temperatura (*B-factor*), métricas de resolução, informações sobre a taxonomia do organismo de origem e condições do ensaio bioquímico.

Mais recentemente, acompanhando a evolução da IA, o PDB expandiu sua infraestrutura para abrigar modelos preditos computacionalmente através do portal *Computed Structure Models* (CSM). Dentre os principais provedores de CSMs destaca-se o *AlphaFold Protein Structure Database* (AFDB) (https://alphafold.ebi.ac.uk/), fruto da parceria entre a Google DeepMind e o *European Bioinformatics Institute* (EMBL-EBI). O AFDB revolucionou a disponibilidade de dados estruturais ao fornecer modelos preditos com alta confiança para mais de 200 milhões de sequências catalogadas no UniProt usando o algoritmo AlphaFold2, preenchendo lacunas cruciais no RDD para alvos que ainda não possuem resolução empírica.

6.2.2.2. Bancos de dados de moléculas químicas

A exploração do espaço químico, a VS e o treinamento de modelos de AM dependem da disponibilidade de bancos de dados de moléculas. Essas plataformas catalogam desde compostos prontos para síntese até fármacos aprovados e suas respectivas atividades biológicas, fornecendo os dados de entrada (*inputs*) essenciais para campanhas de VS e modelagem preditiva. Abaixo, destacam-se os principais repositórios utilizados na área:

ZINC (https://zinc.docking.org/): Desenvolvido e mantido pela Universidade da Califórnia, São Francisco (UCSF), o ZINC é um banco de dados gratuito focado em coleções de compostos disponíveis comercialmente, otimizados especificamente para VS. Suas versões mais recentes (como ZINC15 e ZINC20) catalogam bilhões de moléculas, oferecendo representações em formatos tridimensionais (como MOL2 e SDF) já preparados para *docking*, além de anotações sobre propriedades físico-químicas e viabilidade biológica.

PubChem (https://pubchem.ncbi.nlm.nih.gov/): Mantido pelo National Center for Biotechnology Information (NCBI), o PubChem atua como um repositório aberto massivo que integra estruturas químicas a dados de ensaios biológicos. A arquitetura de dados da plataforma é estruturada em três bancos primários interconectados, operando

 ©2026 SBC – Soc. Bras. de Computação

com diferentes níveis de curadoria e aplicação: *Substances*, *Compounds* e *BioAssays*.

O ***PubChem Substances*** armazena registros químicos brutos exatamente como submetidos pelos depositantes originais, abrigando misturas complexas, extratos não caracterizados e registros redundantes sob identificadores SID (*Substance ID*). O ***PubChem Compounds*** extrai e padroniza os dados da base *Substances* para gerar representações de moléculas químicas únicas e estruturalmente validadas, designando a cada uma um identificador CID (*Compound ID*). Esta etapa elimina ambiguidades e centraliza o espaço químico rastreável. O ***PubChem BioAssays*** consolida os resultados de testes biológicos, vinculando CIDs e SIDs a painéis de toxicidade, ensaios enzimáticos e projetos de triagem de alto rendimento (*High-Throughput Screening* - HTS), registrando tanto métricas quantitativas empíricas (IC_{50}, EC_{50}, K_i) quanto classificações categóricas (ativo/inativo/inconclusivo).

A integração sistemática do módulo *BioAssays* com o *Compounds* serve de base para o treinamento de modelos QSAR (*Quantitative Structure-Activity Relationship*). No fluxo de trabalho de AM, os dados do *BioAssays* fornecem os rótulos de classe ou as variáveis contínuas alvo (*ground truth* experimental) essenciais para a aprendizagem supervisionada.

Catálogos de Fornecedores (Enamine, MolPort e outros): Além dos repositórios acadêmicos, os catálogos de empresas químicas fornecedoras desempenham um papel crucial para garantir que as moléculas preditas possam ser fisicamente testadas. A **Enamine** destaca-se por fornecer bibliotecas massivas baseadas em síntese química sob demanda (*make-on-demand*), como o *REAL Space*, que expande o espaço químico explorável para dezenas de bilhões de compostos. O **MolPort**, por sua vez, atua como um agregador global, consolidando catálogos de múltiplos fornecedores em uma única interface, o que facilita a aquisição logística de moléculas promissoras (*hits*) para ensaios *in vitro*.

ChEMBL (https://www.ebi.ac.uk/chembl/): Gerenciado pelo European Bioinformatics Institute (EMBL-EBI), o ChEMBL é um banco de dados de quimioinformática focado em compostos bioativos com propriedades do tipo fármaco (*drug-like*). Seu grande diferencial é a curadoria de dados extraídos manualmente da literatura científica primária, fornecendo medidas quantitativas rigorosas de afinidade de ligação, como constantes de dissociação (K_d), constantes de inibição (K_i) e concentrações inibitórias meias-máximas (IC_{50}). Essa riqueza de dados experimentais torna o ChEMBL uma das bases mais importantes para o treinamento de FEs empíricas e modelos de regressão em AM. Dados do ChEMBL e PubChem são geralmente complementares e interconectados.

DrugBank (https://go.drugbank.com/): Consiste em um banco de dados abrangente que combina quimioinformática e farmacologia, contendo informações detalhadas sobre fármacos aprovados, compostos em fase de investigação clínica e farmacêuticos. O DrugBank integra a estrutura química de pequenas moléculas com anotações biológicas profundas, incluindo mecanismos de ação, vias de metabolismo, interações medicamentosas e o mapeamento preciso de seus respectivos alvos proteicos.

Therapeutic Target Database (TTD) (https://ttd.idrblab.cn/): Diferentemente das bases puramente químicas, o TTD foca na relação bidirecional entre alvos terapêuti-

 ©2026 SBC – Soc. Bras. de Computação

cos (proteínas, ácidos nucleicos) e as moléculas direcionadas a eles. O banco fornece informações detalhadas sobre alvos associados a doenças específicas, o *status* de aprovação ou fase de ensaio clínico dos compostos correspondentes e as vias bioquímicas envolvidas. O TTD é uma ferramenta particularmente valiosa para estudos de reposicionamento de fármacos (*drug repurposing*) e avaliação de perfis de eficácia estrutural.

6.2.2.3. Bases de Dados experimentais de interação

Para o desenvolvimento de modelos preditivos baseados em estrutura, é imperativo o mapeamento direto entre a conformação tridimensional de um complexo biomolecular e sua respectiva força de interação experimental [Wang et al. 2004]. Nesse contexto, bases de dados especializadas desempenham papel central ao integrar informações estruturais e biofísicas utilizadas no treinamento e validação de modelos de IA aplicados à descoberta de fármacos.

O **PDBbind** (https://www.pdbbind-plus.org.cn/) constitui o principal repositório global curado para integração entre estruturas tridimensionais e dados quantitativos de afinidade molecular [Wang et al. 2004]. A base extrai sistematicamente complexos biomoleculares depositados no PDB, incluindo sistemas proteína-ligante, proteína-proteína e proteína-ácido nucleico, associando essas estruturas a medições experimentais reportadas na literatura primária. Entre os parâmetros documentados destacam-se K_d, K_i e IC_{50}.

A arquitetura do PDBbind é organizada em diferentes níveis de curadoria e rigor experimental. O *general set* contém o conjunto mais amplo de complexos disponíveis; o *refined set* aplica filtros rigorosos relacionados à resolução cristalográfica, consistência estrutural e qualidade molecular; enquanto o *core set*, também denominado CASF *benchmark* (*Comparative Assessment of Scoring Functions*), é amplamente utilizado para *benchmarking* e validação comparativa de FEs e modelos preditivos [Su et al. 2018].

No ecossistema de AM aplicado à descoberta de fármacos, o PDBbind atua como uma das principais fontes de dados rotulados (*ground truth*) [Wang et al. 2004]. A geometria atômica tridimensional do complexo fornece a entrada para extração de descritores espaciais, modelagem baseada em grafos e representação volumétrica tridimensional, enquanto os valores experimentais de afinidade constituem o sinal supervisionado utilizado no treinamento de redes neurais convolucionais 3D, modelos de regressão e FEs híbridas baseadas em IA [Jiménez et al. 2018].

O **BindingDB** (https://www.bindingdb.org) é um banco de dados público especializado em afinidades de ligação experimentalmente determinadas entre proteínas alvo e pequenas moléculas bioativas [Liu et al. 2007]. Diferentemente do PDBbind, a inclusão de entradas no BindingDB não depende da disponibilidade de estruturas cristalográficas tridimensionais resolvidas.

A plataforma agrega dados oriundos de ensaios bioquímicos, testes enzimáticos e medições biofísicas, incluindo valores de K_i, K_d, IC_{50} e constantes cinéticas, extraídos sistematicamente da literatura científica e de documentos de patentes [Gilson et al. 2015]. Essa característica amplia significativamente o volume de dados disponíveis para treinamento de modelos QSAR bidimensionais, redes neurais baseadas em sequências proteicas

 ©2026 SBC – Soc. Bras. de Computação

e arquiteturas fundamentadas exclusivamente em grafos moleculares.

Além disso, devido ao seu elevado volume de interações anotadas experimentalmente, o BindingDB tornou-se uma importante fonte de dados para aplicações de *deep learning*, aprendizado multitarefa e modelos de predição de afinidade proteína-ligante em larga escala [Gilson et al. 2015].

AIRCHECK (Artificial Intelligence-Ready CHEmiCal Knowledge-base) https://aircheck.ai/, é uma iniciativa aberta desenvolvida para disponibilizar dados químicos estruturados e padronizados voltados ao desenvolvimento e avaliação de modelos de AM aplicados à descoberta de fármacos. A plataforma integra dados experimentais provenientes de Bibliotecas Codificadas por DNA (*DNA-Encoded Libraries* - DEL), Espectrometria de Massas com Seleção por Afinidade (*Affinity Selection Mass Spectrometry* - ASMS) e validação de compostos, incluindo exemplos positivos e negativos de interação molecular, além de fornecer recursos para treinamento, *benchmark* e avaliação de modelos preditivos em cenários mais realistas.

6.2.2.4. Métodos Experimentais para Ensaios de *High-Throughput Screening* (HTS)

6.2.2.4.1. Bibliotecas Codificadas por DNA (DEL)

A tecnologia de DEL revolucionou a escala da triagem física de compostos químicos [Goodnow et al. 2016, Gironda-Martínez et al. 2021]. Diferentemente dos paradigmas clássicos, nos quais moléculas são avaliadas individualmente, as DELs permitem a síntese combinatória em esquema *split-and-pool*, combinando sequencialmente diferentes *building blocks* químicos em múltiplos ciclos reacionais (Figura 6.2).

Teoricamente, considerando a utilização de 10^3 *building blocks* distintos em dois ciclos sucessivos de combinação química, o espaço químico gerado alcançaria aproximadamente 10^6 moléculas únicas. Em bibliotecas mais complexas, contendo múltiplas etapas sintéticas adicionais, esse número pode atingir escalas superiores a bilhões de compostos [Goodnow et al. 2016, Franzini et al. 2014].

Cada molécula sintetizada é fisicamente conjugada a uma sequência curta e única de DNA, que atua como um código de barras molecular identificador, permitindo a rastreabilidade do histórico sintético e a identificação posterior dos compostos enriquecidos durante os ensaios de afinidade.

Durante o ensaio de afinidade, a biblioteca completa é incubada com a proteína alvo imobilizada. Compostos sem afinidade significativa são removidos por lavagem, enquanto os ligantes retidos permanecem associados ao alvo biológico. Após dissociação do complexo, os códigos de DNA correspondentes aos compostos enriquecidos são amplificados via PCR e identificados por Sequenciamento de Nova Geração (*Next-Generation Sequencing* – NGS) [Franzini et al. 2014].

No contexto do AM, ensaios DEL produzem matrizes de dados massivas contendo contagens de enriquecimento associadas à ligação molecular. O enorme volume de dados permite o treinamento de modelos robustos de classificação, VS e aprendi-

 ©2026 SBC – Soc. Bras. de Computação

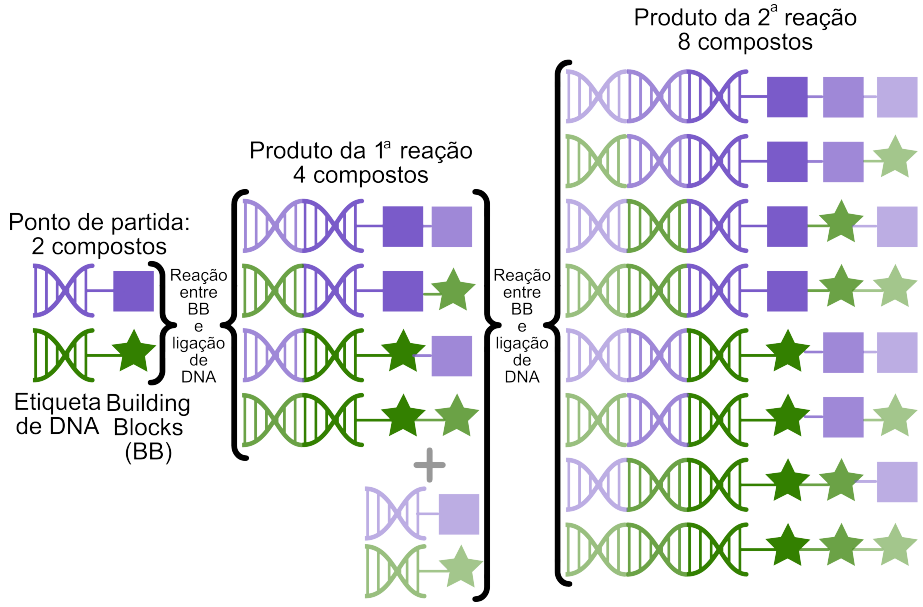


Figura 6.2: Representação esquemática da estratégia de síntese combinatória *split-and-pool* empregada em Bibliotecas Codificadas por DNA (DELs). Inicialmente, diferentes *building blocks* químicos são conjugados a sequências específicas de DNA que atuam como códigos de barras moleculares. Após cada ciclo de reação química, os compostos são reunidos (*pool*), redistribuídos (*split*) e submetidos a novas etapas de síntese com conjuntos adicionais de fragmentos químicos. O processo iterativo permite a geração exponencial de diversidade molecular, possibilitando a construção de bibliotecas contendo milhões a bilhões de compostos distintos em um único experimento.

zado profundo, inclusive em cenários sem estruturas tridimensionais explícitas disponíveis [Goodnow et al. 2016]. Resultados recentes apontam a descoberta de ligantes na ordem de interação de µM, requerendo ciclos de otimização [Ahmad et al. 2023].

Apesar de seu alto poder de escala, esses dados apresentam características que exigem cuidado na análise computacional. O sinal experimental pode ser influenciado por fatores como eficiência de síntese, qualidade da biblioteca, interações inespecíficas, efeitos do DNA conjugado, abundância inicial dos compostos e ruído de sequenciamento. Além disso, a definição de compostos positivos e negativos depende de critérios de enriquecimento e de processamento dos dados. Portanto, embora DEL seja uma fonte extremamente valiosa de dados para AM, esses dados devem ser interpretados como evidências experimentais de priorização, e não como medidas diretas e definitivas de afinidade.

6.2.2.4.2. Espectrometria de Massas com Seleção por Afinidade (ASMS)

Diferentemente de ensaios dependentes de fluoróforos, marcadores radioativos ou códigos de DNA, a ASMS constitui uma abordagem *label-free*, eliminando potenciais interferências físico-químicas decorrentes da marcação molecular [Muchiri and van Breemen 2020]. O método consiste na incubação da proteína alvo com misturas contendo centenas ou milhares de pequenas moléculas simultaneamente.

Posteriormente, técnicas como cromatografia de exclusão por tamanho (*Size Exclusion Chromatography* – SEC) ou ultrafiltração molecular são empregadas para separar

 ©2026 SBC – Soc. Bras. de Computação

rapidamente os complexos proteína-ligante das moléculas livres em solução. Em seguida, os ligantes associados são liberados sob condições desnaturantes e identificados por espectrometria de massas de alta resolução [Prudent et al. 2023].

Uma variação recente dessa abordagem é a **eASMS** (*enantioselective* ASMS). Frequentemente, pequenas moléculas quirais existem como pares de enantiômeros, isto é, formas moleculares que possuem a mesma composição química e conectividade, mas que se organizam no espaço como imagens especulares não sobreponíveis. Como proteínas também são estruturas tridimensionais quirais (formada por L-aminoácidos - levógiros), seus sítios de ligação podem reconhecer de forma preferencial um enantiômero em relação ao outro. Assim, a observação de ligação enantiosseletiva fornece uma evidência adicional de que a interação detectada é específica e dependente de complementaridade molecular tridimensional.

Bibliotecas ricas em compostos quirais são particularmente úteis porque pares de enantiômeros funcionam como controles internos muito próximos, que compartilham massa, fórmula molecular e muitas propriedades físico-químicas, mas diferem na orientação espacial de seus grupos químicos. Portanto, diferenças consistentes de enriquecimento entre enantiômeros podem ajudar a distinguir interações específicas de artefatos inespecíficos de retenção, agregação ou ruído experimental, elevando a taxas de verdadeiros positivos.

