

JAI 2022

41ª Jornada de Atualização em Informática

Organizadoras

Eliana Silva de Almeida

Flávia Maria Santoro

42°CSBC

CONGRESSO DA SOCIEDADE
BRASILEIRA DE COMPUTAÇÃO



Organizadoras

Eliana Silva de Almeida

Flávia Maria Santoro

**41^a Jornada de Atualização
em Informática
JAI 2022**

Porto Alegre

Sociedade Brasileira de Computação –SBC

2022

Dados Internacionais de Catalogação na Publicação (CIP)

C749 Congresso da Sociedade Brasileira de Computação (42. : 31 jul.-
5 ago. 2022 : Niterói)

41ª Jornada de Atualização em Informática (JAI 2022)
[recurso eletrônico] / organização: Eliana Silva de Almeida ;
Flávia Maria Santoro. Dados eletrônicos. – Porto Alegre :
Sociedade Brasileira de Computação, 2022.

199 p. ; il. : PDF ; 15,3MB

Inclui bibliografia

ISBN 978-85-7669-499-1 (e-book)

1. Computação – Brasil – Congressos. 2. Informática. I.
Almeida, Eliana Silva de. II. Santoro, Flávia Maria. III.
Universidade Federal Fluminense. IV. Sociedade Brasileira de
Computação. V. Título.

CDU 004(063)

Ficha catalográfica elaborada por Jéssica Paola Macedo Müller – CRB-10/2662

Biblioteca Digital da SBC – SBC OpenLib

Índice para catálogo sistemático:

1. Ciência e tecnologia informáticas : Computação : Processamento de dados –
Publicação de conferências, congressos, simpósios etc... 004(063)

41ª Jornada de Atualização em Informática (JAI2022)

Editora

Sociedade Brasileira de Computação (SBC)

Coordenação Geral

Eliana Silva de Almeida – Universidade Federal de Alagoas

Flávia Maria Santoro – Univ. do Estado do Rio de Janeiro e Instituto de Tecnologia e Liderança

Coordenação Local

Célio Albuquerque – Universidade Federal Fluminense

Comitê de Programa

Aline Andrade – Universidade Federal da Bahia

Alirio Sá – Universidade Federal da Bahia

Bruno Lopes – Universidade Federal Fluminense

Claudia Cappelli – Universidade do Estado do Rio de Janeiro

Claudia Motta – Universidade Federal do Rio de Janeiro

Daniel Batista – Universidade do Estado de São Paulo

Daniel Avila Vecchiato – Universidade Federal do Mato Grosso

Fabiola Guerra Nakamura – Universidade Federal do Amazonas

Flávio Assis – Universidade Federal da Bahia

Márcio Castro – Universidade Federal de Santa Catarina

Márcio Cornélio – Universidade Federal de Pernambuco

Rita Suzana Pitangueira Maciel – Universidade Federal da Bahia

Thais Vasconcelos Batista – Universidade Federal do Rio Grande do Norte

Vera Werneck – Universidade do Estado do Rio de Janeiro

A 41ª Jornada de Atualização em Informática (JAI) foi realizada como parte do 42º Congresso a Sociedade Brasileira de Computação

Realização

Sociedade Brasileira de Computação (SBC)

Organização

Universidade Federal Fluminense

Coordenação Geral

Flavia Bernardini

José Viterbo

Comitê-Executivo

Aline Paes (Programação Científica)

José Raphael Bokehi (Infraestrutura)

Raquel Bravo (Operacional)

Simone Martins (Financeiro)

Vânia Neves (Comunicações)



Apresentação

A Jornada de Atualização em Informática (JAI) é um evento que se tornou referência na apresentação de tópicos relevantes para a pesquisa e desenvolvimento dentro do Congresso da Sociedade Brasileira de computação. A JAI vem contribuindo de forma significativa com a disseminação de conhecimento de ponta para os alunos, profissionais e pesquisadores em Computação no Brasil. Em sua 41ª edição, são tratados os temas processamento de linguagem natural e aprendizagem profunda; estratégias ágeis em projetos de PD&I; big data e economia circular; e blockchain.

O Capítulo 1 (Processamento de Linguagem Natural via Aprendizagem Profunda) apresenta como técnicas de Aprendizagem Profunda podem ser utilizadas na resolução de tarefas de Processamento de Linguagem Natural (PLN), tais como como Classificação e Sumarização de Sentenças, visando o benefício do poder computacional disponível atualmente e da baixa necessidade de engenharia de fatures na utilização destes modelos.

O Capítulo 2 (Estratégias Ágeis Aplicadas à Projetos de PD&I: Da Teoria à Prática) discute os principais desafios, conceitos, relatos e estudos de caso da adoção de estratégias ágeis, em particular o framework Scrum, em pesquisa científica voltada para produção de inovação tecnológica.

O Capítulo 3 (Big Data e Tecnologias Digitais Aplicadas à Economia Circular: Oportunidades para Cadeias Produtivas mais Sustentáveis) mostra como big data e as tecnologias digitais, tais como internet das coisas, computação em nuvem e blockchain, desempenham um papel chave na economia circular, como atendem às necessidades desses modelos de negócio no contexto, contribuindo para os desafios atuais de sustentabilidade.

O Capítulo 4 (Visitando na teoria e na prática o Cartesi RollUps: para além das limitações da Blockchain, uma solução de futuro para aplicativos descentralizados) trata de uma tecnologia promissora- o Cartesi Rollups – que é capaz de alçar Blockchain ao status de padrão das aplicações na internet do futuro, minimizando suas limitações de escalabilidade.

Agradecemos ao comitê de programa e a todos os autores que submeteram propostas. Tivemos cinco propostas submetidas e selecionamos quatro para serem desenvolvidas como capítulos neste livro. Agradecemos também aos organizadores do CSBC 2022 e toda a equipe de organização pelo apoio na elaboração deste evento. Um agradecimento especial ao coordenador local da JAI, prof. Celio de Albuquerque Vinicius Neves.

Ficamos muito felizes com o resultado e esperamos que todos aproveitem bastante!

Eliana Almeida (UFAL) e Flavia Maria Santoro (Inteli e UERJ)

Coordenadoras da JAI 2022

Índice

Processamento de Linguagem Natural via Aprendizagem Profunda 8

*Bárbara Stéphanie Neves Oliveira, Luís Gustavo C. do Rêgo, Lucas Peres,
Ticiane L. Coelho da Silva e José Antônio F. de Macêdo*

Estratégias ágeis aplicadas à projetos de PD&I: da teoria à prática 55

Andre L L Aquino, Givanildo Nascimento-Jr. e Fabiane Queiroz

Big Data e Tecnologias Digitais Aplicadas à Economia Circular:
Oportunidades para Cadeias Produtivas mais Sustentáveis..... 106

*Rosangela de Fátima Pereira Marquesone, Francisco Pereira Junior e
Tereza Cristina Melo de Brito Carvalho*

Visitando na teoria e na prática o Cartesi RollUps: para além das
limitações da Blockchain, uma solução de futuro para aplicativos
descentralizados 151

*Antonio A. de A. Rocha, Arthur A. Vianna, Bruno T. Gondim,
Eduardo B. Loivos e Rayan G. O. J. Lima*

Capítulo

1

Processamento de Linguagem Natural via Aprendizagem Profunda

Bárbara Stéphanie Neves Oliveira, Luís Gustavo C. do Rêgo, Lucas Peres, Ticiania L. Coelho da Silva, José Antônio F. de Macêdo

Abstract

Humans need to communicate. Out of this basic need combined with the Web, a vast amount of text has been generated on a daily basis. Given the presence of a lot of information allocated in different resources, it becomes vital to enable machines to understand spoken and written texts. This chapter presents how Deep Learning techniques can solve Natural Language Processing (NLP) tasks (e.g., Text Classification and Sentence Summarization), aiming to benefit from the computational power currently available and the low need for feature engineering when using these models. Initially, some essential concepts about NLP and Deep Learning are presented. Then, different pre-processing and textual representation techniques are explained to be used as input in Deep Learning models. Finally, it is shown how to apply the knowledge acquired in real applications of NLP.

Resumo

Seres humanos precisam de comunicação. A partir da junção dessa necessidade básica com a Web, uma grande quantidade de texto tem sido gerada diariamente. Dada a presença de muitas informações alocadas em diferentes meios, torna-se vital permitir que máquinas compreendam textos falados e escritos. Este capítulo apresenta como técnicas de Aprendizagem Profunda podem ser utilizadas na resolução de tarefas de Processamento de Linguagem Natural (PLN), como Classificação e Sumarização de Sentenças, visando o benefício do poder computacional disponível atualmente e da baixa necessidade de engenharia de features na utilização destes modelos. Inicialmente, são apresentados alguns conceitos importantes sobre PLN e Aprendizagem Profunda. Em seguida, diferentes técnicas de pré-processamento e representação textuais são explicadas a fim de serem usadas como entrada em modelos de Aprendizagem Profunda. Por fim, é mostrado como aplicar os conhecimentos adquiridos em aplicações reais do PLN.

1.1. Introdução

A comunicação, como uma necessidade básica da condição humana, juntamente com a existência da *Web* permitem que uma vasta quantidade de textos escritos e falados seja gerada diariamente. Dado o conteúdo textual presente em mídias sociais, aplicativos de bate-papo, *e-mails*, análises de produtos, artigos de notícias, trabalhos de pesquisa e *ebooks*, tornou-se vital a existência de um processamento automático de textos a fim de oferecer assistência ou tomar decisões para diversas tarefas diárias.