Essa característica torna a eASMS especialmente atraente para a descoberta de ligantes fracos ou iniciais, que são frequentes em campanhas voltadas a alvos pouco explorados. Em um estudo recente, essa abordagem foi aplicada a 31 proteínas humanas utilizando uma biblioteca de mais de 8 mil compostos quirais, permitindo identificar 16 ligantes fracos para 12 alvos desafiadores, gerar evidências de seletividade e oferecer confirmação ortogonal de ligação [Wang et al. 2025]. Os autores destacam que a eASMS é útil para identificar e caracterizar ligantes seletivos para proteínas previamente sem ligantes conhecidos ou consideradas desafiadoras para descoberta química.

Entretanto, assim como em DEL, os dados de eASMS exigem interpretação cuidadosa. Um sinal de enriquecimento não deve ser confundido automaticamente com uma medida quantitativa de afinidade, como um valor de K_d. A intensidade observada pode depender de condições experimentais, composição da biblioteca, concentração da proteína, competição entre compostos, eficiência de separação e sensibilidade da espectrometria de massas. Portanto, *hits* identificados por eASMS geralmente devem ser confirmados por métodos ortogonais, como Ressonância Plasmônica de Superfície (*Surface Plasmon Resonance* - SPR), ITC, Ensaios de Mudança Térmica (*Thermal Shift Assay* - TSA), ensaios funcionais ou estudos estruturais.

Para aplicações em IA, os dados oriundos de ASMS são considerados de alta qualidade experimental devido ao reduzido nível de ruído e à evidência física direta de interação não covalente. Frequentemente, esses conjuntos são utilizados como referência experimental (*gold standard*) para validação de modelos treinados com bases mais volumosas e ruidosas, como DELs [Muchiri and van Breemen 2020]. Dados de DEL e ASMS já vêm sendo disponibilizados para diferentes alvos na plataforma AIRCHECK.

Além de dados estruturais oriundos de complexos previamente cristalizados, como

 ©2026 SBC – Soc. Bras. de Computação

aqueles disponíveis em bases especializadas a exemplo do PDBBind [Wang et al. 2004], campanhas modernas de HTS produzem volumes massivos de dados experimentais de interação molecular. Esses ensaios desempenham papel central no treinamento de modelos preditivos, pois fornecem o "gabarito" experimental necessário para abordagens supervisionadas, incluindo classificação binária (*ativo* versus *inativo*) e estimativas quantitativas de afinidade. Entre as tecnologias experimentais mais relevantes neste contexto destacam-se metodologias de larga escala e alta precisão, capazes de alimentar arquiteturas modernas de IA aplicada à descoberta de fármacos.

6.2.2.5. Métodos Ortogonais de Validação de Interação

6.2.2.5.1. Ressonância Plasmônica de Superfície

A técnica de SPR constitui um método óptico avançado para monitoramento em tempo real de interações biomoleculares sem necessidade de marcadores fluorescentes [Homola 2008]. Nesse sistema, a proteína alvo é imobilizada sobre um biossensor metálico, geralmente um *chip* de ouro, enquanto soluções contendo os ligantes fluem continuamente por um sistema microfluídico. A interação molecular altera localmente o índice de refração da superfície, produzindo um sinal óptico mensurável instantaneamente.

Diferentemente de ensaios de ponto final, a SPR permite determinar simultaneamente parâmetros termodinâmicos e cinéticos da interação molecular, incluindo a constante de dissociação (K_d), as taxas de associação (k_{on}) e dissociação (k_{off}) [Rich and Myszka 2008]. Em descoberta de fármacos assistida por IA, o chamado "tempo de residência" molecular ($1/k_{off}$) emergiu como um importante parâmetro preditivo de eficácia *in vivo*, motivando o desenvolvimento de modelos focados em dinâmica cinético-estrutural [Copeland et al. 2006].

6.2.3. Formato dos dados

A aplicação de métodos de AM na descoberta de fármacos depende fortemente da forma como as estruturas moleculares e os dados químicos são representados e armazenados. Ao longo das últimas décadas, diferentes formatos de arquivos foram desenvolvidos para atender a necessidades específicas das áreas de bioinformática e quimioinformática. Dentre essas necessidades, destacam-se a descrição tridimensional de macromoléculas, o armazenamento de pequenas moléculas com propriedades associadas e a codificação compacta de estruturas químicas, permitindo o armazenamento eficiente, a interoperabilidade entre ferramentas computacionais e o processamento de grandes volumes de dados em aplicações de modelagem preditiva e VS. Nesta subseção, são apresentados alguns dos principais formatos utilizados em bioinformática estrutural e química computacional; PDB, mmCIF, MOL2, SDF e SMILES.

O formato **PDB** (Protein Data Bank) é o padrão histórico para a representação de estruturas tridimensionais de macromoléculas biológicas, incluindo proteínas, ácidos nucleicos e complexos proteína-ligante, veja um exemplo na FIGURA 6.3. Criado na década de 1970 juntamente com o repositório homônimo, o formato PDB armazena co-

 ©2026 SBC – Soc. Bras. de Computação

ordenadas atômicas em três dimensões, informações sobre conectividade, elementos químicos, fatores de ocupação e temperatura, além de metadados experimentais, como o método de determinação estrutural (por exemplo, cristalografia por raios X, RMN ou criomicroscopia eletrônica) [Berman et al. 2000, Burley et al. 2021]. Apesar de sua ampla adoção, o formato PDB apresenta limitações conhecidas, como restrições rígidas de largura de campos, ausência de suporte nativo a propriedades químicas detalhadas e descrições incompletas de estados de protonação ou ordens de ligação. Essas limitações motivaram o desenvolvimento do formato mmCIF, mais estruturado e extensível. Ainda assim, arquivos PDB continuam sendo largamente utilizados em *pipelines* de *docking* molecular, modelagem molecular e AM baseado em estruturas tridimensionais de proteínas, principalmente devido à grande quantidade de estruturas disponíveis publicamente [Westbrook and Fitzgerald 2003].

```
ATOM      1  N   VAL A   2     -78.391 -19.186  13.764  1.00 76.19           N
ATOM      2  CA  VAL A   2     -78.926 -20.535  13.907  1.00 80.33           C
ATOM      3  C   VAL A   2     -79.769 -20.649  15.173  1.00 83.98           C
ATOM      4  O   VAL A   2     -79.303 -20.341  16.269  1.00 92.42           O
ATOM      5  CB  VAL A   2     -77.800 -21.580  13.919  1.00 79.68           C
ATOM      6  CG1 VAL A   2     -78.368 -22.972  14.144  1.00 73.58           C
ATOM      7  CG2 VAL A   2     -77.010 -21.523  12.620  1.00 97.52           C
```

Figura 6.3: Trecho ilustrativo de um arquivo no formato PDB referente à helicase NSP13 do SARS-CoV-2 (código PDB: 7NN0), no qual são apresentadas linhas do tipo ATOM contendo a identificação dos átomos, resíduos, cadeias e suas coordenadas tridimensionais.

O formato **mmCIF** (macromolecular Crystallographic Information File) é atualmente o padrão oficial adotado pelo Protein Data Bank para o armazenamento e distribuição de estruturas tridimensionais de macromoléculas biológicas. Diferentemente do formato PDB clássico,o mmCIF utiliza uma estrutura baseada em pares chave-valor organizados em tabelas, permitindo maior flexibilidade, extensibilidade e consistência na descrição dos dados estruturais, veja um exemplo em 6.4. O mmCIF foi projetado para acomodar estruturas macromoleculares de grande porte e elevada complexidade, como complexos multiproteicos, ribossomos, vírus e estruturas determinadas por criomicroscopia eletrônica em alta resolução, que frequentemente excedem os limites do formato PDB. Além das coordenadas tridimensionais dos átomos, arquivos mmCIF permitem a representação mais completa de metadados experimentais, informações cristalográficas, relações de simetria e anotações estruturais, facilitando a interoperabilidade entre bases de dados e ferramentas computacionais.

O formato **MOL2** foi desenvolvido originalmente pela Tripos Associates e é amplamente utilizado para representar pequenas moléculas em contextos de química computacional e modelagem molecular, veja na Figura 6.5. Diferentemente do PDB, o MOL2 inclui informações químicas mais detalhadas, como tipos atômicos específicos do campo de força, ordens de ligação, cargas parciais e identificação explícita de subestruturas, [Tripos 2005]. Essas características tornam o MOL2 particularmente adequado para aplicações como *docking* molecular, parametrização de campos de força e geração de descritores moleculares. No contexto de AM, arquivos MOL2 são frequentemente usados como ponto de partida para extração de grafos moleculares enriquecidos, nos quais átomos e ligações podem ser representados por atributos mais informativos do que aqueles

 ©2026 SBC – Soc. Bras. de Computação

```
loop_
_atom_site.group_PDB
_atom_site.id
_atom_site.type_symbol
_atom_site.label_atom_id
_atom_site.label_alt_id
_atom_site.label_comp_id
_atom_site.label_asym_id
_atom_site.label_entity_id
_atom_site.label_seq_id
_atom_site.Cartn_x
_atom_site.Cartn_y
_atom_site.Cartn_z
_atom_site.occupancy
_atom_site.B_iso_or_equiv
_atom_site.pdbx_PDB_model_num

ATOM 1  N   N   . VAL A 1 2 -78.391 -19.186 13.764 1.00 76.19 1
ATOM 2  C   CA  . VAL A 1 2 -78.926 -20.535 13.907 1.00 80.33 1
ATOM 3  C   C   . VAL A 1 2 -79.769 -20.649 15.173 1.00 83.98 1
ATOM 4  O   O   . VAL A 1 2 -79.303 -20.341 16.269 1.00 92.42 1
ATOM 5  C   CB  . VAL A 1 2 -77.800 -21.580 13.919 1.00 79.68 1
ATOM 6  C   CG1 . VAL A 1 2 -78.368 -22.972 14.144 1.00 73.58 1
ATOM 7  C   CG2 . VAL A 1 2 -77.010 -21.523 12.620 1.00 97.52 1
```

Figura 6.4: Trecho ilustrativo de um arquivo no formato mmCIF (.cif) referente à helicase NSP13 do SARS-CoV-2 (PDB: 7NN0). O bloco `loop_` define a tabela `_atom_site`, que armazena informações equivalentes às linhas `ATOM` do formato PDB, incluindo identificação atômica, resíduo, cadeia e coordenadas tridimensionais.

disponíveis em formatos mais simples [Morris et al. 2009]. Entretanto, o MOL2 não é um formato padronizado por uma entidade internacional, e variações na forma como tipos atômicos e cargas são atribuídos podem impactar a reprodutibilidade entre diferentes ferramentas.

```
@<TRIPOS>ATOM
1 N1   -0.012  1.245  0.000 N.ar  1 CAF  -0.347
2 C2    1.234  0.678  0.000 C.2   1 CAF   0.215
3 N3    1.198 -0.745  0.000 N.ar  1 CAF  -0.289
4 C4   -0.034 -1.312  0.000 C.2   1 CAF   0.181
5 O5   -1.245 -0.845  0.000 O.2   1 CAF  -0.512
@<TRIPOS>BOND
1 1 2 ar
2 2 3 ar
3 3 4 2
4 4 5 2
```

Figura 6.5: Trecho ilustrativo de um arquivo no formato MOL2 representando a molécula de cafeína (PubChem CID: 2519), no qual são explicitadas a conectividade química, os tipos atômicos e as cargas parciais, informações comumente utilizadas em estudos de *docking* molecular e modelagem computacional.

O **SDF** (Structure Data File) é uma extensão do formato MOL desenvolvida pela Molecular Design Limited (MDL) e amplamente adotada para o armazenamento de conjuntos de pequenas moléculas, especialmente em bases de dados químicas e bibliotecas de compostos. Uma de suas principais vantagens é a capacidade de associar, a cada molécula, um conjunto arbitrário de propriedades, como atividades biológicas, valores de afinidade, descritores fisicoquímicos ou informações experimentais [Dalby et al. 1992]. Cada entrada em um arquivo SDF pode conter coordenadas 2D ou 3D, conectividade química e

 ©2026 SBC – Soc. Bras. de Computação

um bloco de dados estruturados em pares chave–valor. Essa flexibilidade torna o SDF um formato central em tarefas de aprendizado supervisionado, como predição de atividade, toxicidade ou propriedades ADMET, sendo amplamente utilizado em ferramentas como RDKit, Open Babel e *pipelines* de machine learning em química [Landrum 2006]. Como limitação, o SDF tende a ser relativamente verboso, o que pode impactar desempenho e armazenamento em grandes coleções de compostos.

```
2519
  -OEChem-04192618422D

 24 25  0     0  0  0  0  0  0999 V2000
    3.7321    2.0000    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
    2.0000   -1.0000    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
6  8  8
7  8  8
7  9  8

$$$$
```

Figura 6.6: Trecho ilustrativo de um arquivo no formato SDF (Structure Data File) representando a molécula de cafeína (PubChem CID: 2519), no qual a estrutura química é acompanhada de propriedades associadas, permitindo o armazenamento conjunto de dados estruturais e informações químicas ou biológicas.

O **SMILES** (Simplified Molecular Input Line Entry System) é uma representação textual linear que codifica a estrutura de uma molécula como uma sequência de caracteres ASCII. Desenvolvido com o objetivo de ser compacto e facilmente legível por máquinas, o SMILES descreve conectividade atômica, ramificações, ordens de ligação e estereoquímica por meio de uma gramática formal bem definida [Weininger 1988]. Devido à sua simplicidade e eficiência, o SMILES tornou-se extremamente popular em aplicações de AM, particularmente em modelos baseados em redes neurais recorrentes, transformers e modelos generativos, onde moléculas podem ser tratadas como sequências ou tokens [Gómez-Bombarelli et al. 2018]. No entanto, uma mesma molécula pode ser representada por múltiplos SMILES equivalentes, o que levou ao desenvolvimento de conceitos como SMILES canônico e randomized SMILES para aumentar a robustez de modelos estatísticos e redes neurais [Bjerrum 2017]. Apesar de não conter explicitamente coordenadas tridimensionais, o SMILES é frequentemente combinado com etapas posteriores de geração de conformações 3D. Como exemplo, a molécula de cafeína é representada em SMILES como: `CN1C=NC2=C1C(=O)N(C(=O)N2C)C`.

6.2.4. *Fingerprints* de moléculas

As *fingerprints* circulares, exemplificadas pelo algoritmo Morgan (*Extended-Connectivity Fingerprints*, ECFP) implementado no RDKit, operam por meio de um processo iterativo que mapeia a vizinhança atômica de maneira concêntrica. O cálculo inicia-se pela atribuição de identificadores numéricos a cada átomo individual com base em propriedades fundamentais, como número atômico, valência, carga formal e hibridização. A cada iteração, ou "raio", o identificador de um átomo é atualizado para incorporar as informações de seus vizinhos imediatos, gerando representações capazes de codificar subestruturas de complexidade crescente [Rogers and Hahn 2010, Landrum 2013]. O principal diferencial dessa abordagem reside em sua elevada capacidade de representar o ambiente químico local

 ©2026 SBC – Soc. Bras. de Computação

com alta resolução estrutural, sendo amplamente considerada o padrão-ouro para capturar centros reativos e padrões específicos de substituição química [Rogers and Hahn 2010].

Em contraste, as *fingerprints* baseadas em caminhos (*path-based fingerprints*) utilizam uma lógica fundamentada na conectividade linear da molécula. Nesse método, o algoritmo identifica exaustivamente todas as sequências possíveis de átomos e ligações até um comprimento máximo predefinido, geralmente limitado a sete ligações consecutivas [Daylight Chemical Information Systems 2024]. Cada caminho identificado é posteriormente processado por uma função de *hash*, responsável por ativar posições específicas em um vetor binário. Enquanto as representações circulares modelam "conchas" atômicas ao redor de um centro, as *fingerprints* baseadas em caminhos priorizam a topologia do esqueleto molecular e a continuidade das ligações químicas. Consequentemente, essas representações tendem a apresentar desempenho superior na discriminação de famílias de *scaffolds* lineares, sistemas ramificados e arquiteturas moleculares que não possuem um centro radial claramente definido [Cereto-Massagué et al. 2015].

As chaves estruturais de dicionário, exemplificadas pelas *MACCS Keys*, abandonam o paradigma de *hashing* em favor de uma comparação determinística contra um conjunto fixo de fragmentos químicos previamente definidos [Durant et al. 2002]. O cálculo consiste na verificação binária da presença ou ausência de cada um dos 166 padrões estruturais descritos no dicionário químico da ferramenta. O principal diferencial dessa abordagem é sua elevada interpretabilidade, uma vez que cada posição do vetor possui um significado químico explícito e invariável. Entretanto, sua principal limitação em relação às representações circulares ou baseadas em caminhos reside na incapacidade de capturar novidades estruturais, padrões emergentes ou fragmentos químicos que não tenham sido previamente incluídos no dicionário original [Cereto-Massagué et al. 2015].