A capacidade de entender textos ou áudios em linguagem natural por uma máquina é um problema que vem sendo investigado há muito tempo [Chollet 2021]. As primeiras tentativas de construção de sistemas de Processamento de Linguagem Natural (PLN, de *Natural Language Processing* ou NLP) foram feitas através da análise intrínseca de linguagens que são naturalmente moldadas por um processo de evolução (por isso o termo “natural”). O PLN moderno envolve não apenas a habilidade de entendimento de uma linguagem como também possibilita, de forma automática, a extração de informações por meio de tarefas, tais como Classificação de Textos, Reconhecimento de Entidades Nomeadas (NER, de *Named Entity Recognition*), Desambiguação do Sentido das Palavras, e *Part-of-Speech* (POS) *tagging*.

Modelos de Aprendizagem Profunda, do inglês *Deep Learning* (também conhecida como Aprendizado Profundo ou Redes Neurais Profundas), aprendem vários níveis de representação de complexidade/abstração dos dados de forma crescente. Vários fatores evidenciam porque esses modelos têm sido amplamente usados em tarefas de PLN: (i) exigem pouca engenharia de *features*; (ii) produzem representações vetoriais que capturam similaridades de unidades linguísticas (palavras, por exemplo) presentes em textos, permitindo que sistemas de PLN possuam uma espécie de dependência de conhecimento; (iii) permitem aprendizado não supervisionado ou semi-supervisionado, o que é importante quando se tem um grande volume de dados e nenhum rótulo; (iv) aprendem vários níveis de representação, possibilitando que o nível mais baixo geralmente possa ser compartilhado entre diferentes tarefas; e (v) naturalmente lidam com a recursividade da linguagem humana, sendo capazes de capturar informações de forma sequencial.

O uso de modelos de Aprendizagem Profunda no PLN iniciou-se com a investigação da capacidade de compreensão de linguagem por Redes Neurais Recorrentes (RNNs, do inglês *Recurrent Neural Networks*) e redes LSTM (*Longest Shortest Term Memory*) [Hochreiter and Schmidhuber 1997]. Essas duas arquiteturas dominaram o PLN de forma geral de 2015 a 2017, uma vez que processam textos de comprimento variável. Os modelos LSTM bidirecionais, em particular, definiram o estado da arte em muitas tarefas importantes, desde Sumarização de Textos até Tradução Automática. Contudo, por volta de 2017 e 2018, uma nova arquitetura surgiu para “substituir” as RNNs: o *Transformer* [Vaswani et al. 2017], que permitiu um progresso considerável do PLN em um curto período de tempo.

O objetivo deste capítulo não é expor todas essas aplicações e arquiteturas de forma abrangente. Em vez disso, o foco está em como aplicar de forma prática representações textuais existentes e obtidas através de técnicas de Aprendizagem Profunda para resolver problemas de PLN. Este capítulo aborda ainda as diferentes etapas de processamento de textos realizada antes de treinar uma rede neural para uma tarefa de PLN. Ao

longo do capítulo, o leitor irá se familiarizar com as diferentes representações textuais que existem na literatura, desde a forma mais tradicional por *bag-of-words*, até representações via *word embeddings*. Apresentadas as principais representações textuais, este capítulo explorará duas tarefas de PLN populares e representativas: Classificação de Textos (ou Análise de Sentimentos) e Sumarização de Sentenças, que basicamente classificam conteúdos de textos e abreviam certos tipos de frases, respectivamente.

Em suma, este capítulo está organizado da seguinte forma: a Seção 1.2 apresenta os conceitos preliminares para entendimento deste capítulo. A Seção 1.3 fornece um roteiro de como alimentar uma rede neural com textos para treinamento de uma tarefa de PNL. A Seção 1.4 introduz as várias formas de representação de textos em vetores numéricos presentes na literatura. A Seção 1.5 discute como construir uma boa representação textual em tarefas de PLN. A Seção 1.6 exemplifica, por meio de tarefas de PLN, o uso de redes neurais profundas como mecanismo de solução. Por fim, as Seções 1.7 e 1.8 apresentam alguns desafios de pesquisa e as considerações finais, respectivamente.

1.1.1. Tecnologias Utilizadas

Ao longo deste curso, muitas tecnologias foram empregadas no desenvolvimento dos códigos relacionados a cada um dos conceitos apresentados e das tarefas de PLN utilizadas como exemplos.

A linguagem *Python* na versão 3.8 foi utilizada em *notebooks Colaboratory* ou, apenas, *Colab*¹. *Notebooks* são páginas *Web* que permitem a escrita e execução de código em um mesmo documento.

Para auxiliar no desenvolvimento do código, as principais bibliotecas foram utilizadas:

- ***Numpy***² adiciona suporte para trabalhar, de forma eficiente, com vetores e matrizes numéricas, bem como funções otimizadas para essas estruturas.
- ***Keras***³ provê interfaces para a criação, treinamento e avaliação das principais estruturas de redes neurais artificiais. Foi construída utilizando como base a biblioteca *TensorFlow*⁴.
- ***Scikit-learn***⁵ contém os principais algoritmos para diferentes tarefas de Aprendizagem de Máquina, como Classificação e Regressão.
- ***Matplotlib***⁶ e ***Seaborn***⁷, utilizadas para construção de gráficos.

A biblioteca *spaCy*⁸ e *NLTK*⁹ também foram utilizadas, bem como o *framework*

¹<https://colab.research.google.com>

²<https://numpy.org/>

³<https://keras.io/>

⁴<https://www.tensorflow.org/>

⁵<https://scikit-learn.org/stable/>

⁶<https://matplotlib.org/>

⁷<https://seaborn.pydata.org/>

⁸<https://spacy.io/>

⁹<https://www.nltk.org/>

*flairNLP*¹⁰, para trechos de códigos que envolvem o uso de modelos de PLN mais modernos, além de outras bibliotecas tradicionais da linguagem *Python*, como `string` e `unicode`.

1.2. Preliminares

A presente seção apresenta um histórico de como diferentes arquiteturas de Aprendizagem Profunda se relacionam com problemas de PLN. Além disso, são descritos alguns conceitos básicos sobre redes neurais e algumas das arquiteturas mais úteis no contexto do Processamento de Linguagem Natural.

1.2.1. Breve Histórico do Uso de Aprendizagem Profunda no PLN

A linguagem natural é complexa, ambígua, e está em constante mudança. Se para um ser humano é preciso de tempo e estudo para entender o sistema de princípios, condições e regras que formam os elementos ou propriedades de todas as línguas, criar algoritmos que possam dar sentido à linguagem é uma das, se não a mais inerente e importante, frente do PLN.

Além disso, não existe um idioma universal. A linguagem também não é flexível à formalização, isto é, não é trivial criar um conjunto complexo de regras que defina inteiramente o Português, o Inglês, o Japonês, ou esses três idiomas juntos, por exemplo. Após várias décadas de esforço para a criação de regras manuais para certas línguas [Chollet 2021], hoje, o PLN usa técnicas de Aprendizagem Profunda e grandes conjuntos de dados para que seja possível não só entender uma linguagem, como também solucionar diferentes problemas, como:

- **Classificação de Texto.** *"Qual é o tema deste texto?", "Esta notícia é verdadeira ou falsa?"*
- **Reconhecimento de Entidade Nomeada.** *"Como extrair o nome de todas as pessoas e lugares presentes nesta página?"*
- **Language Modeling.** *"Qual deve ser a próxima palavra nesta frase incompleta?"*
- **Sumarização.** *"Como você resumiria este artigo em um parágrafo?"*

A Figura 1.1 resume a evolução da área de Processamento de Linguagem Natural durante a era da Aprendizagem Profunda. A partir dos anos 2000, redes neurais começaram a ser utilizadas para a tarefa de *Language Modeling*, que visa prever a próxima palavra em uma sentença tomando como base as palavras anteriores [Bengio et al. 2000b]. Já na última década, pesquisas em PLN tiveram avanços consideráveis principalmente a partir da introdução do algoritmo *Word2Vec* [Mikolov et al. 2013]. O modelo trouxe uma nova forma de representar relações e similaridades entre termos através de vetores de palavras, ao demonstrar que palavras similares estariam próximas umas às outras em um mesmo espaço vetorial (ou espaço/plano Euclidiano), de dimensão finita. Apesar de existirem modelos mais recentes, o *Word2Vec* ainda é uma escolha popular e amplamente utilizada hoje.

¹⁰<https://github.com/flairNLP/flair>

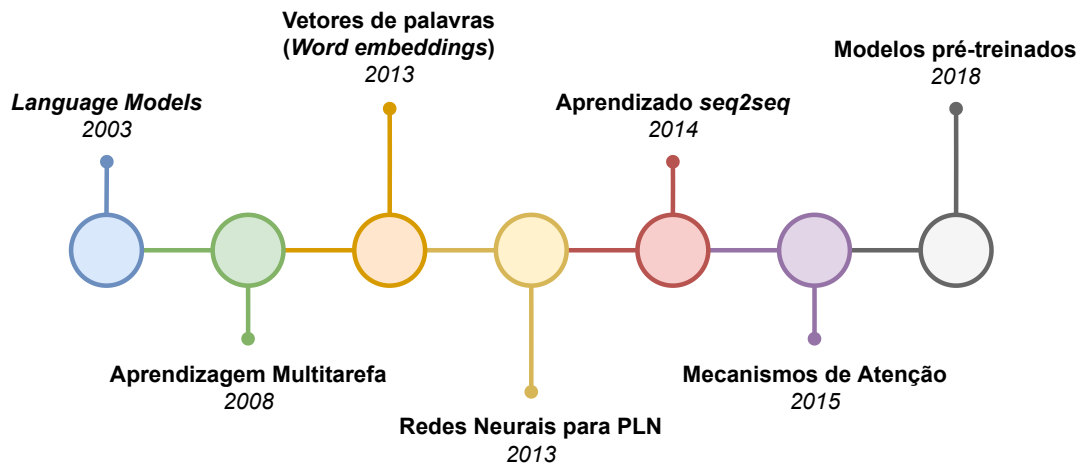


Figura 1.1. Avanços na área de PLN e Aprendizagem Profunda ao longo do tempo. Adaptado de [Louis 2020].