A lógica das *fingerprints* de pares de átomos (*Atom Pair Fingerprints*) difere substancialmente das abordagens anteriores ao priorizar relações de distância topológica em vez de conectividade local imediata. Para cada par de átomos pesados presentes na molécula, o algoritmo registra simultaneamente o tipo químico dos dois átomos e o número mínimo de ligações que os separa [Carhart et al. 1985]. Dessa forma, a representação passa a capturar aspectos relacionados à forma global e às dimensões topológicas da molécula. Essa estratégia permite que modelos de AM identifiquem similaridades estruturais entre compostos que apresentam esqueletos químicos distintos, mas preservam distribuições espaciais equivalentes de grupos funcionais no espaço bidimensional molecular [Cereto-Massagué et al. 2015].

Por fim, as *fingerprints* de torsão topológica (*Topological Torsion Fingerprints*) concentram-se na descrição da rigidez molecular e de aspectos associados à estereoquímica teórica. O cálculo identifica sistematicamente todas as sequências compostas por quatro átomos consecutivamente ligados, codificando propriedades específicas de cada átomo da sequência, como hibridização, aromaticidade e número de elétrons π [Nilakantan et al. 1987]. Ao focar explicitamente em unidades de quatro átomos, essa abordagem torna-se capaz de distinguir isômeros e padrões conformacionais que poderiam ser considerados equivalentes por *fingerprints* circulares ou baseadas em caminhos. Como consequência, essa representação fornece aos modelos de IA informações relevantes sobre conformação molecular, liberdade rotacional e restrições geométricas internas

 ©2026 SBC – Soc. Bras. de Computação

da estrutura química [Cereto-Massagué et al. 2015].

6.2.5. Aprendizado de Máquina na descoberta de fármacos

Uma vez estabelecidas as representações moleculares e os conjuntos de dados utilizados na descoberta de fármacos, partimos para a aplicação sistemática de métodos de Aprendizado de Máquina para a construção de modelos preditivos. Nesta subseção, apresentamos uma visão integrada das principais etapas dessa metodologia, abrangendo desde técnicas de seleção de atributos e redução de dimensionalidade, passando por modelos supervisionados clássicos baseados em ensembles, até abordagens modernas de aprendizado profundo, bem como estratégias de validação e avaliação adequadas às particularidades da área de descoberta de fármacos.

6.2.5.1. Seleção de atributos e redução de dimensionalidade

A seleção de atributos constitui uma etapa fundamental em *pipelines* de AM, especialmente no contexto da descoberta de fármacos. As representações computacionais de moléculas, como *fingerprints*, descritores físico-químicos ou vetores derivados de grafos moleculares, frequentemente resultam em espaços de atributos de alta dimensionalidade. Embora tais representações sejam altamente informativas, é comum que contenham atributos redundantes, irrelevantes ou fortemente correlacionados, com potencial de impactar negativamente o desempenho dos modelos preditivos e aumentar o risco de *overfitting*.

Nesse cenário, técnicas de seleção de atributos desempenham um papel central ao identificar subconjuntos de variáveis mais relevantes para a tarefa de predição. A redução adequada do espaço de atributos contribui para melhorar a capacidade de generalização dos modelos, reduzir o custo computacional e facilitar a interpretação dos resultados, aspecto particularmente importante em aplicações de bioinformática estrutural.

De forma geral, as abordagens de seleção de atributos podem ser classificadas em métodos do tipo **filtro**, ***wrapper*** e **embutidos** [Guyon and Elisseeff 2003]. Métodos do tipo **filtro** avaliam a relevância dos atributos de maneira independente do modelo preditivo, utilizando critérios estatísticos como correlação, informação mútua ou testes univariados. Já os métodos ***wrapper*** empregam o desempenho de um modelo específico como critério de seleção, explorando diferentes subconjuntos de atributos. Essa combinação de subconjuntos de atributos inevitavelmente resulta em um maior custo computacional. Por fim, métodos **embutidos** realizam a seleção de atributos durante o próprio processo de treinamento do modelo, sendo comuns em algoritmos baseados em árvores de decisão.

Em descoberta de fármacos, algoritmos como Random Forest e métodos de *gradient boosting* oferecem mecanismos naturais para estimar a importância relativa dos atributos, sendo frequentemente utilizados tanto como modelos preditivos quanto como ferramentas auxiliares para seleção de características relevantes. Alternativamente, técnicas de redução de dimensionalidade, como a Análise de Componentes Principais (PCA), podem ser empregadas quando o objetivo é compactar a representação dos dados preservando a maior parte da variância explicada, ainda que à custa de uma menor interpretabilidade dos atributos originais.

 ©2026 SBC – Soc. Bras. de Computação

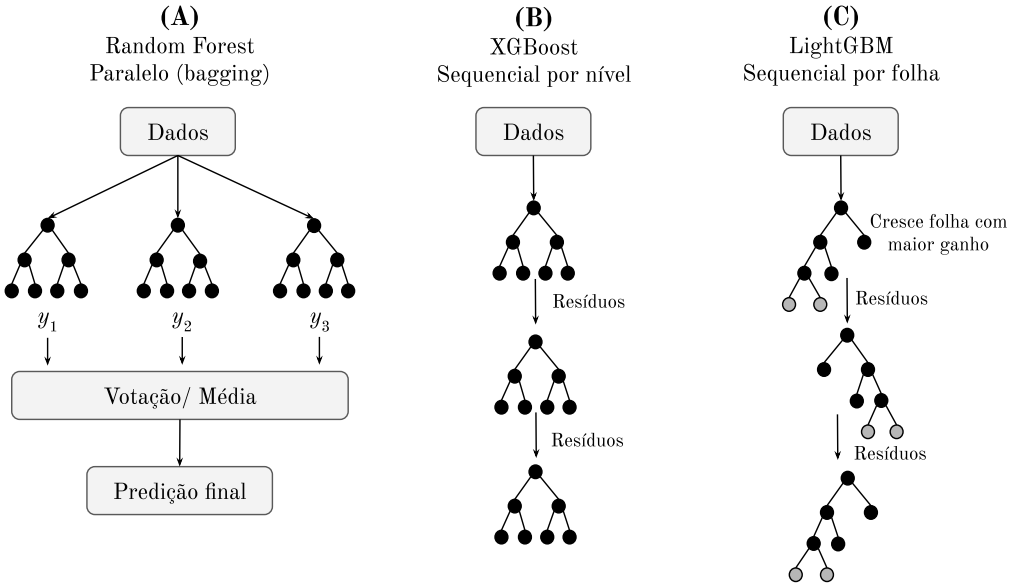


Figura 6.7: Comparação entre métodos de Aprendizado de Máquina baseados em árvores: (A) Random Forest: Ilustração do método de bagging, onde múltiplas árvores são treinadas de forma independente e paralela, gerando uma predição final por meio de votação ou média. (B) XGBoost: Estrutura de boosting sequencial com crescimento por nível (level-wise), focada na redução progressiva de resíduos. (C) LightGBM: Estrutura sequencial com crescimento por folha (leaf-wise), priorizando a expansão do nó que oferece o maior ganho de informação.

6.2.5.2. Modelos supervisionados clássicos baseados em Ensemble

Esses modelos desempenham um papel fundamental na etapa de indução de modelos preditivos, que se inicia após a preparação dos dados. Nesse contexto, técnicas de AM são empregadas para explorar representações previamente definidas e permitir a aprendizagem a partir de exemplos rotulados. Entre essas abordagens, modelos baseados em ensembles (ou conjuntos de modelos) destacam-se por sua ampla adoção em estudos acadêmicos e aplicações industriais. Tal predominância decorre da capacidade desses métodos de combinar múltiplos modelos simples, resultando em predições mais robustas, com menor variância e maior estabilidade diante de dados ruidosos e de alta dimensionalidade, frequentemente encontrados em conjuntos de dados químicos e biológicos [Hastie 2009, Murphy 2012].

Em tarefas típicas da área, como predição de atividade biológica, afinidade proteína-ligante ou propriedades ADMET, os dados disponíveis frequentemente apresentam número limitado de amostras, forte correlação entre atributos e distribuição desequilibrada entre classes. Nesse cenário, algoritmos de ensemble baseados em árvores de decisão têm se mostrado particularmente eficazes, pois são capazes de capturar relações não lineares complexas sem exigir suposições explícitas sobre a distribuição dos dados, além de apresentarem bom desempenho mesmo na presença de atributos irrelevantes.

Dentre estes modelos, o **Random Forest** destaca-se por sua simplicidade conceitual e robustez [Breiman 2001]. Essa características garantem uma grande aplicabilidade em problemas de descoberta de fármacos. O método baseia-se no princípio de bagging

 ©2026 SBC – Soc. Bras. de Computação

(bootstrap aggregating), cujo objetivo é reduzir a variância de modelos de alta complexidade, como árvores de decisão, por meio da combinação de múltiplos modelos treinados de forma independente. Nesse contexto, o Random Forest constrói um conjunto de B árvores de decisão, cada uma treinada a partir de diferentes subconjuntos aleatórios dos dados de treinamento e dos atributos disponíveis, o que contribui para mitigar o risco de sobreajuste e aumentar a capacidade de generalização do modelo, veja na Figura 6.7(A). A predição final resulta da agregação das predições individuais das árvores que compõem o ensemble, sendo calculada pela média no caso de tarefas de regressão ou por votação majoritária em problemas de classificação. Formalmente, para uma amostra de entrada x, a predição do modelo pode ser expressa como:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^{B} f_b(x) \tag{1}$$

onde $f_b(x)$ representa a predição produzida pela b-ésima árvore. Além de seu desempenho preditivo, o Random Forest fornece, como subproduto do processo de treinamento, estimativas da importância relativa dos atributos, característica particularmente relevante em descoberta de farmácos. Nessa área, compreender quais descritores ou características estruturais exercem maior influência sobre a atividade molecular pode ser tão importante quanto alcançar uma alta acurácia preditiva [Svetnik et al. 2003].

Enquanto o Random Forest explora o princípio de bagging, no qual múltiplos modelos são treinados de forma independente para reduzir a variância, outras abordagens de ensemble, como **XGBoost** e **LightGBM**, adotam uma estratégia distinta baseada em boosting [Friedman 2001]. Nessas abordagens, os modelos são construídos de maneira sequencial, com o objetivo de corrigir progressivamente os erros cometidos nas etapas anteriores. Esses algoritmos oferecem maior flexibilidade e, em muitos casos, melhor desempenho preditivo, sendo amplamente utilizados em competições e *benchmarks* de descoberta de fármacos.

Nesse contexto, o XGBoost (eXtreme Gradient Boosting) implementa o *Gradient Boosting* de forma otimizada, expandindo as árvores nível a nível (*level-wise*) para manter o equilíbrio da estrutura, [Chen and Guestrin 2016]. O algoritmo minimiza de forma iterativa uma função de objetivo regularizada, onde cada nova árvore tenta ajustar os resíduos das predições anteriores, veja Figura 6.7(B). A função de objetivo no passo t é dada por:

$$\mathscr{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

onde $\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \|w\|^2$ representa o termo de regularização para controlar a complexidade do modelo. Nessa formulação, x_i e y_i representam, respectivamente, o vetor de atributos e o rótulo verdadeiro da i-ésima amostra do conjunto de treinamento, enquanto $\hat{y}_i^{(t-1)}$ corresponde à predição acumulada do modelo até a iteração anterior. A função de perda $l(\cdot)$ mede o erro entre a predição e o valor real, podendo assumir diferentes formas conforme a tarefa de regressão ou classificação. O termo $f_t(x_i)$ representa a contribuição da árvore adicionada na iteração t, que é ajustada para modelar os resíduos das predições

 ©2026 SBC – Soc. Bras. de Computação

anteriores. O termo de regularização $\Omega(f_t)$ penaliza a complexidade da árvore, onde γ controla o número de folhas T e λ regula a magnitude dos pesos w, favorecendo modelos com melhor capacidade de generalização.

Distinto do XGBoost, o LightGBM, [Ke et al. 2017], utiliza uma estratégia de crescimento por folha (*leaf-wise*), escolhendo a folha que resulta na maior redução da função de perda (*loss*) para expansão, veja Figura 6.7(C). Essa abordagem é matematicamente eficiente pois foca no erro residual local, embora exija maior controle de profundidade para evitar *overfitting*. O ganho de divisão para uma folha é calculado como:

$$\text{Gain} = \frac{1}{2}\left[\frac{(\sum g_i)_L^2}{\sum h_i + \lambda} + \frac{(\sum g_i)_R^2}{\sum h_i + \lambda} - \frac{(\sum g_i)_{Total}^2}{\sum h_i + \lambda}\right]$$

Nessa expressão, g_i e h_i representam, respectivamente, o gradiente de primeira ordem e o hessiano de segunda ordem da função de perda em relação à predição do modelo para a i-ésima amostra. Os índices L e R denotam os subconjuntos de amostras atribuídos às folhas esquerda (*left*) e direita (*right*) após a divisão candidata, enquanto o índice $Total$ refere-se ao conjunto de amostras presente na folha original antes da divisão. Dessa forma, os termos $(\sum g_i)_L$ e $(\sum g_i)_R$ correspondem às somas dos gradientes das amostras associadas às folhas resultantes, enquanto $(\sum g_i)_{Total}$ representa a soma dos gradientes da folha original. O parâmetro λ é um hiperparâmetro de regularização L_2, introduzido para penalizar valores elevados dos pesos das folhas e controlar a complexidade do modelo, contribuindo para a redução do *overfitting*. O valor do ganho quantifica a redução esperada da função de perda decorrente da divisão da folha, sendo utilizado pelo LightGBM para selecionar, de forma gulosa, a divisão que maximiza o benefício preditivo.

Em conjunto, esses modelos supervisionados clássicos estabelecem uma linha de base sólida para aplicações de AM em descoberta de fármacos, servindo tanto como ferramentas preditivas de alto desempenho quanto como referências para a avaliação de abordagens mais complexas baseadas em aprendizado profundo.

6.2.5.3. Modelos baseados em Transformers

Os recentes avanços em aprendizado profundo têm impulsionado o uso de modelos baseados em **Transformers** em diversas áreas da ciência, incluindo a descoberta de fármacos. Originalmente desenvolvidos no contexto do processamento de linguagem natural [Vaswani et al. 2017], esses modelos foram projetados para lidar com sequências simbólicas (principalmente texto) e capturar dependências complexas presentes na linguagem humana escrita. Esse mesmo arcabouço conceitual tem sido atualmente sido estendido para outras "linguagens" que não são naturalmente interpretáveis por humanos, como a linguagem química, bioquímica e de proteínas, permitindo que modelos computacionais operem diretamente sobre essas representações simbólicas para capturar e modelar os efeitos das interações moleculares, ainda que a interpretação semântica explícita destas "linguagens" não seja conhecida.

Diferentemente de modelos supervisionados clássicos, esses métodos são capazes de aprender representações distribuídas diretamente a partir dos dados de entrada, redu-

 ©2026 SBC – Soc. Bras. de Computação

zindo a dependência de descritores manualmente projetados. A arquitetura dos Transformers permite modelar dependências de longo alcance de forma eficiente e paralelizável.

No contexto da química e da bioinformática, os Transformers têm sido aplicados com sucesso em diferentes tipos de representações, como sequências SMILES, sequências de aminoácidos e, mais recentemente, representações estruturais e multimodais [Schwaller et al. 2019]. Em tarefas como predição de atividade biológica, geração de novas moléculas, modelagem de propriedades fisicoquímicas e afinidade proteína-ligante, esses modelos demonstram elevada capacidade de capturar padrões complexos presentes em grandes volumes de dados.

Ademais, estratégias de *pré-treinamento* em grandes bases de dados químicos ou biológicos, seguidas de *fine-tuning* em tarefas específicas, têm se mostrado bastante eficazes em cenários com dados rotulados limitados, uma situação comum na descoberta de fármacos [Chithrananda et al. 2020]. Apesar de seu elevado custo computacional, e menor interpretabilidade em comparação a modelos clássicos, os Transformers representam uma classe poderosa de modelos para aplicações em larga escala e continuam a expandir seu papel em *pipelines* modernos de AM na área.

6.2.5.4. Validação e avaliação de modelos

A validação adequada de modelos de AM é um aspecto crítico na descoberta de fármacos, uma vez que desempenhos preditivos inflados podem levar a conclusões incorretas e decisões experimentais custosas. Diferentemente de aplicações tradicionais de AM, conjuntos de dados químicos e biológicos frequentemente apresentam dependências estruturais, desbalanceamento entre classes, *data leakage* e viés de amostragem, exigindo estratégias de validação cuidadosamente projetadas [Wallach and Heifets 2018].

Abordagens clássicas, como validação cruzada aleatória, são amplamente utilizadas, mas podem superestimar o desempenho do modelo quando compostos estruturalmente semelhantes aparecem simultaneamente nos conjuntos de treinamento e teste. Nesse contexto, estratégias mais rigorosas, como *scaffold splitting*, são frequentemente empregadas para avaliar a capacidade de generalização do modelo para novas classes químicas.