Durante o apogeu do *Word2Vec*, também em 2013, ocorreu o marco da adoção de modelos de redes neurais no PLN, em particular dois tipos bem definidos: Redes Neurais Recorrentes (RNNs, de *Recurrent Neural Networks*) e Redes Neurais Convolucionais (*Convolutional Neural Networks* ou CNNs). Graças a sua arquitetura, RNNs tornaram-se renomadas por lidar com sequências de entrada (como frases e parágrafos) de forma dinâmica e por possuir uma característica de recorrência, que permite representar a ordem das palavras e seu efeito em palavras anteriores e posteriores. Ao mesmo tempo, as redes CNNs que estavam começando a ser amplamente adotadas pela comunidade de Visão Computacional, também começaram a ser aplicadas em PLN. A vantagem de usar CNNs para lidar com sequências de texto é que elas são mais paralelizáveis do que RNNs.

Uma outra desvantagem das RNNs abre caminho para o problema da perda de informações através do cálculo dos gradientes. Como essas redes são muito profundas, os erros retro-propagados podem ser, potencialmente, números cada vez menores, resultando em um longo processo de treinamento ou simplesmente nenhum treinamento depois de um certo período de tempo. Um problema similar ocorre quando os erros são muito grandes, ocasionando no problema da explosão dos erros dos gradientes, fazendo com que os pesos não possuam nenhuma representação significativa. Para tentar mitigar os problemas das RNNs, unidades de memória de longo e curto prazo (*long short-term memory* ou LSTM) foram introduzidas. As conhecidas redes LSTM possuem a capacidade de detectar dependências de longo prazo nos dados. No contexto de PLN, isso significa que elas são capazes de analisar sentenças/textos cada vez maiores e não “esquecer” das representações aprendidas previamente.

Em 2014, [Sutskever et al. 2014a] propôs uma abordagem chamada de *sequence-to-sequence learning* (aprendizado de sequência a sequência ou, apenas, *seq2seq*), que mapeia uma sequência para outra usando uma rede neural. A tarefa de Tradução de Texto é uma referência clássica de *sequence-to-sequence learning*, por exemplo. O progresso

com essa abordagem foi tão significativo que o *Google* anunciou em 2016 que estava substituindo oficialmente seus modelos de Tradução Automática por um modelo neural *sequence-to-sequence*. Arquiteturas que utilizam redes neurais RNN ou LSTM para um problema *sequence-to-sequence*, quando recebem uma sequência como entrada, as transformam em um vetor (operação chamada de codificação), realizam as operações necessárias e transformam o vetor resultante em uma sequência (operação chamada de decodificação). Como essas operações são praticamente etapas bem definidas, esse modelo acabou sendo cunhado como codificador-decodificador¹¹ [Sutskever et al. 2014b].

Uma das barreiras encontradas para esse tipo de modelo é a sua limitação no tamanho das sequências das representações internas das palavras (vetores utilizados na codificação). O codificador comprime toda a sentença de origem em um único vetor, o que pode ser muito complicado de calcular porque o número de significados possíveis é infinito. Quando o codificador é forçado a colocar todas as informações em um único vetor, é provável que ele “esqueça” alguma coisa.

Mecanismos de atenção [Bahdanau et al. 2014] de certa forma tratam com esse problema e foram propostos tentando mimetizar a forma com que humanos lidam com frases em geral: atentando-se apenas para trechos específicos de uma sentença por vez e não ao texto inteiro. Nas RNNs, por exemplo, é considerado apenas o último estado do codificador no modelo codificador-decodificador (ou seja, apenas as últimas palavras de uma sequência de entrada); na abordagem utilizando mecanismos de atenção, todos os estados do codificador são considerados em cada etapa de decodificação. Dessa forma, é possível acessar informações sobre todos os elementos da sequência de entrada.

De fato, mecanismos de atenção extraem informações de toda a sequência de entrada a partir da média ponderada usada como uma espécie de pontuação (*score*) de todos os estados passados do codificador, permitindo que o decodificador atribua maior importância a um determinado elemento da entrada para cada elemento da saída. O mecanismo é potencialmente útil para qualquer tarefa que exija a tomada de decisões com base em certas partes da entrada. A limitação desses mecanismos de atenção está relacionada ao seu tempo de treinamento, uma vez que para analisar um determinado elemento de entrada, todos os seus elementos anteriores devem ter sido analisados, tornando o treinamento dessas redes demorado e computacionalmente custoso.