A escolha das métricas de avaliação deve ser compatível com a tarefa em questão. Em problemas de **classificação**, são comumente usadas métricas como AUC (*Area Under the Curve*), ROC (*Receiver Operating Characteristic*), que avalia a capacidade do modelo de discriminar entre classes ao longo de diferentes limiares de decisão, precisão e revocação, que quantificam respectivamente a proporção de predições positivas corretas e a cobertura dos exemplos positivos. Em contextos de descoberta de fármacos, métricas de enriquecimento precoce também são frequentemente empregadas, pois avaliam a capacidade do modelo de priorizar compostos ativos nas primeiras posições de uma lista ordenada, um aspecto particularmente relevante em cenários de VS. Em tarefas de **regressão**, métricas como RMSE (*Root Mean Squared Error*) e MAE (*Mean Absolute Error*) medem o erro médio das predições em relação aos valores reais, enquanto o coeficiente de determinação $\left(R^2\right)$ indica a fração da variância dos dados explicada pelo modelo [Hastie 2009]. Uma validação criteriosa, aliada à interpretação adequada dessas

 ©2026 SBC – Soc. Bras. de Computação

métricas, é fundamental para garantir que os modelos desenvolvidos apresentem desempenho confiável e relevância prática em cenários reais de descoberta de fármacos.

6.2.6. Competições de proposta de novas moléculas: CACHE e DREAM

A avaliação de modelos preditivos em descoberta de fármacos apresenta viés de sobreajuste (*overfit*) quando restrita a validações retrospectivas em bancos de dados estáticos. Desafios científicos estruturados sob os princípios da ciência aberta reduzem essa limitação ao exigir predições cegas prospectivas atreladas à validação experimental. Essas competições padronizam métricas de desempenho, expõem a taxa real de sucesso (*hit rate*) de diferentes arquiteturas de AM e forçam a construção de *pipelines* reprodutíveis sob condições de estrito isolamento de dados de teste, evitando assim o vazamento de dados.

O CACHE Challenge atua como um mecanismo direto de validação empírica para *pipelines* computacionais de VS. Um resumo das edições anunciadas até o presente momento (maio de 2026) do CACHE Challenge está apresentado na Tabela 6.1.

Tabela 6.1: Resumo das edições do CACHE Challenge (*Critical Assessment of Computational Hit-finding Experiments*)

#	Alvo	Categoria	Objetivo
1	Domínio WD40 da LRRK2	Alvo sem ligantes conhecidos	Descoberta de ligantes para o domínio WD40 da proteína LRRK2 associada à Doença de Parkinson
2	Helicase NSP13 do SARS-CoV-2	Sítio de ligação ao RNA	Identificação de compostos ligantes para a helicase viral NSP13 visando antivirais contra coronavírus
3	Macrodomínio Mac1 da NSP3 do SARS-CoV-2	Ligantes competitivos de ADP-ribose	Descoberta de novos ligantes para o macrodomínio viral evitando grupos ácido carboxílico
4	Domínio TKB da CBLB	Interação proteína-proteína	Identificação de ligantes para o domínio TKB da ubiquitina ligase CBLB
5	Tau K18 fibrils	Agregação proteica	Descoberta de moléculas capazes de interagir com fibrilas Tau associadas a doenças neurodegenerativas
6	Receptor Sigma-2/TMEM97	GPCR/alvo de membrana	Identificação de novos ligantes para o receptor Sigma-2/TMEM97
7	MPro variante resistente	Resistência antiviral	Descoberta de ligantes ativos contra variantes resistentes da protease principal do SARS-CoV-2
8	GID4 (CTLH E3 ligase)	Degradação proteica direcionada	Identificação de moléculas capazes de ocupar o bolso de reconhecimento de substrato da GID4

A organização define alvos biológicos de alta prioridade terapêutica e recebe submissões de estruturas moleculares ranqueadas *in silico* pelas equipes participantes. O consórcio executa a aquisição ou síntese química das moléculas selecionadas e conduz os

 ©2026 SBC – Soc. Bras. de Computação

ensaios biofísicos e estruturais em laboratório. O desempenho dos algoritmos é medido exclusivamente pela identificação *in vitro* de ligantes ativos reais, reduzindo significativamente artefatos estatísticos associados a modelos superajustados e conjuntos de validação artificiais.

O *First DREAM Target 2035 Drug Discovery Challenge*, operado via consórcio MAINFRAME foi estruturado em múltiplas etapas sequenciais, nas quais os participantes desenvolveram modelos capazes de identificar compostos ativos em cenários de avaliação cega. Nas etapas iniciais, os modelos foram treinados utilizando uma biblioteca DEL contendo milhões de compostos representados por *fingerprints* moleculares, juntamente com dezenas de moléculas conhecidas em formato SMILES, sendo posteriormente avaliados em conjuntos ASMS compostos majoritariamente por exemplos negativos e um número reduzido de ligantes verdadeiros (determinados por SPR) que precisam ser identificados participantes. Além da classificação e ranqueamento molecular, o desafio avalia a capacidade dos modelos em identificar compostos estruturalmente diversos e generalizar para espaços químicos previamente não observados. Na etapa final, os melhores modelos são utilizados para selecionar compostos a partir de bibliotecas químicas contendo milhões de moléculas, cujos candidatos priorizados passam por validação experimental posterior. Dessa forma, o DREAM Challenge, associado à iniciativa MAINFRAME, estabelece um ambiente colaborativo para comparação sistemática de métodos de IA em cenários mais próximos das condições reais da descoberta de fármacos.

6.3. Estratégias Computacionais para a descoberta de fármacos

Esta seção apresenta estratégias computacionais para a descoberta de fármacos, incluindo o *docking* molecular (algoritmos e FEs) e a técnica de *co-folding*.

6.3.1. *Docking* Molecular

Avanços tanto na Ciência da Computação quanto na Bioinformática têm permitido o desenvolvimento de novas estratégias computacionais aplicadas na indústria farmacêutica durante os primeiros estágios de desenvolvimento de um novo fármaco, um processo definido como RDD [Kuntz 1992, Meng et al. 2011]. A aplicação em conjunto de métodos computacionais e métodos experimentais tem importante impacto na identificação de novos compostos promissores para alvos terapêuticos, reduzindo o tempo e os custos envolvidos nesse processo [Ferreira et al. 2015].

A interação entre receptores e ligantes pode ser simulada em um nível atômico por algoritmos de *docking* molecular [Lybrand 1995, Lengauer and Rarey 1996]. O *docking* molecular é tradicionalmente formulada como um problema composto por dois componentes principais: a amostragem conformacional (*sampling*), responsável por explorar poses, orientações e conformações do ligante no sítio de ligação do receptor-alvo, e a avaliação dessas soluções por FEs, responsáveis por estimar a afinidade e selecionar as poses mais prováveis [Lengauer and Rarey 1996, Trott and Olson 2010, Pagadala et al. 2017]. Nessas simulações, centenas de milhares de orientações e conformações do ligante no sítio de ligação do receptor são avaliadas e ranqueadas de acordo com a estabilidade do complexo estimada por uma FEB ou escore predita/calculada por uma FE) [Lengauer and Rarey 1996, Halperin et al. 2002,

 ©2026 SBC – Soc. Bras. de Computação

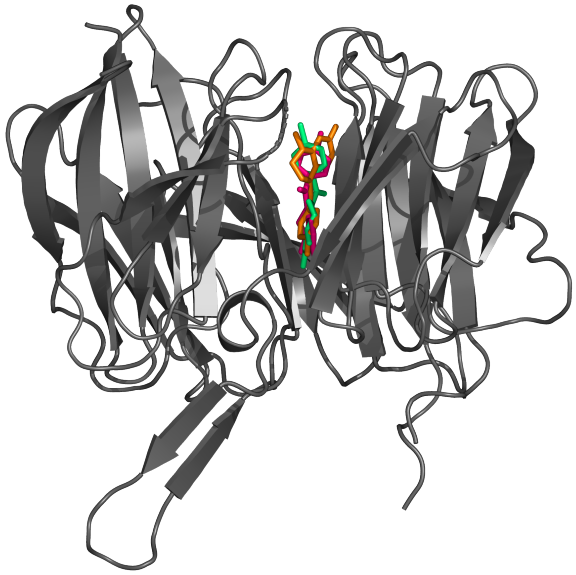


Figura 6.8: Exemplo de resultado de *docking* molecular. O receptor está em cinza, no formato de *cartoon* (Código PDB 9DTA, proteína WDR91). O ligante de exemplo está em 3 poses geradas pelo *docking* molecular com o programa AutoDock Vina 2021, no formato de *sticks* em rosa, laranja e verde. Figura gerada pelo software Pymol com resultado de *docking* gerado pelo grupo.

Meng et al. 2011, Crampon et al. 2022].

A Figura 6.8 mostra um exemplo de resultado de *docking* molecular. O receptor é a proteína WDR91 (Código PDB 9DTA), e três diferentes poses finais do ligante após a *docking* molecular são exibidas em rosa (a melhor pose), verde e laranja.

As simulações de *docking* molecular não são simples pois muitos fatores entrópicos e entálpicos influenciam a interação entre o receptor e o ligante. Sendo assim, muitos desafios estão envolvidos nesse processo e cada vez mais há a necessidade de aproximar métodos *in-silico* de métodos *in-vitro*, tornando as ferramentas computacionais de simulação mais próximas do que acontece na realidade. Entre esses desafios destacam-se a adequada representação da solvatação, efeitos induzidos por ajuste conformacional (*induced fit*), tratamento da flexibilidade do receptor e limitações inerentes das FEs em reproduzir afinidades experimentais [Pagadala et al. 2017, Pinzi and Rastelli 2019]. Além disso, há a necessidade de que novas tecnologias sejam desenvolvidas para aumentar a eficiência e eficácia da descoberta de novos fármacos. Mais recentemente, avanços em AM e aprendizado profundo vêm sendo incorporados nesse processo, ampliando o papel da *docking* molecular em *pipelines* modernos de descoberta de fármacos [Crampon et al. 2022, Scantlebury et al. 2020, Xu 2024].

6.3.1.1. Amostragem conformacional

Um dos principais desafios da *docking* molecular consiste na busca pela pose correta do ligante no sítio de ligação do receptor. Esse processo é conhecido como amostragem conformacional (*conformational sampling*) e envolve a exploração computacional das possí-

 ©2026 SBC – Soc. Bras. de Computação

veis orientações, posições e conformações do ligante durante a interação proteína-ligante [Lengauer and Rarey 1996].

O problema da amostragem conformacional é complexo devido ao alto número de graus de liberdade envolvidos. Em geral, os algoritmos de *docking* molecular consideram o receptor como rígido e permitem graus de liberdade em algumas ligações dos ligantes. Porém, mesmo pequenas moléculas podem apresentar diversas ligações rotacionáveis, resultando em um grande número de conformações possíveis. Além disso, diferentes orientações e translações do ligante no sítio ativo precisam ser consideradas simultaneamente. Em cenários mais complexos, e para aproximar o resultado *in-silico* do *in-vitro* também pode ser necessário considerar flexibilidade do receptor, aumentando ainda mais o espaço conformacional explorado.

De forma geral, os algoritmos de amostragem utilizados em *docking* molecular podem ser classificados em:

- Métodos determinísticos/sistemáticos: exploram o espaço conformacional de forma sistemática avaliando combinações discretas de ângulos torcionais e orientações possíveis. Podem apresentar alto custo computacional, especialmente para moléculas mais flexíveis;
- Métodos estocásticos: é o método utilizado pela maioria dos programas de *docking* molecular. Nesse método, o espaço conformacional é explorado de maneira probabilística, onde diferentes conformações do ligante são geradas aleatoriamente e refinadas iterativamente, buscando minimizar a energia estimada do complexo [Morris et al. 2009] calculada por uma FE. Exemplos de algoritmos comuns utilizados nesta estratégia são os algoritmos genéticos e métodos de Monte Carlo;
- Métodos baseados em fragmentos: nessa abordagem, o ligante é dividido em partes menores e menos flexíveis para reduzir a complexidade do problema. Posteriormente, os fragmentos são reconectados e refinados para reconstruir a molécula completa. Essa estratégia reduz o custo computacional associado à exploração direta de todas as conformações possíveis do ligante completo;
- Métodos baseados em AM e aprendizado profundo: utilizam modelos computacionais treinados em grandes conjuntos de dados de complexos proteína-ligante (como os disponíveis no PDBbind) para aprender padrões estruturais e energéticos associados ao processo de interação molecular. Diferentemente das abordagens tradicionais, esses métodos podem auxiliar diretamente na priorização de regiões conformacionais mais favoráveis, previsão de poses de ligação e redução do espaço conformacional explorado durante a *docking* [Crampon et al. 2022, Masters et al. 2023]

Um dos mais conhecidos métodos de amostragem é o algoritmo utilizado pelo AutoDock, baseado em Algoritmos Genéticos Lamarckianos (Lamarckian Genetic Algorithm - LGA). Nesse método, populações de poses do ligante evoluem ao longo de múltiplas gerações utilizando operações inspiradas em evolução biológica, como mutação, recombinação e seleção [Morris et al. 2009]. O termo "Lamarckiano" decorre da

 ©2026 SBC – Soc. Bras. de Computação

incorporação de refinamentos locais diretamente nos indivíduos gerados, permitindo acelerar a convergência para soluções energeticamente favoráveis. O AutoDock Vina, por sua vez, introduziu uma estratégia distinta baseada em busca global iterativa combinada com otimização local [Trott and Olson 2010]. O Vina utiliza uma representação eficiente dos graus de liberdade do ligante e explora o espaço conformacional por meio de heurísticas estocásticas associadas a refinamentos locais rápidos. Essa abordagem permitiu ganhos significativos tanto em velocidade quanto em precisão em comparação com versões anteriores do AutoDock.

Além da flexibilidade do ligante, a flexibilidade do receptor constitui outro grande desafio no *docking* molecular. Um grande número de abordagens alternativas para incorporar a flexibilidade do receptor na *docking* molecular tem sido desenvolvido, conforme revisado em [Ganesan et al. 2017, Amaro et al. 2018]. Entre as principais estratégias destacam-se o *soft-docking*, a flexibilidade das cadeias laterais, métodos de *induced fit docking* e o *ensemble docking*. No *soft-docking*, a flexibilidade do receptor é parcialmente incorporada por meio da suavização dos termos energéticos relacionados às interações de *van der Waals*, permitindo um certo grau de sobreposição entre os átomos do ligante e do receptor [Ferrari et al. 2004]. Já os métodos baseados na flexibilidade das cadeias laterais consideram mudanças conformacionais locais em resíduos do sítio de ligação, geralmente utilizando bibliotecas de rotâmeros previamente determinadas experimentalmente ou por análises estatísticas [Morris et al. 2009]. Entretanto, essas abordagens normalmente são limitadas a movimentos locais e não conseguem representar adequadamente alterações conformacionais globais da proteína [Ganesan et al. 2017].

Entre as diferentes abordagens, destaca-se o método de *ensemble docking*. Nessa estratégia, diferentes conformações do receptor podem ser combinadas em uma única representação estrutural ou utilizadas individualmente em múltiplas execuções independentes de *docking* molecular [Machado et al. 2010]. As diferentes conformações da proteína podem ser obtidas experimentalmente (e buscadas em bases de dados como o *Protein Data Bank*), ou ainda geradas computacionalmente por meio de simulações de dinâmica molecular (DM) [Amaro et al. 2018, Machado et al. 2010]. De acordo com Ganesan et al. [Ganesan et al. 2017], o *ensemble docking* tornou-se uma das abordagens mais aceitas em SBDD para incorporar a flexibilidade do receptor. Entretanto, um dos principais desafios desse método está relacionado ao elevado custo computacional, uma vez que a utilização de múltiplas estruturas do receptor implica na execução de múltiplas simulações de *docking* molecular [Scaini et al. 2019, Machado et al. 2010, Amaro et al. 2018].

Mais recentemente, estratégias baseadas em AM e aprendizado profundo têm sido propostas para auxiliar na incorporação da flexibilidade do receptor em métodos de *docking* molecular. Esses métodos buscam identificar conformações mais relevantes da proteína, reduzir o conjunto de estruturas necessárias em abordagens de *ensemble docking* ou aprender padrões conformacionais diretamente de grandes conjuntos de complexos proteína-ligante [Crampon et al. 2022]. Nesse contexto, modelos como o EDM-Dock [Masters et al. 2023] utilizam redes neurais profundas para prever diretamente matrizes de distância intermoleculares entre proteína e ligante, permitindo reconstruir poses de ligação sem a necessidade de algoritmos de busca iterativos e incorporando a flexibilidade do receptor de forma implícita por meio de representações coarse-grained das cadeias laterais. Essa abordagem reduz significativamente o custo computacional e contorna limi-

 ©2026 SBC – Soc. Bras. de Computação

tações associadas à amostragem explícita de múltiplas conformações do sítio de ligação. Essas abordagens baseadas em AM representam uma alternativa promissora para reduzir os custos computacionais associados à flexibilidade do receptor, além de possibilitar a incorporação mais eficiente de movimentos conformacionais em FEs e algoritmos de *docking* molecular

Apesar dos avanços, a amostragem conformacional ainda representa um dos principais gargalos da *docking* molecular. A dificuldade em explorar adequadamente o espaço conformacional, especialmente em sistemas altamente flexíveis, continua sendo um desafio importante para a obtenção de poses biologicamente relevantes e afinidades confiáveis.