A arquitetura *Transformer* [Vaswani et al. 2017] (e suas variações) é o atual estado-da-arte para tarefas de PLN. Esse modelo foi pensado exclusivamente para a tarefa de Tradução de Texto e extrai informações de cada palavra utilizando mecanismos de auto-atenção (*self-attention*) para inferir a importância de todas as palavras de uma sequência em relação à palavra que está sendo analisada, sem nenhuma unidade de RNN envolvida na operação. Em suma, mecanismos de auto-atenção são usados para observar as palavras ao redor (contexto) em um texto, para obter representações de palavras mais contextualmente sensíveis. Modelos *Transformers* utilizam apenas os *scores* e funções de ativação para que a arquitetura como um todo possa ser altamente paralelizável e eficiente.

A última grande inovação no mundo do PLN são, sem dúvida, os grandes *Lan-*

¹¹Também é conhecido por modelo *encoder-decoder* na literatura e será abordado com mais detalhes na Seção 1.6.1.

guage Models (LMs, ou Modelos de Linguagem) pré-treinados. Embora tenham sido propostos pela primeira vez em 2015 [Dai and Le 2015], apenas recentemente que mostraram uma grande melhoria em relação aos métodos de última geração para uma gama diversificada de tarefas. Os vetores de saída de LMs pré-treinados podem ser usados como *features* para um modelo final ou LMs pré-treinados podem ser ajustados aos dados da tarefa de destino, já que permitem um aprendizado eficiente com significativamente menos dados.

1.2.2. Conceitos Básicos de Redes Neurais

Imagine duas folhas de papel colorido: uma vermelha e outra azul. Coloque uma em cima da outra e amasse-as juntas em uma pequena bola. Considere que essa bola de papel amassada são os dados de entrada, e cada folha de papel é uma classe em um problema de classificação. O que uma rede neural deve fazer é descobrir uma forma de transformação da bola de papel que a desamassaria, de modo a tornar as duas classes separáveis novamente (veja a Figura 1.2). Com a Aprendizagem Profunda, isso seria implementado como uma série de transformações simples, como aquelas que poderiam ser aplicadas na bola de papel com os dedos, um movimento de cada vez.

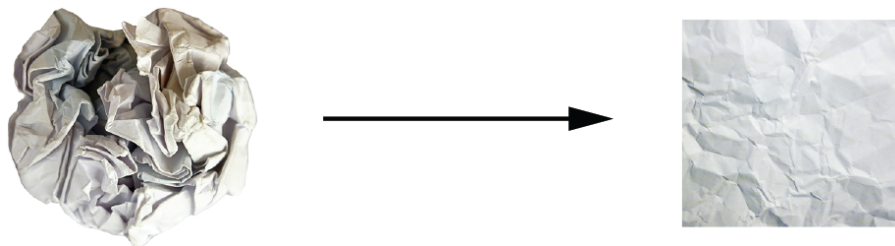


Figura 1.2. Desamassando uma variedade complicada de dados. Fonte: [Chollet 2021].

A Aprendizagem Profunda, também conhecida como Aprendizado Profundo (termos oriundos da expressão em inglês *Deep Learning*), é uma subárea do Aprendizado de Máquina que combina dezenas ou até centenas de camadas sucessivas de representações de processamento não linear para extrair e transformar atributos de entrada em modelos preditivos, chamados de redes neurais [Zhang et al. 2018]. Em suma, redes neurais profundas possuem uma estrutura matemática que aprende representações em vários estágios: as camadas mais próximas aos dados aprendem características simples, enquanto as de alto nível aprendem características mais complexas, sendo estas derivadas das camadas anteriores.

Apesar de modelos de Aprendizagem Profunda serem comumente chamados de redes neurais, estes métodos originaram-se como modelos de Aprendizagem de Máquina inspirados em neurônios biológicos. A flexibilidade dessas estruturas as tornaram ideais para tarefas complexas e, principalmente, que envolvem dados não estruturados, como imagens, sons, vídeos e textos.

As primeiras versões das redes neurais eram extremamente simples e capazes de realizar apenas cálculos lógicos básicos [McCulloch and Pitts 1943]. Com o passar dos anos, essas primeiras arquiteturas evoluíram para o *Perceptron* [Rosenblatt 1958]. A Figura 1.3 apresenta a unidade básica de um *Perceptron*, chamada de Unidade Lógica de

Limiar (do inglês, TLU). Essa unidade recebe entradas numéricas x e as associa a pesos w ao aplicar uma soma ponderada. Em seguida, uma função degrau é aplicada ao resultado dessa soma. A expressão $h_w(x) = \text{degrau}(z)$, onde $z = x^T w$ e $\text{degrau}(z)$ é a função de Heaviside, apresenta como as entradas são processadas em uma TLU.