6.3.1.2. Funções de escore - enfoque nas baseadas em Aprendizado de Máquina

De acordo com Lin et al. [Li et al. 2019], as FEs para *docking* molecular proteina-ligante podem ser organizadas em 4 tipos, conforme resume a Tabela 6.2.

Como este capítulo trata sobre AM aplicado a descoberta de fármacos, esse tipo de FE é detalhado a seguir. As FE baseadas em AM surgiram como uma alternativa às funções clássicas (baseadas em fisica e empiricas), principalmente devido à capacidade desses modelos em aprender relações complexas entre proteínas e ligantes a partir de grandes conjuntos de dados. Diferentemente das funções empíricas, baseadas em força de campo ou conhecimento, as FE baseadas em AM utilizam dados experimentais previamente conhecidos para aprender padrões associados à afinidade de ligação [Ballester and Mitchell 2010, Shen et al. 2020].

As FE baseadas em AM, conforme revisado por [Shen et al. 2020], tem utilizado diferentes algoritmos de AM, como Random Forest, *Support Vector Machines*, XGBoost, entre outros. Em geral, o desenvolvimento dessas FE envolve algumas etapas principais: obtenção do conjunto de dados para treinamento, extração de descritores moleculares, treinamento do modelo e validação [Arrua et al. 2024].

A primeira etapa consiste na obtenção de um conjunto de complexos proteína-ligante com dados experimentais de afinidade de ligação, normalmente provenientes de bases como o PDBbind [Liu et al. 2017]. A partir desses complexos, são calculados descritores que representam as interações proteína-ligante. Esses descritores podem incluir termos energéticos, propriedades físico-químicas, *fingerprints* moleculares, contatos atômicos, informações estruturais [Ballester and Mitchell 2010, Shen et al. 2020]. A qualidade da representação dos dados possui impacto direto no desempenho do modelo [Shen et al. 2020].

A validação do modelo é uma etapa fundamental para avaliar a capacidade de generalização da FE. Diversas métricas podem ser utilizadas, incluindo coeficiente de correlação de Pearson (R), Root Mean Square Error (RMSE), *Mean Absolute Error* (MAE). Além dessas métricas tradicionais de regressão, um dos *benchmarks* mais utilizados para avaliação de FEs é o CASF (Comparative Assessment of Scoring Functions) [Su et al. 2018], que avalia diferentes aspectos das FE, incluindo capacidade de previsão de afinidade de ligação (scoring power), predição de identificaçao de ligantes nativos entre *decoys* (*docking power*), ranqueamento de ligantes (*ranking power*) e enriquecimento

 ©2026 SBC – Soc. Bras. de Computação

Tabela 6.2: Classificação das funções de escore para *docking* proteína-ligante

Tipo	**Princípio**	**Vantagens**	**Limitações**	**Exemplos**
Baseadas em física (*Physics-based*)	Baseadas em termos físicos derivados da mecânica molecular, modelando interações como van der Waals e eletrostáticas	Interpretação física clara; base teórica consistente	Custo computacional mais alto; tratamento simplificado de solvente	DOCK [Allen et al. 2015]
Empíricas (*Empirical*)	Combinação de termos que representam diferentes interações proteína-ligante, com pesos ajustadosa partir de dados experimentais de afinidade	Maior eficiência computacional; boa correlação com dados experimentais no domínio de treino	Dependência do conjunto de treinamento; menor capacidade de generalização	Vina [Trott and Olson 2010]
Basedas em Conhecimento (*Knowledge-based*)	Potenciais derivados de análises estatísticas de complexos proteína-ligante	É o tipo de FE mais rápida; capturam padrões estruturais recorrentes	Dependência da qualidade e tamanho do banco de dados; interpretação indireta	Convex-PL [Kadukova et al. 2021]
Baseadas em Aprendizado de Máquina	Modelos de AM supervisionados que aprendem relações entre descritores estruturais e afinidade de ligação experimental	Bom desempenho preditivo; capturam relações não lineares complexas	Baixa interpreta-bilidade; forte dependência de dados de treinamento	RFL-Score[Arrua et al. 2024], CRRF-Score [Werhli et al. 2025], MedusaGraph [Jiang et al. 2022], Pafnucy [Stepniewska-Dziubinska et al. 2018]

 ©2026 SBC – Soc. Bras. de Computação

em VS (*screening power*) [Su et al. 2018].

O ComBi-Lab, na FURG, tem desenvolvido FE baseadas em AM. Um exemplo é a RFL-Score [Arrua et al. 2024] que consiste em uma FE desenvolvida a partir de um conjunto híbrido de dados contendo complexos proteína–ligante experimentais e estruturas *decoys*. A representação dos complexos é construída por meio da extração de um conjunto amplo de 722 descritores provenientes do receptor, do ligante e de suas interações. Para reduzir a dimensionalidade e identificar atributos relevantes, foi aplicada a regressão *Lasso* como método de seleção de atributos. A FE propriamente dita é então construída utilizando o algoritmo *Random Forest*. A validação da FE RFL-Score foi realizada utilizando o *benchmark* CASF-2016 [Su et al. 2018] e os resultados mostraram que o RFL-Score apresenta desempenho competitivo, alcançando valores de correlação próximos ao estado da arte.

Duas áreas ganharam atenção crescente nos últimos anos nesse contexto: o *Deep Learning* - DL (Aprendizado Profundo) e os modelos generativos. Em FE para *docking* molecular, modelos de *Deep Learning* têm sido utilizados para aprender automaticamente representações complexas das interações proteína-ligante diretamente a partir de estruturas tridimensionais dos complexos moleculares [Shen et al. 2020, Meli et al. 2022]. Diferentes arquiteturas têm sido propostas para essa finalidade, incluindo redes neurais convolucionais tridimensionais (3D-CNNs), redes baseadas em grafos moleculares, entre outros. Essas abordagens permitem capturar relações espaciais e padrões não lineares das interações moleculares com maior capacidade de generalização quando comparadas a FEs tradicionais.

Entre as muitas FE baseadas em DL, uma das precursoras foi oa Pafnucy [Stepniewska-Dziubinska et al. 2018] que propôs redes neurais convolucionais 3D aplicadas a representações em grade dos complexos proteína–ligante para predizer afinidade de ligação. Também destaca-se o GNINA, cuja versão 1.3 [McNutt et al. 2025] utiliza redes neurais convolucionais 3D como FE e para avaliar e reclassificar as poses geradas no *docking*, a partir de representações tridimensionais da interação proteína-ligante, melhorando a precisão sem alterar o processo tradicional de amostragem. No contexto de FE baseadas em DL, tem-se as redes do tipo *Graph Neural Network* (GNN), que permitem representar o complexo proteína–ligante como um grafo, no qual os átomos correspondem a nós e as interações entre eles são modeladas como arestas, incorporando diretamente informações estruturais do sistema [Jiang et al. 2022]. Nesse sentido, o método MedusaGraph utiliza duas redes neurais baseadas em grafos: a primeira ajusta a posição dos átomos do ligante a partir de uma pose inicial, refinando a conformação no sítio de ligação, enquanto a segunda avalia a qualidade da pose resultante, classificando-a como próxima ou distante da estrutura experimental [Jiang et al. 2022]. Dessa forma, o modelo reduz a dependência de FE tradicionais e melhora a identificação de poses relevantes.

Mais recentemente, os *Large Language Models* (LLMs) também passaram a despertar interesse em aplicações relacionadas à descoberta de fármacos e bioinformática estrutural. Embora originalmente desenvolvidos para tarefas de processamento de linguagem natural, esses modelos podem ser adaptados para aprender padrões presentes em sequências biológicas, representações moleculares e interações proteína-ligante. Nesse contexto, arquiteturas baseadas em Transformers [Vaswani et al. 2017] têm sido utiliza-

 ©2026 SBC – Soc. Bras. de Computação

das para modelagem de sequências de proteínas, geração de representações moleculares e aprendizado multimodal integrando informações estruturais e químicas. Revisões recentes indicam que esses LLMs também vêm sendo aplicados à predição de afinidade de ligação proteína-ligante a partir de sequências e representações textuais [Zheng et al. 2025], ainda que, diferentemente das FE tradicionais, não operem diretamente sobre a geometria tridimensional de poses.

Essas abordagens baseadas em DL e LLMs representam uma mudança importante no desenvolvimento de FE para *docking* molecular, permitindo reduzir a dependência de descritores definidos manualmente e possibilitando que os próprios modelos aprendam representações relevantes diretamente dos dados. Dessa forma, esses modelos tem potencial para contribuir tanto na predição de afinidade de ligação quanto na identificação de padrões moleculares complexos associados ao reconhecimento proteína-ligante.

6.3.2. Co-folding

Esta seção apresenta o *co-folding* é tratado como uma classe de métodos computacionais que predizem simultaneamente a geometria de múltiplos componentes moleculares (proteínas, ácidos nucleicos, ligantes, íons ou modificações pós-traducionais) a partir de representações de sequência e estrutura química.

Como discutido nas seções anteriores, o *docking* molecular é uma das estratégias mais utilizadas em estudos de SBDD. De modo geral, essa abordagem parte de uma estrutura tridimensional do alvo biológico, usualmente uma proteína, e busca predizer a pose mais provável de uma pequena molécula em seu sítio de ligação. Apesar de sua ampla aplicação, o *docking* molecular ainda apresenta limitações importantes, especialmente no tratamento da flexibilidade do receptor, na descrição de ajustes conformacionais induzidos pelo ligante e na estimativa quantitativa da afinidade proteína-ligante.

6.3.2.1. Da modelagem do receptor ao aprendizado profundo

Uma alternativa para explorar novas conformações de alvos não disponíveis experimentalmente é realizar a predição tridimensional do receptor. A modelagem por homologia foi, por décadas, uma das principais estratégias utilizadas. Essa abordagem depende da existência de uma estrutura molde relacionada, normalmente com identidade de sequência acima de 30% e cobertura suficientes para permitir a construção de um modelo confiável. Essa qualidade depende também da conservação das regiões funcionais, da qualidade do alinhamento e da resolução da estrutura experimental utilizada como referência. Entretanto, para proteínas sem moldes próximos, regiões flexíveis ou domínios pouco caracterizados, a acurácia desses métodos era frequentemente limitada inviabilizando o *docking* molecular para diversos potenciais alvos.

Nesse contexto, métodos recentes baseados em aprendizado profundo têm surgido como uma alternativa ou complemento às abordagens tradicionais, ao tentar predizer diretamente a estrutura de complexos biomoleculares. Essa estratégia é frequentemente denominada *co-folding*, pois busca modelar simultaneamente a conformação dos componentes moleculares e suas interações no complexo [Senior et al. 2020].

 ©2026 SBC – Soc. Bras. de Computação

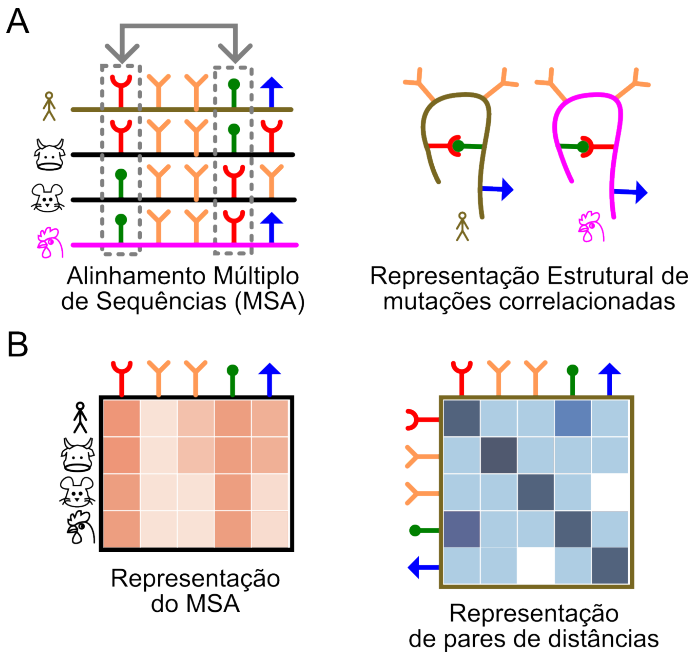


Figura 6.9: Esquema do uso de informação evolutiva na predição estrutural de proteínas. (A) Alinhamentos múltiplos de sequências permitem identificar posições que variam de forma correlacionada ao longo da evolução. Essas correlações podem indicar pares de resíduos que interagem ou que estão espacialmente próximos na estrutura tridimensional, uma vez que mutações em uma posição podem ser compensadas por mutações em outra posição. (B) Em modelos modernos de predição estrutural, como o AlphaFold2, as informações extraídas do MSA são convertidas em representações numéricas e integradas a representações de pares de resíduos, que codificam relações geométricas, como distâncias esperadas entre aminoácidos. Essa integração entre representação evolutiva e representação geométrica auxilia a inferência da estrutura tridimensional da proteína. Figura adaptada de Jumper et al. [Jumper et al. 2021].

Esse cenário foi profundamente alterado com o desenvolvimento de métodos de predição estrutural baseados em aprendizado profundo, em especial a partir das contribuições de Dr. John Jumper, Dr. Demis Hassabis e colaboradores da DeepMind com o AlphaFold (AF). A primeira versão do método, AF1, explorava informações de covariação evolutiva extraídas de alinhamentos múltiplos de sequência, ou MSA, do inglês *multiple sequence alignment* [Altschuh et al. 1987]. A premissa central é que resíduos espacialmente próximos em uma proteína tendem a apresentar padrões correlacionados de mutação ao longo da evolução (Figura 6.9). Assim, quando uma mutação ocorre em uma posição da sequência, outra posição próxima no espaço tridimensional pode sofrer uma mutação compensatória, preservando interações estruturais ou funcionais. Essas informações evolutivas são utilizadas para inferir relações entre pares de resíduos, como contatos ou distribuições de distância e ângulos de torção, que auxiliam na construção do modelo tridimensional [Senior et al. 2020]. Apesar do avanço em relação a métodos anteriores, a acurácia ainda era limitada, especialmente para alvos sem bons moldes.

6.3.2.2. AlphaFold2: predição estrutural, atenção e métricas de confiança

O novo conceito principal do AF2 [Jumper et al. 2021] foi o uso de mecanismos inspirados em Transformers, arquitetura baseada em atenção e também presente em modelos

 ©2026 SBC – Soc. Bras. de Computação

como chatGPT (*Generative Pre-trained Transformer*). A atenção funciona de forma parecida com a tradução de uma frase: para traduzir uma palavra, o modelo não olha apenas para essa palavra isoladamente, mas também para outras palavras da frase que ajudam a definir seu significado. Por exemplo, em "*bank of the river*", a palavra "*river*" recebe mais peso para traduzir "*bank*" como "margem", e não como "banco". No AF, ocorre algo análogo: para atualizar a informação de um resíduo, o modelo aprende a dar mais peso a outros resíduos, posições do MSA ou pares de resíduos que ajudam a definir sua posição na estrutura 3D. Todos esses elementos são convertidos em representações numéricas, que são o formato compreendido pelo computador (*embeddings*) e que permite estabelecer relações [Jumper et al. 2021].

AF2 propôs o Evoformer (Figura 6.10 e Tabela 6.3), modelo que integra uma representação evolutiva (gerado a partir do MSA) e uma representação geométrica (relacionado aos pares de distâncias). A representação de pares é atualizada por operações triangulares, nas quais a relação entre dois resíduos é refinada usando um terceiro resíduo como contexto, favorecendo consistência geométrica em 3D. Quando uma hipótese geométrica se mostra inconsistente, as atualizações entre MSA e representação de pares permitem redistribuir a atenção e refinar a representação. Após 48 blocos de análises, um módulo estrutural constrói os resíduos através da predição da rotação e translação de cada resíduo a partir do modelo anterior. Após 3 ciclos entre o Evoformer e o módulo estrutural, hipótese(s) de estruturas da proteína (em formato PDB) são geradas [Jumper et al. 2021].

Um aspecto particularmente importante do AF2 foi a associação dos modelos gerados a métricas internas de confiança conforme ilustrado na Figura 6.11 contendo o modelo predito da WDR91 a partir de sua sequência inteira. O pLDDT fornece uma estimativa local da confiabilidade da predição, indicando quais regiões da estrutura tendem a estar bem modeladas e quais devem ser interpretadas com maior cautela [Jumper et al. 2021]. Já a matriz PAE, ou *Predicted Aligned Error*, estima o erro esperado na posição relativa entre pares de resíduos ou regiões da proteína. Essas métricas são fundamentais para o uso prático dos modelos, pois permitem avaliar se uma região de interesse, como um sítio de ligação, domínio funcional ou interface de interação, apresenta suporte suficiente para análises posteriores. Outra interpretação para resíduos em loops com baixa confiança é serem regiões intrinsecamente desordenadas, uma observação empírica feita por alguns estudos. Assim, o AF2 forneceu ferramentas para interpretar a confiabilidade dessas predições.

O sucesso do AF2 foi obtido quando esta ferramenta desbancou todos os antigos métodos baseados em conceitos físicos na edição do 14º CASP (*Critical Assessment of Structure Prediction*) ao predizer o enovelamento de proteínas com um erro semelhante ao observado experimentalmente [Kryshtafovych et al. 2021]. Nesta competição, estruturas com enovelamentos inéditos são determinadas experimentalmente sem o conhecimento dos competidores, que têm a chance de predizê-las a partir da sequência.

Outro importante impacto do AF2 foi a criação do AlphaFold Protein Structure Database, ou AFDB, desenvolvido em parceria entre a Google DeepMind e o EMBL-EBI [Varadi et al. 2022] descrito na seção 6.2.2.1. Para a SBDD, esse avanço foi relevante porque aumentou o número de proteínas potencialmente analisáveis por métodos estru-

 ©2026 SBC – Soc. Bras. de Computação

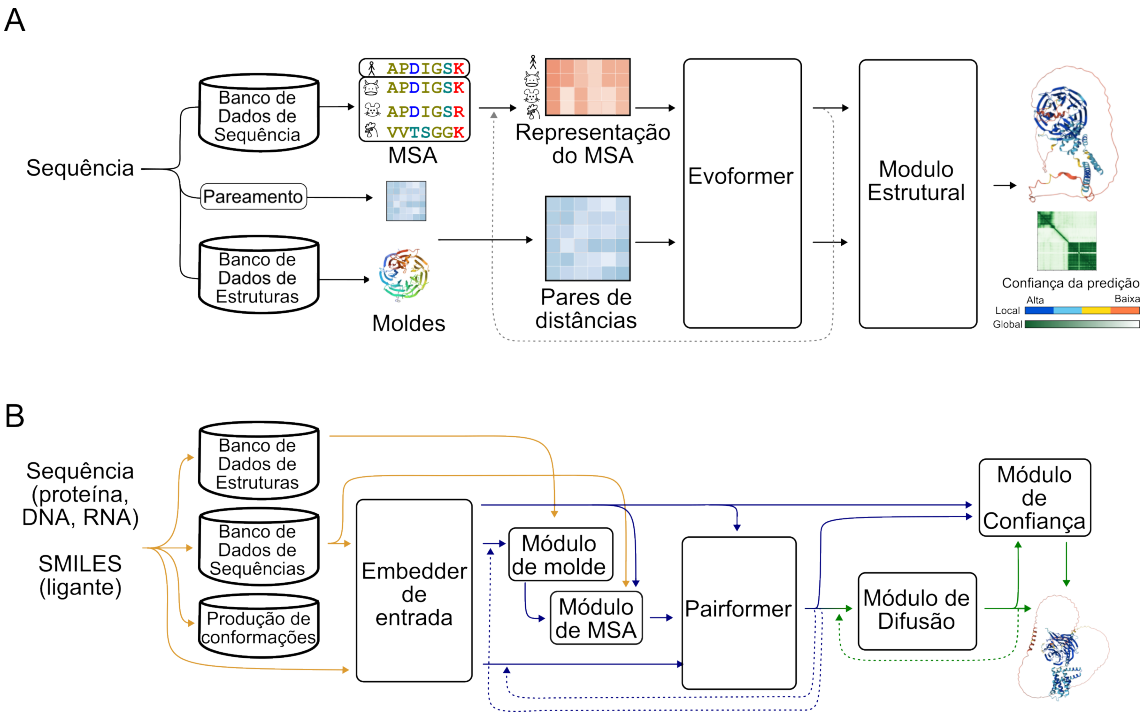


Figura 6.10: Comparação esquemática entre as arquiteturas do AlphaFold2 e do AlphaFold3. (A) O AlphaFold2 utiliza a sequência proteica, alinhamentos múltiplos de sequência (MSA) e moldes estruturais para gerar representações de MSA e de pares de distâncias, que são processadas pelo Evoformer e convertidas em uma estrutura tridimensional pelo módulo estrutural. (B) O AlphaFold3 expande essa lógica para complexos biomoleculares, incorporando proteínas, ácidos nucleicos e ligantes, e substitui o módulo estrutural por um módulo de difusão condicionado por representações processadas pelo Pairformer. Figura adaptada de Jumper *et al.* [Jumper et al. 2021] e Abramson *et al.* [Abramson et al. 2024].

 ©2026 SBC – Soc. Bras. de Computação

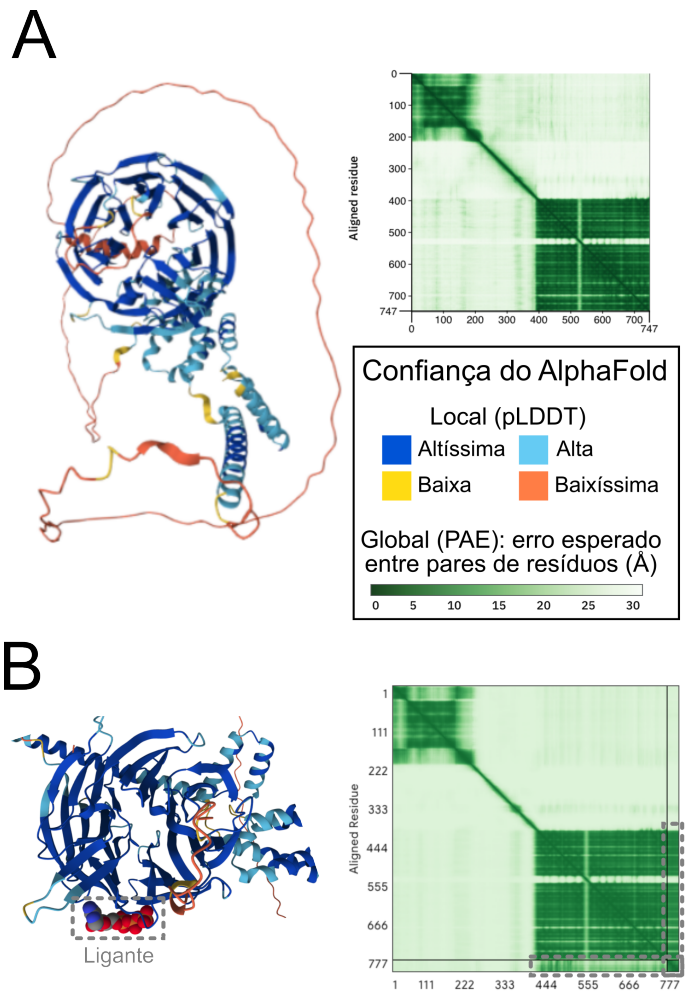


Figura 6.11: Comparação entre modelos da WDR91 preditos por AlphaFold2 (A) e AlphaFold3 (B). Em A, é mostrado o modelo da proteína completa, acompanhado das métricas de confiança local (pLDDT) e global (PAE). Em B, apenas o domínio WDR é representado em *cartoon*, enquanto o ligante predito pela estratégia de *co-folding* é representado por esferas e destacado por um retângulo tracejado.

©2026 SBC – Soc. Bras. de Computação

turais. No entanto, modelos do AFDB devem ser utilizados com cautela em aplicações de *docking* molecular, principalmente quando o sítio de ligação apresenta baixa confiança local, quando há regiões desordenadas ou quando a conformação funcional depende da presença de ligantes, cofatores, membranas ou parceiros biomoleculares.

6.3.2.3. AlphaFold3 e a emergência do co-folding para complexos biomoleculares

O AF3 expandiu esse paradigma ao propor um modelo voltado à predição conjunta de complexos biomoleculares, incluindo proteínas, DNA, RNA, pequenas moléculas e modificações pós-traducionais [Abramson et al. 2024] (Figura 6.10B e Tabela 6.3). Essa abordagem é frequentemente descrita como co-folding, pois busca predizer simultaneamente a conformação dos componentes e suas interações no complexo, diferentemente do *docking* molecular.

Na arquitetura do AF3 [Abramson et al. 2024] (Figura 6.10), diferentes entradas são processadas por diferentes abordagens: i) a sequência da macromolécula é utilizada para realizar uma busca por moldes (templates) e ser processada no módulo de molde; essa sequência também será utilizada para construir o MSA e processá-lo para gerar a matriz de pares de distâncias que alimentará o módulo de MSA, que reduz a matriz e os cálculos reduzidos; iii) moléculas representadas por SMILES têm seus confôrmeros gerados pelo RDKit. Diferentemente do AF2, o AF3 substitui o Evoformer por uma arquitetura denominada Pairformer e utiliza um modelo de difusão no módulo estrutural. Em termos conceituais, modelos de difusão aprendem a reconstruir dados a partir de versões dos dados progressivamente corrompidas por ruído. Seria como transformar uma imagem em um chiado de televisão antiga e aprender como resgatar o início a partir do ruído. No caso do AF3, a estrutura inicial pode ser entendida como um conjunto de átomos distribuídos de forma ruidosa no espaço, que é progressivamente refinado com base nas representações aprendidas pelo modelo.

6.3.3. Novos modelos generativos para co-folding com novas funções

Diante do sucesso do AF3, outros grupos desenvolveram modelos generativos ou foundation models para co-folding. Um dos primeiros foi o RoseTTAFold All-Atom do Dr. David Baker [Krishna et al. 2024] (Tabela 6.3), também utilizando um modelo de difusão, complexos iniciam com todos os constituintes aleatoriamente dispostos no espaço que vão se ordenando com informações evolutivas, sobre moldes e no caso dos ligantes, por sua representação em grafos.

Outras abordagens, como Chai-1 (Tabela 6.3) e Boltz, também avançaram ao permitir a inclusão de restrições espaciais como parte da entrada do modelo [Chai Discovery Team 2024]. Essas restrições podem ter origem em dados experimentais ou em conhecimento prévio sobre o sistema, como contatos esperados, distâncias aproximadas ou regiões envolvidas em interação. Essa capacidade é particularmente relevante para aplicações em biologia estrutural integrativa, nas quais informações incompletas de diferentes fontes podem ser combinadas para orientar a predição de complexos biomoleculares.

Mais recentemente, propostas como Boltz-2 (Tabela 6.3) passaram a sugerir tam-

 ©2026 SBC – Soc. Bras. de Computação

bém a predição de afinidade proteína-ligante e o desenho de novas moléculas (*de novo design*) com possibilidade de síntese [Passaro et al. 2025]. Essas capacidades, se acuradas, poderiam não apenas encontrar novos *hits* como também otimizá-los a leads. Para alcançar tais objetivos, os modelos foram treinados após a aplicação de curadoria a diversos bancos de dados, como ChEMBL, BindingDB, PubChem, entre outros. No entanto, como algumas dessas abordagens ainda foram divulgadas inicialmente como preprints, suas capacidades devem ser interpretadas com cautela até validação mais ampla pela comunidade.

Tabela 6.3: Comparação entre programas de predição estrutural baseados em aprendizado profundo e métodos recentes de *co-folding* para complexos biomoleculares.

Programa	**Princípio**	**Vantagens**	**Referência**
AF2	Evoformer + módulo estrutural	Marco na predição estrutural de proteínas; não utiliza difusão generativa e tende a sinalizar regiões incertas por métricas de confiança	[Jumper et al. 2021]
AF3	Pairformer + difusão	Acelera a predição estrutural e expande a modelagem para complexos biomoleculares heterogêneos	[Abramson et al. 2024]
RFAA	Difusão + representação atômica	Modelagem em nível atômico com refinamento da geometria final dos complexos	[Krishna et al. 2024]
Chai1	Modelo generativo com difusão	Permite incorporar restrições espaciais ou conhecimento prévio como entrada do modelo	[Chai Discovery Team 2024]
Boltz2	Modelo generativo / foundation model	Propõe integração entre predição estrutural e estimativa de afinidade proteína-ligante	[Passaro et al. 2025]

6.3.4. Limitações e uso crítico em SBDD

Apesar do entusiasmo em torno dessas abordagens, é importante interpretar seus resultados com cautela. Em uma avaliação sistemática, Škrinjar *et al*. demonstraram que abordagens atuais de *co-folding* tendem a memorizar poses de ligantes presentes no treinamento [Škrinjar et al. 2026]. De forma semelhante, o *benchmark* FoldBench mostrou que o sucesso na modelagem de ligantes aumenta com a similaridade química em relação ao conjunto de treinamento, sugerindo que esses modelos são mais robustos para recapitular complexos conhecidos ou quimicamente próximos do que para generalizar para ligantes verdadeiramente novos [Xu et al. 2026]. Essas observações são indicativos que *benchmarkings* mais realistas e desconhecidos precisam ser adotados e que a capacidade destes modelos de generalizar para complexos verdadeiramente inéditos e impõe cautela em aplicações de descoberta de moléculas em abordagem *de novo*.

Alguns estudos reportaram limitações do AF3 na predição de estados conformacionais alternativos já caracterizados experimentalmente. Este é o caso para ubiquitina ligase E3 que um estado aberto e outro fechado foram determinados experimentalmente,

 ©2026 SBC – Soc. Bras. de Computação

contudo apenas o último é predito [Abramson et al. 2024]. Outros exemplos são de proteínas órfãs, i.e., sem ou com poucas proteínas homólogas nos bancos de dados de sequências. Estratégias de aumentar número de modelos gerados ou aumentar as sementes (seeds) que podem melhor a confiança dos modelos gerados.

A arquitetura generativa do AF3 implica que às vezes produz categorias de erro decorrente de ordem estrutural espúria (alucinações) em regiões desordenadas que não foram observadas no AF2. Neste, regiões alucinadas geralmente são marcadas com baixíssima confiança, com pontuações pLDDT bem abaixo de 50, e em conformação de loop disperso no espaço. No entanto, AF3 pode apresentá-las como hélices [Abramson et al. 2024].

Outra limitação importante é o maior tempo de predição dos métodos de *co-folding*, geralmente na ordem de minutos por complexo, em comparação ao *docking* molecular, que costuma operar na escala de segundos por ligante. Esse custo computacional ainda limita sua aplicação direta em etapas de *virtual screening* em larga escala, nas quais é necessário avaliar milhares ou milhões de moléculas. Assim, para que métodos de *co-folding* sejam incorporados de forma rotineira em *pipelines* de descoberta de fármacos, será necessário reduzir substancialmente seu custo computacional ou utilizá-los de maneira complementar, por exemplo, na reclassificação de conjuntos menores previamente filtrados por *docking*, AM ou triagens experimentais.

A predição correta da geometria de um complexo não implica necessariamente a predição correta de afinidade, seletividade ou atividade funcional. Interações proteína-ligante dependem de fatores físico-químicos complexos, incluindo efeitos entrópicos, solvatação, protonação, tautomeria, flexibilidade conformacional e estados funcionais alternativos da proteína. Assim, o *co-folding* deve ser entendido menos como substituto direto do *docking* molecular e mais como uma nova camada de modelagem estrutural probabilística. Seu maior valor está em gerar hipóteses tridimensionais para complexos biomoleculares, explorar modos de interação plausíveis e integrar informações estruturais, evolutivas e químicas em *pipelines* de descoberta de fármacos. Entretanto, sua aplicação em SBDD ainda exige validação experimental, controle rigoroso de vazamento de dados e interpretação crítica das métricas de confiança, especialmente quando o objetivo envolve afinidade, seletividade ou desenho de moléculas inéditas.

6.4. Estudo de Caso: WDR91

A proteína WDR91 é uma escolha interessante do ponto de vista didático e científico porque representa um alvo desafiador, pouco explorado farmacologicamente e associado a dados experimentais recentes de triagem molecular de DEL. Proteínas desse tipo são exemplos importantes do problema enfrentado por iniciativas como o Target 2035: há muitos alvos biologicamente relevantes para os quais ainda não existem ligantes químicos bem caracterizados ou ferramentas moleculares amplamente disponíveis.

A WDR91 é uma proteína contendo um domínio de repetição Trp-Asp (WDR) com enovelamento característico de β-*propeller*. Essa classe de proteínas são codificadas por aproximadamente 350 genes e intrinsicamente envolvidos em doenças [Ackloo et al. 2025]. A WDR91 regula a conversão precoce do endossomo e desempenha papéis vitais na fusão, reciclagem e transporte do endossomo, sendo importante em

 ©2026 SBC – Soc. Bras. de Computação

processos do desenvolvimento neuronal e apontada como potencial fator hospedeiro para infecção viral por experimentos de *wide-genome CRISPR* [Ahmad et al. 2023]. Dessa forma, essa classe de proteínas é frequentemente associada à formação de plataformas de interação proteína-proteína, participando de processos celulares regulatórios relevantes para DD. Esse tipo de arquitetura estrutural pode tornar a descoberta de ligantes particularmente desafiadora, pois seus sítios de interação nem sempre correspondem a cavidades profundas e bem definidas, como aquelas observadas em enzimas clássicas. Por isso, a identificação de pequenas moléculas capazes de interagir com proteínas desse tipo é um problema relevante tanto para a biologia química quanto para o desenvolvimento de novas estratégias computacionais.

No contexto do minicurso, o domínio WDR da proteína WDR91 (região compreendida entre os resíduos 392 a 747) será tratada como um alvo-modelo para demonstrar como dados experimentais de triagem podem ser convertidos em um problema de classificação molecular. Cada molécula presente no conjunto de dados é associada à uma representação química computacional e a uma classe experimental: positiva ou negativa. O objetivo do modelo será aprender padrões moleculares associados ao enriquecimento experimental observado na triagem DEL. A partir disso, os alunos poderão treinar modelos capazes de estimar a probabilidade de uma nova molécula pertencer à classe de compostos potencialmente ligantes.