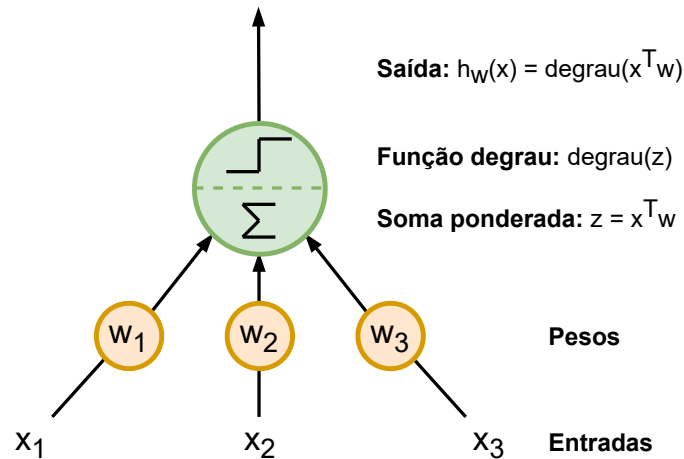


Figura 1.3. Unidade Lógica de Limiar (TLU). Adaptado de [Géron 2019].

Um *Perceptron* basicamente é uma estrutura formada por apenas uma camada de entrada e uma camada de TLUs, onde todas as TLUs estão ligadas a todas as entradas. O *Perceptron* também conta com um tipo adicional de neurônio chamado viés (*bias*), conectado a todas as TLUs. A Figura 1.4 mostra um exemplo de *Perceptron*. De forma similar à TLU, é possível calcular a saída de um *Perceptron* utilizando a fórmula $h_{W,b}(X) = \varphi(XW + b)$, onde W é a matriz dos pesos que ligam as entradas a todas as TLUs, X é a matriz de entradas, b é o valor do neurônio de viés, e φ a função de ativação das TLUs.

Para treinar um *Perceptron*, deve-se aplicar a seguinte regra de aprendizado após o fornecimento de uma instância dos dados de treinamento:

$$w_{i,j}^{proxima_iteracao} = w_{i,j} + \eta(y_j - \hat{y}_j)x_i \quad (1)$$

onde

- $w_{i,j}$ é o peso entre a entrada i e a TLU j ,
- η é a taxa de aprendizagem,
- y_j é o valor real da saída,
- \hat{y}_j é o valor estimado pelo *Perceptron* como saída,
- x_i é o valor da entrada i .

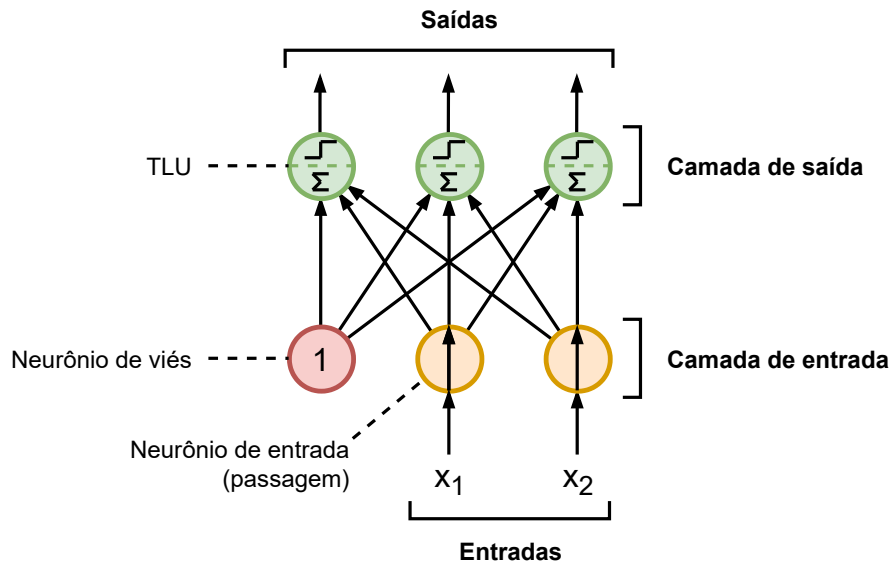


Figura 1.4. Arquitetura de um *Perceptron*. Adaptado de [Géron 2019].

O código do Programa 1.2.1 utiliza a implementação da biblioteca *Scikit-learn* de um *Perceptron* para classificar diferentes tipos da planta Íris.

Programa 1.2.1. Predição do tipo de planta Íris utilizando um *Perceptron* simples. Adaptado de [Géron 2019].

```
import numpy as np
from sklearn.datasets import load_iris
from sklearn.linear_model import Perceptron

iris = load_iris()
X = iris.data[:, (2,3)]
y = (iris.target == 0).astype(np.int)

per_clf = Perceptron()
per_clf.fit(X, y)
y_pred = per_clf.predict([[2, 0.5]])
```

Uma das principais limitações dos *Perceptrons* é o fato de não conseguirem resolver problemas relativamente simples, como o problema de classificação da porta lógica XOR. O *Perceptron* Multicamada (do inglês, MLP) surge como uma evolução do *Perceptron* convencional. A estrutura de uma MLP é formada por uma camada de entrada, n camadas empilhadas de TLUs, chamadas de camadas ocultas (*hidden layers*), e uma camada adicional de TLUs que representa a saída da MLP. A Figura 1.5 apresenta uma MLP simples com uma camada oculta. Quando uma MLP possui mais do que duas camadas ocultas, é considerada uma DNN (sigla para a expressão Rede Neural Profunda em inglês).