É importante destacar que, em dados de triagem experimental, as classes positiva e negativa não devem ser interpretadas de forma absoluta. Um composto classificado como positivo não é necessariamente um ligante validado com alta afinidade, assim como um composto classificado como negativo não é necessariamente incapaz de se ligar à proteína. Em DEL, por exemplo, a classificação pode refletir enriquecimento relativo sob determinadas condições experimentais. Portanto, é mais adequado interpretar os positivos como compostos enriquecidos ou candidatos a ligantes, e os negativos como compostos não enriquecidos ou sem evidência de ligação nas condições da triagem. Essa distinção é fundamental para que os alunos compreendam as incertezas e limitações inerentes a dados experimentais reais.

6.4.1. Subconjunto utilizado no minicurso

Para fins didáticos e escalonáveis para computadores domésticos, o minicurso utilizará uma versão simplificada de dados DEL associados à WDR91 [Ahmad et al. 2023]. O subconjunto será composto por aproximadamente 30 mil exemplos positivos e 150 mil exemplos negativos descritos por apenas um *fingerprint*, ECFP4, resultando em uma proporção aproximada de um positivo para cada cinco negativos. Essa distribuição mantém uma característica importante de problemas reais de triagem molecular: o desbalanceamento entre classes. Em campanhas experimentais, compostos com evidência de ligação costumam representar apenas uma pequena fração do total avaliado. Assim, mesmo em uma versão reduzida, o conjunto de dados preserva um desafio central para modelos de classificação em descoberta de fármacos.

Cada instância apresenta os ID identificadores dos *building blocks* usados para gerar a molécula final, que é representada pelo *fingerprint* ECFP4. *Fingerprints* moleculares são representações numéricas da estrutura química, utilizadas para converter moléculas

 ©2026 SBC – Soc. Bras. de Computação

em vetores que podem ser interpretados por algoritmos de AM. O ECFP4, em particular, descreve ambientes atômicos circulares ao redor de cada átomo da molécula, capturando padrões locais de conectividade química. Essa representação é amplamente utilizada em quimioinformática porque permite comparar moléculas, calcular similaridade química e treinar modelos preditivos de atividade biológica.

A escolha de utilizar apenas o ECFP4 tem uma finalidade pedagógica. Em aplicações reais, é comum combinar diferentes tipos de descritores, como *fingerprints* adicionais, propriedades físico-químicas, representações por grafos, modelos tridimensionais, informações estruturais da proteína, *docking* ou *co-folding*. No entanto, para um primeiro contato com AM aplicado à descoberta de fármacos, o uso de uma única representação molecular reduz a complexidade técnica e permite que os alunos compreendam com clareza as etapas fundamentais do processo. Assim, o foco do minicurso será entender como preparar os dados, dividir o conjunto em treino e teste, lidar com desbalanceamento, treinar classificadores, avaliar métricas de desempenho e interpretar os resultados obtidos.

Além do treinamento e da avaliação inicial dos classificadores com os dados DEL, os modelos desenvolvidos no minicurso serão posteriormente pontuados em um conjunto independente de teste composto por moléculas identificadas por ASMS e validadas experimentalmente por SPR. Essa etapa permitirá avaliar se os padrões aprendidos a partir dos dados de enriquecimento em DEL são capazes de generalizar para compostos detectados por uma tecnologia experimental distinta e confirmados por um método ortogonal de ligação. Dessa forma, o conjunto de teste baseado em ASMS/SPR funcionará como uma validação mais rigorosa e biologicamente relevante dos modelos, aproximando a atividade prática de um cenário real de priorização de moléculas em descoberta de fármacos.

Após a descrição do estudo de caso específico da proteína WDR91, é útil abstrair os detalhes particulares e apresentar o fluxo de trabalho geral que orienta a aplicação de AM nesse tipo de problema. A Figura 6.12 apresenta uma estratégia conceitual e genérica para a seleção de moléculas em desafios de identificação de ligantes para um determinado alvo biológico.

No fluxo ilustrado na Figura 6.12, o painel (1) representa os conjuntos de dados de entrada e suas respectivas representações computacionais, destacando o uso de *fingerprints* moleculares para descrever compostos provenientes de bibliotecas codificadas por DNA (DEL), nas quais as estruturas químicas não estão explicitamente disponíveis na forma de SMILES. O painel (2) apresenta diferentes estratégias de seleção de atributos, aplicadas com o objetivo de reduzir a dimensionalidade e ressaltar características moleculares relevantes. Em seguida, o painel (3) descreve as etapas de treinamento e validação dos modelos de AM. Por fim, os painéis (4) e (5) ilustram a aplicação do conjunto de teste, composto por dados do tipo ASMS, ao modelo treinado, culminando no ranqueamento das moléculas candidatas de acordo com seu ranking estimado de atividade.

É importante ressaltar que a Figura 6.12 tem caráter conceitual e genérico, ilustrando um fluxo de trabalho típico para a aplicação de AM na identificação e priorização de moléculas bioativas a partir de dados de triagem molecular. A estratégia apresentada não corresponde exclusivamente ao caso da proteína WDR91, mas representa um arcabouço metodológico geral, aplicável a diferentes alvos biológicos e cenários experimentais, nos quais dados de treinamento e de teste podem ser provenientes de fontes

 ©2026 SBC – Soc. Bras. de Computação

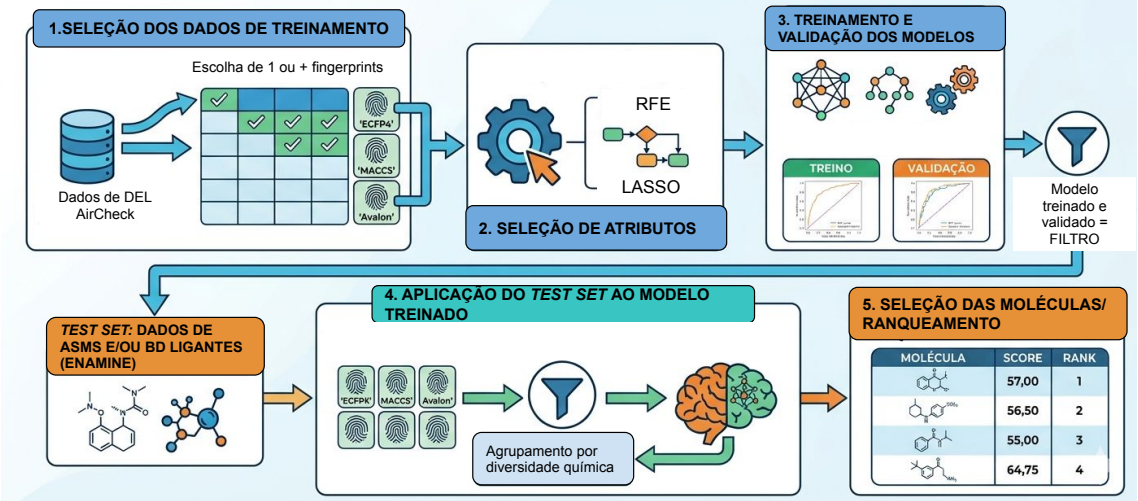


Figura 6.12: Proposta de estratégia de aplicação de Aprendizado de Máquina para a seleção de moléculas ativas em um cenário onde os dados de treinamento são provenientes de bibliotecas codificadas por DNA (DEL) e os dados de teste do tipo ASMS. A figura apresenta os dados e suas representações como *fingerprints* no painel 1. Já no painel 2 são apresentadas algumas opções de seleção de atributos, para logo em seguida no painel 3 mostrar o treino e a validação. Os painéis 4 e 5 mostram respectivamente a aplicação do test set ao modelo e a o ranqueamento das moléculas.

distintas.

6.5. Considerações finais

Esta seção sintetiza os principais conceitos discutidos ao longo do capítulo, destacando o papel crescente do AM na descoberta de fármacos e a importância da integração entre dados experimentais e métodos computacionais.

Ao longo deste capítulo, foram apresentados conceitos fundamentais para compreender como métodos de AM e IA vêm sendo incorporados à descoberta de fármacos. Partindo de fundamentos de bioinformática estrutural, representação molecular, bancos de dados químicos e estruturais, técnicas experimentais de triagem e modelos supervisionados, discutimos como diferentes tipos de dados podem ser transformados em entradas computacionais para a construção de modelos preditivos. Também foram abordadas estratégias clássicas e modernas de SBDD, incluindo *docking* molecular, FEs baseadas em AM e métodos recentes de *co-folding*. Essas técnicas foram discutidas em suas vantagens e limitações, com o objetivo de preparar a comunidade acadêmica para a escolha crítica das metodologias mais adequadas a diferentes desafios em SBDD.

Um ponto central é que o desempenho de modelos computacionais não depende apenas da escolha do algoritmo. A qualidade, a origem, a padronização e a interpretação dos dados são fatores igualmente determinantes. Em descoberta de fármacos, os dados experimentais frequentemente são escassos, heterogêneos, ruidosos ou gerados em condições específicas, o que dificulta a comparação entre métodos e a generalização para novos alvos ou novas classes químicas.

Para enfrentar esse problema, diferentes iniciativas internacionais vêm buscando organizar a geração, padronização e compartilhamento de dados experimentais para

 ©2026 SBC – Soc. Bras. de Computação

SBDD. Uma delas é o projeto **LIGAND-AI**, que propõe disponibilizar de forma aberta no **AIRCHECK.ai** dados de triagens biofísicas, como DEL e ASMS, para uma fração representativa do proteoma humano. A obtenção de dados tabulados, homogêneos e comparáveis entre diferentes proteínas poderá fornecer uma base experimental robusta para treinar e avaliar métodos preditivos capazes de priorizar ligantes e, progressivamente, apoiar estimativas de potência e seletividade de pequenas moléculas. Em longo prazo, esse tipo de iniciativa se alinha à visão do **Target 2035**, que busca identificar moduladores farmacológicos ou sondas químicas para todas as proteínas humanas até 2035, incluindo aquelas ainda pouco exploradas experimentalmente, e finalmente acelerar a descoberta de fármacos.

O estudo de caso com **WDR91** ilustra essa lógica de integração entre dados experimentais disponíveis no **AIRCHECK.ai** e AM no SBDD. A utilização de um subconjunto simplificado de dados DEL, representado por *fingerprints* ECFP4, permite demonstrar de forma prática como um problema de descoberta de ligantes pode ser formulado como uma tarefa supervisionada de classificação e como moléculas disponíveis comercialmente podem ser priorizadas.

Nesse contexto, desafios comunitários, como o da WDR91 no **DREAM Target 2035 Drug Discovery Challenge** e outros no **CACHE**, têm papel complementar, pois transformam conjuntos de dados experimentais em *benchmarks* abertos para a comunidade, permitindo estabelecer o estado da arte, mapear desafios atuais e identificar potenciais soluções. Essas iniciativas podem desempenhar, para a descoberta de fármacos, um papel análogo ao que o **CASP** exerceu para a predição de estruturas de proteínas, ao criar um ambiente competitivo, padronizado e rigoroso que favoreceu o surgimento de métodos como o AlphaFold. Dessa forma, a comunidade científica brasileira encontra-se em um momento estratégico para participar dessas competições e contribuir para avanços relevantes possibilitados pela IA em SBDD. A equipe brasileira RFL_Bambu ilustra esse cenário ao vencer o CACHE Challenge #2, sendo a única equipe do Sul Global na competição.

Por fim, espera-se que os conceitos apresentados neste capítulo forneçam aos leitores e participantes uma visão crítica e aplicada da área. O uso de AM na descoberta de fármacos exige domínio técnico, mas também compreensão das limitações experimentais, dos vieses dos dados, das métricas de validação e do contexto biológico dos alvos estudados. Ao integrar fundamentos teóricos, exemplos práticos e iniciativas colaborativas de ciência aberta, este capítulo busca preparar os participantes para avaliar, desenvolver e aplicar modelos computacionais de forma responsável em problemas reais de descoberta de fármacos, contribuindo para a formação de uma nova geração de *drug hunters*.

6.6. Referências

Referências

[Abramson et al. 2024] Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., et al. (2024). Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630:493–500.

 ©2026 SBC – Soc. Bras. de Computação

[Ackloo et al. 2025] Ackloo, S., Li, F., Szewczyk, M., et al. (2025). A target class ligandability evaluation of wd40 repeat-containing proteins. *Journal of Medicinal Chemistry*, 68(2).

[Ahmad et al. 2023] Ahmad, S., Xu, J., Feng, J., Hutchinson, A., Zeng, H., Ghiabi, P., Dong, A., Centrella, P., Clark, M., Guié, M.-A., et al. (2023). Discovery of a first-in-class small-molecule ligand for wdr91 using dna-encoded chemical library selection followed by machine learning. *Journal of Medicinal Chemistry*, 66(23):16051–16061.

[Allen et al. 2015] Allen, W. J., Balius, T. E., Mukherjee, S., Brozell, S. R., Moustakas, D. T., Lang, P. T., Case, D. A., Kuntz, I. D., and Rizzo, R. C. (2015). Dock 6: impact of new features and current docking performance. *Journal of Computational Chemistry*, 36(15):1132–1156.

[Altschuh et al. 1987] Altschuh, D., Lesk, A. M., Bloomer, A. C., and Klug, A. (1987). Correlation of coordinated amino acid substitutions with function in viruses related to tobacco mosaic virus. *Journal of Molecular Biology*, 193(4):693–707.

[Amaro et al. 2018] Amaro, R. E., Baudry, J., Chodera, J., Demir, O., McCammon, J. A., Miao, Y., and Smith, J. C. (2018). Ensemble docking in drug discovery. *Biophysical journal*, 114(10):2271–2278.

[Arrua et al. 2024] Arrua, O. E., Aderhold, A., Werhli, A. V., and Machado, K. D. S. (2024). Rfl-score: random forest with lasso scoring function for protein-ligand molecular docking. In *2024 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–8. IEEE.

[Ballester and Mitchell 2010] Ballester, P. J. and Mitchell, J. B. (2010). A machine learning approach to predicting protein–ligand binding affinity with applications to molecular docking. *Bioinformatics*, 26(9):1169–1175.

[Berman et al. 2000] Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The protein data bank. *Nucleic acids research*, 28(1):235–242.

[Bjerrum 2017] Bjerrum, E. J. (2017). Smiles enumeration as data augmentation for neural network modeling of molecules. *arXiv preprint arXiv:1703.07076*.

[Branden and Tooze 2012] Branden, C. I. and Tooze, J. (2012). *Introduction to protein structure*. Garland Science.

[Bray et al. 2024] Bray, F., Laversanne, M., Sung, H., Ferlay, J., Siegel, R. L., Soerjomataram, I., and Jemal, A. (2024). Global cancer statistics 2022: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 74(3):229–263.

[Breiman 2001] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

 ©2026 SBC – Soc. Bras. de Computação

[Burley et al. 2021] Burley, S. K., Bhikadiya, C., Bi, C., Bittrich, S., Chen, L., Crichlow, G. V., Christie, C. H., Dalenberg, K., Di Costanzo, L., Duarte, J. M., et al. (2021). Rcsb protein data bank: powerful new tools for exploring 3d structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences. *Nucleic acids research*, 49(D1):D437–D451.

[CACHE Initiative 2024] CACHE Initiative (2024). Cache challenge #2 results. Accessed March 2026.

[Carhart et al. 1985] Carhart, R. E., Smith, D. H., and Venkataraghavan, R. (1985). Atom pairs as molecular features in structure-activity studies: definition and applications. *Journal of Chemical Information and Computer Sciences*, 25(2):64–73.

[Cereto-Massagué et al. 2015] Cereto-Massagué, A., Ojeda, M. J., Valls, C., Mulero, M., Garcia-Vallvé, S., and Pujadas, G. (2015). Molecular fingerprint similarity search in virtual screening. *Methods*, 71:58–63.

[Chai Discovery Team 2024] Chai Discovery Team (2024). Chai-1: Decoding the molecular interactions of life. *bioRxiv*. Preprint.

[Chen and Guestrin 2016] Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *KDD*, pages 785–794.

[Chithrananda et al. 2020] Chithrananda, S., Grand, G., and Ramsundar, B. (2020). Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*.

[Copeland et al. 2006] Copeland, R. A., Pompliano, D. L., and Meek, T. D. (2006). Drug–target residence time and its implications for lead optimization. *Nature Reviews Drug Discovery*, 5(9):730–739.

[Crampon et al. 2022] Crampon, K., Giorkallos, A., Deldossi, N., Baud, S., and Steffenel, L. A. (2022). Machine-learning methods for ligand–protein molecular docking. *Drug Discovery Today*, 27(1):151–164.

[Dalby et al. 1992] Dalby, A., Nourse, J. G., Hounshell, W. D., Gushurst, A. K., Grier, D. L., Leland, B. A., and Laufer, J. (1992). Description of several chemical structure file formats used by computer programs developed at molecular design limited. *Journal of chemical information and computer sciences*, 32(3):244–255.

[Daylight Chemical Information Systems 2024] Daylight Chemical Information Systems (2024). Daylight theory manual: Fingerprints. Accessed: 2026-05-11.

[Durant et al. 2002] Durant, J. L., Leland, B. A., Henry, D. R., and Nourse, J. G. (2002). Reoptimization of mdl keys for use in drug discovery. *Journal of Chemical Information and Computer Sciences*, 42(6):1273–1280.

[Edwards et al. 2025] Edwards, A. M. et al. (2025). Protein–ligand data at scale to support machine learning. *Nature Reviews Chemistry*, 9:634–645.

 ©2026 SBC – Soc. Bras. de Computação

[Ferrari et al. 2004] Ferrari, A. M., Wei, B. Q., Costantino, L., and Shoichet, B. K. (2004). Soft docking and multiple receptor conformations in virtual screening. *Journal of medicinal chemistry*, 47(21):5076–5084.

[Ferreira et al. 2015] Ferreira, L. G., dos Santos, R. N., Oliva, G., and Andricopulo, A. D. (2015). Molecular docking and structure-based drug design strategies. *Molecules*, 20(7):13384–13421.

[Franzini et al. 2014] Franzini, R. M., Neri, D., and Scheuermann, J. (2014). Dna-encoded chemical libraries: Advancing beyond conventional small-molecule libraries. *Accounts of Chemical Research*, 47(4):1247–1255.

[Friedman 2001] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

[Ganesan et al. 2017] Ganesan, A., Coote, M. L., and Barakat, K. (2017). Molecular dynamics-driven drug discovery: leaping forward with confidence. *Drug discovery today*, 22(2):249–269.

[Gilson et al. 2015] Gilson, M. K., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., and Chong, J. (2015). Bindingdb in 2015: A public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic Acids Research*, 44(D1):D1045–D1053.

[Gironda-Martínez et al. 2021] Gironda-Martínez, A., Donckele, E. J., Samain, F., and Neri, D. (2021). Dna-encoded chemical libraries: A comprehensive review with succession stories and future challenges. *ACS Pharmacology & Translational Science*, 4(4):1265–1279.

[Gómez-Bombarelli et al. 2018] Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. (2018). Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276.

[Goodnow et al. 2016] Goodnow, R. A., Dumelin, C. E., and Keefe, A. D. (2016). Dna-encoded chemistry: enabling the deeper sampling of chemical space. *Nature Reviews Drug Discovery*, 16(2):131–147.

[Guyon and Elisseeff 2003] Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

[Halperin et al. 2002] Halperin, I., Ma, B., Wolfson, H., and Nussinov, R. (2002). Principles of docking: An overview of search algorithms and a guide to scoring functions. *Proteins: Structure, Function, and Bioinformatics*, 47(4):409–443.

[Hastie 2009] Hastie, T. (2009). The elements of statistical learning: data mining, inference, and prediction.

 ©2026 SBC – Soc. Bras. de Computação

[Herasymenko et al. 2025] Herasymenko, O. et al. (2025). Cache challenge #2: Targeting sars-cov-2 nsp13. *Journal of Chemical Information and Modeling*.

[Homola 2008] Homola, J. (2008). Surface plasmon resonance sensors for detection of chemical and biological species. *Chemical Reviews*, 108(2):462–493.

[Jiang et al. 2022] Jiang, H., Wang, J., Cong, W., Huang, Y., Ramezani, M., Sarma, A., Dokholyan, N. V., Mahdavi, M., and Kandemir, M. T. (2022). Predicting protein–ligand docking structure with graph neural network. *Journal of chemical information and modeling*, 62(12):2923–2932.

[Jiménez et al. 2018] Jiménez, J., Škalič, M., Martínez-Rosell, G., and De Fabritiis, G. (2018). k_{deep}: Protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. *Journal of Chemical Information and Modeling*, 58(2):287–296.

[Jumper et al. 2021] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with alphafold. *Nature*, 596:583–589.

[Kadukova et al. 2021] Kadukova, M., Machado, K. d. S., Chacón, P., and Grudinin, S. (2021). Korp-pl: a coarse-grained knowledge-based scoring function for protein–ligand interactions. *Bioinformatics*, 37(7):943–950.

[Ke et al. 2017] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

[Kernohan and Boycott 2024] Kernohan, K. D. and Boycott, K. M. (2024). The expanding diagnostic toolbox for rare genetic diseases. *Nature Reviews Genetics*, 25(6):401–415.

[Krishna et al. 2024] Krishna, R., Wang, J., Ahern, W., Sturmfels, P., Venkatesh, P., Kalvet, I., Lee, G. R., Morey-Burrows, F. S., Anishchenko, I., Humphreys, I. R., et al. (2024). Generalized biomolecular modeling and design with rosettafold all-atom. *Science*, 384:eadl2528.

[Kryshtafovych et al. 2021] Kryshtafovych, A., Schwede, T., Topf, M., Fidelis, K., and Moult, J. (2021). Critical assessment of methods of protein structure prediction (casp)—round xiv. *Proteins: Structure, Function, and Bioinformatics*, 89(12):1607–1617.

[Kuntz 1992] Kuntz, I. D. (1992). Structure-based strategies for drug design. *Science*, 257:1078–1082.

[Landrum 2006] Landrum, G. (2006). Rdkit: Open-source cheminformatics. 2006. *Google Scholar*.

[Landrum 2013] Landrum, G. (2013). Rdkit: Open-source cheminformatics software. Accessed: 2026-05-11.

 ©2026 SBC – Soc. Bras. de Computação

[Lengauer and Rarey 1996] Lengauer, T. and Rarey, M. (1996). Computational methods for biomolecular docking. *Current Opinion in Structural Biology*, 6(3):402–406.

[Li et al. 2024] Li, F., Ackloo, S., Arrowsmith, C. H., Ban, F., Barden, C. J., Beck, H., Beránek, J., Berenger, F., Bolotokova, A., Bret, G., et al. (2024). Cache challenge# 1: targeting the wdr domain of lrrk2, a parkinson's disease associated protein. *Journal of Chemical Information and Modeling*, 64(22):8521–8536.

[Li et al. 2019] Li, J., Fu, A., and Zhang, L. (2019). An overview of scoring functions used for protein–ligand interactions in molecular docking. *Interdisciplinary Sciences: Computational Life Sciences*, 11(2):320–328.

[Lima et al. 2016] Lima, T. A., Bezerra, I. C., Rocha, B. C. G., Viana, J. R., and Silva, C. R. (2016). Use of docking approaches to predict the affinity and orientation between molecules: a review. *Biophysical Reviews*, 8(2):157–165.

[Liu et al. 2007] Liu, T., Lin, Y., Wen, X., Jorissen, R. N., and Gilson, M. K. (2007). Bindingdb: a web-accessible database of experimentally determined protein-ligand binding affinities. *Nucleic Acids Research*, 35(Database):D198–D201.

[Liu et al. 2017] Liu, Z., Su, M., Han, L., Liu, J., Yang, Q., Li, Y., and Wang, R. (2017). Forging the basis for developing protein–ligand interaction scoring functions. *Accounts of chemical research*, 50(2):302–309.

[Lybrand 1995] Lybrand, T. P. (1995). Ligand–protein docking. *Current Opinion in Structural Biology*, 5(2):224–228.

[Machado et al. 2010] Machado, K. S., Winck, A. T., Ruiz, D. D., and de Souza, O. N. (2010). Mining flexible-receptor docking experiments to select promising protein receptor snapshots. *BMC genomics*, 11(5):1–13.

[Masters et al. 2023] Masters, M. R., Mahmoud, A. H., Wei, Y., and Lill, M. A. (2023). Deep learning model for efficient protein–ligand docking with implicit side-chain flexibility. *Journal of Chemical Information and Modeling*, 63(6):1695–1707.

[McCammon and Harvey 1987] McCammon, J. A. and Harvey, S. C. (1987). *Dynamics of Proteins and Nucleic Acids*. Cambridge University Press.

[McNutt et al. 2025] McNutt, A. T., Li, Y., Meli, R., Aggarwal, R., and Koes, D. R. (2025). Gnina 1.3: the next increment in molecular docking with deep learning. *Journal of Cheminformatics*, 17(1):28.

[Meli et al. 2022] Meli, R., Morris, G. M., and Biggin, P. C. (2022). Scoring functions for protein-ligand binding affinity prediction using structure-based deep learning: a review. *Frontiers in bioinformatics*, 2:885983.

[Meng et al. 2011] Meng, X. et al. (2011). Molecular docking: a powerful approach. *Current Computer-Aided Drug Design*, 7(2):146–157.

 ©2026 SBC – Soc. Bras. de Computação

[Morris et al. 2009] Morris, G. M., Huey, R., Lindstrom, W., Sanner, M. F., Belew, R. K., Goodsell, D. S., and Olson, A. J. (2009). Autodock4 and autodocktools4: Automated docking with selective receptor flexibility. *Journal of computational chemistry*, 30(16):2785–2791.

[Muchiri and van Breemen 2020] Muchiri, R. N. and van Breemen, R. B. (2020). Affinity selection–mass spectrometry for the discovery of pharmacologically active compounds from combinatorial libraries and natural products. *Journal of Mass Spectrometry*, 56(5).

[Murphy 2012] Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.

[Murray et al. 2022] Murray, C. J. L., Ikuta, K. S., Sharara, F., Swetschinski, L., Robles Aguilar, J., Gray, A., Han, C., Bisignano, C., Rao, P., et al. (2022). Global burden of bacterial antimicrobial resistance in 2019: a systematic analysis. *The Lancet*, 399(10325):629–655.

[Nilakantan et al. 1987] Nilakantan, R., Bauman, N., Dixon, J. S., and Venkataraghavan, R. (1987). Topological torsion: a new molecular descriptor for sar applications. comparison with other descriptors. *Journal of Chemical Information and Computer Sciences*, 27(2):82–85.

[Pagadala et al. 2017] Pagadala, N. S., Syed, K., and Tuszynski, J. (2017). Software for molecular docking: a review. *Biophysical Reviews*, 9(2):91–102.

[Passaro et al. 2025] Passaro, S., Corso, G., Wohlwend, J., Reveiz, M., Thaler, S., Somnath, V. R., Getz, N., Portnoi, T., Roy, J., Stark, H., et al. (2025). Boltz-2: Towards accurate and efficient binding affinity prediction. *bioRxiv*. Preprint.

[Petsko and Ringe 2004] Petsko, G. A. and Ringe, D. (2004). *Protein Structure and Function*. Primers in Biology. New Science Press; distributed by Oxford University Press.

[Pinzi and Rastelli 2019] Pinzi, L. and Rastelli, G. (2019). Molecular docking: shifting paradigms in drug discovery. *International Journal of Molecular Sciences*, 20(18):4331.

[Prudent et al. 2023] Prudent, R., Lemoine, H., Walsh, J., and Roche, D. (2023). Affinity selection mass spectrometry speeding drug discovery. *Drug Discovery Today*, 28(11):103760.

[Rich and Myszka 2008] Rich, R. L. and Myszka, D. G. (2008). Survey of the year 2007 commercial optical biosensor literature. *Journal of Molecular Recognition*, 21(6):355–400.

[Rogers and Hahn 2010] Rogers, D. and Hahn, M. (2010). Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, 50(5):742–754. PMID: 20426451.

 ©2026 SBC – Soc. Bras. de Computação

[Scaini et al. 2019] Scaini, J. L. R., Camargo, A. D., Seus, V. R., von Groll, A., Werhli, A. V., da Silva, P. E. A., and dos Santos Machado, K. (2019). Molecular modelling and competitive inhibition of a mycobacterium tuberculosis multidrug-resistance efflux pump. *Journal of Molecular Graphics and Modelling*, 87:98–108.

[Scantlebury et al. 2020] Scantlebury, J. et al. (2020). Data set augmentation allows deep learning-based virtual screening to better generalize to unseen target classes. *Journal of Chemical Information and Modeling*, 60(8):3722–3730.

[Schwaller et al. 2019] Schwaller, P., Laino, T., Gaudin, T., Bolgar, P., Hunter, C. A., Bekas, C., and Lee, A. A. (2019). Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science*, 5(9):1572–1583.

[Senior et al. 2020] Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W. R., Bridgland, A., et al. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, 577:706–710.

[Shen et al. 2020] Shen, C., Ding, J., Wang, Z., Cao, D., Ding, X., and Hou, T. (2020). From machine learning to deep learning: Advances in scoring functions for protein–ligand docking. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 10(1):e1429.

[Škrinjar et al. 2026] Škrinjar, P., Eberhardt, J., Studer, G., et al. (2026). Evaluating generalization in protein–ligand cofolding methods. *Nature Structural & Molecular Biology*.

[Stepniewska-Dziubinska et al. 2018] Stepniewska-Dziubinska, M. M., Zielenkiewicz, P., and Siedlecki, P. (2018). Development and evaluation of a deep learning model for protein–ligand binding affinity prediction. *Bioinformatics*, 34(21):3666–3674.

[Su et al. 2018] Su, M., Yang, Q., Du, Y., Feng, G., Liu, Z., Li, Y., and Wang, R. (2018). Comparative assessment of scoring functions: the casf-2016 update. *Journal of chemical information and modeling*, 59(2):895–913.

[Svetnik et al. 2003] Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., and Feuston, B. P. (2003). Random forest: a classification and regression tool for compound classification and qsar modeling. *Journal of chemical information and computer sciences*, 43(6):1947–1958.

[Tripos 2005] Tripos, S. (2005). Tripos mol2 file format.

[Trott and Olson 2010] Trott, O. and Olson, A. J. (2010). Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of Computational Chemistry*, 31(2):455–461.

[Varadi et al. 2022] Varadi, M., Anyango, S., Deshpande, M., Nair, S., Natassia, C., Yordanova, G., Yuan, D., Stroe, O., Wood, G., Laydon, A., et al. (2022). Alphafold protein structure database: Massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Research*, 50(D1):D439–D444.

 ©2026 SBC – Soc. Bras. de Computação

[Vaswani et al. 2017] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

[Wallach and Heifets 2018] Wallach, I. and Heifets, A. (2018). Most ligand-based classification benchmarks reward memorization rather than generalization. *Journal of chemical information and modeling*, 58(5):916–932.

[Wang et al. 2004] Wang, R., Fang, X., Lu, Y., and Wang, S. (2004). The pdbbind database: Collection of binding affinities for protein-ligand complexes with known three-dimensional structures. *Journal of Medicinal Chemistry*, 47(12):2977–2980.

[Wang et al. 2025] Wang, X. et al. (2025). Enantioselective protein affinity selection mass spectrometry (e-asms). *Nature Communications*, 17(651).

[Weininger 1988] Weininger, D. (1988). Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36.

[Werhli et al. 2025] Werhli, A. V., Lopes, P. P., Arrua, O. E., Aderhold, A., and Machado, K. d. S. (2025). Crrf-score-cumulative ranking random forest scoring function for free energy of binding prediction in protein-ligand docking. In *Brazilian Conference on Intelligent Systems*, pages 199–213. Springer.

[Westbrook and Fitzgerald 2003] Westbrook, J. D. and Fitzgerald, P. (2003). The pdb format, mmcif, and other data formats. *Methods Biochem Anal*, 44:161–179.

[Xu et al. 2026] Xu, S., Feng, Q., Qiao, L., Wu, H., Shen, T., Cheng, Y., Zheng, S., and Sun, S. (2026). Benchmarking all-atom biomolecular structure prediction with foldbench. *Nature Communications*, 17:442.

[Xu 2024] Xu, W. (2024). Current status of computational approaches for small molecule drug discovery. *Journal of Medicinal Chemistry*, 67(21):18633–18636.

[Xu et al. 2025] Xu, Z. et al. (2025). A generative ai-discovered tnik inhibitor for idiopathic pulmonary fibrosis. *Nature Medicine*, 31:2602–2610.

[Zheng et al. 2025] Zheng, Y., Koh, H. Y., Ju, J., Yang, M., May, L. T., Webb, G. I., Li, L., Pan, S., and Church, G. (2025). Large language models for drug discovery and development. *Patterns*, 6(10).

 ©2026 SBC – Soc. Bras. de Computação

SBCAS2026

REALIZAÇÃO

SBC Sociedade Brasileira de Computação

ORGANIZAÇÃO

decom departamento de computação

UFOP Universidade Federal de Ouro Preto

PATROCÍNIO

CNPq

CAPES

FAPEMIG

INSTITUIÇÕES COLABORADORAS

UFG Universidade Federal de Goiás

PUC Minas

UFPI

UFMG Universidade Federal de Minas Gerais

ICMC USP São Carlos

INF Instituto de Informática UFRGS

CPS Centro Paula Souza SÃO PAULO GOVERNO DO ESTADO

UNISINOS Nossa sala de aula é o mundo.

IN ALTUM UESC

UFF Universidade Federal Fluminense

UFPR Universidade Federal do Paraná

APOIO

BRASIL BEM CUIDADO MAIS SAÚDE PARA QUEM MAIS PRECISA

SUS

MINISTÉRIO DA SAÚDE

GOVERNO FEDERAL BRASIL UNIÃO E RECONSTRUÇÃO

SITE