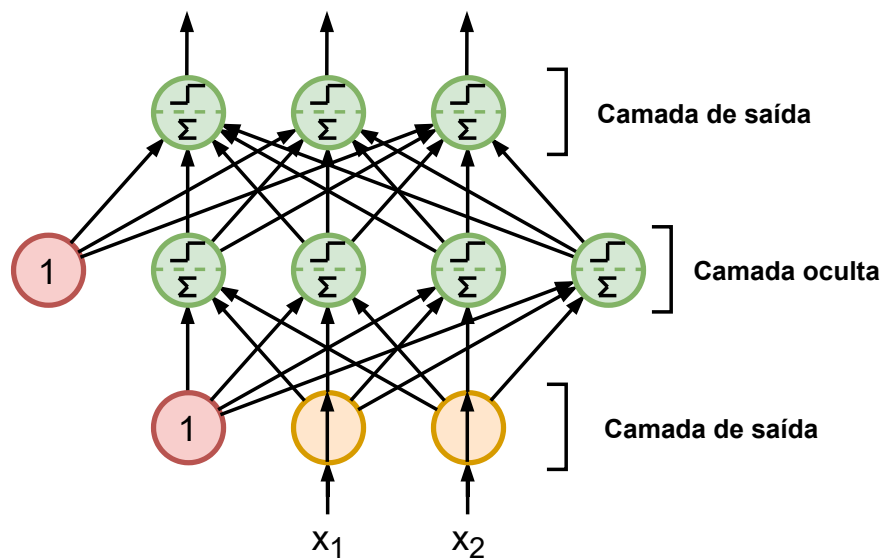


Figura 1.5. Arquitetura de um MLP com duas entradas e uma camada oculta. Adaptado de [Géron 2019].

Um dos conceitos mais importantes associados às DNNs é o algoritmo de retropropagação (*backpropagation*) [Rumelhart et al. 1985] utilizado para treiná-las. Esse algoritmo pode ser dividido em duas etapas: o *forward pass* e o *backward pass*. Na etapa de *forward pass*, a DNN recebe um *batch* (lote) de instâncias de dados e calcula seus resultados após a interação com todas as suas camadas. Em seguida, o algoritmo calcula o erro entre a saída gerada pela rede e a saída esperada presente nos dados, e utiliza esse resultado para calcular como cada peso da DNN contribui com esse erro encontrado. Por fim, o algoritmo executa um passo de gradiente descendente para atualizar todos os pesos da rede neural.

Um outro fator importante que diferencia um *Perceptron* de uma DNN são as funções de ativação utilizadas em cada neurônio (TLU): as redes profundas utilizam funções de ativação não lineares, como tangente hiperbólica e linear retificada (ReLU) para aproximar qualquer função contínua. O trecho de código a seguir exemplifica a utilização de uma DNN ao construir¹² uma rede capaz de classificar diferentes frases contendo, em geral, uma conotação de sentimentos positivos ou negativos.

O Programa 1.2.2 cria uma DNN com quatro camadas para o problema de Análise de Sentimentos de frases: a primeira recebe um vetor de TF-IDF (detalhes na Seção 1.4.1), que por sua vez é utilizado como entrada para a rede. A segunda e terceira camadas são criadas em um formato denso, isto é, com todos os seus neurônios conectados a todos os neurônios da camada anterior, com cinco (5) neurônios cada e função de ativação ReLU. Por fim, a quarta camada densa com dois (2) neurônios utiliza a função de ativação *softmax* para informar qual classe a frase de entrada tem mais chances de pertencer, se zero (0), para frases com sentimentos negativos, ou um (1), para frases com sentimentos positivos.

¹²A importação e divisão dos dados serão omitidos para que se possa focar apenas na DNN.

Programa 1.2.2. Criação, compilação e treinamento de uma rede neural profunda com duas camadas escondidas com cinco (5) neurônios cada.

```
import tensorflow as tf
from tensorflow import keras

model = keras.models.Sequential()
model.add(keras.layers.Flatten())
model.add(keras.layers.Dense(5, activation="relu"))
model.add(keras.layers.Dense(5, activation="relu"))
model.add(keras.layers.Dense(2, activation="softmax"))

model.compile(loss="sparse_categorical_crossentropy",
              optimizer="sgd", metrics=["accuracy"])

history = model.fit(X_train, y_train, epochs=30)
```

Uma vez que a DNN é criada com a biblioteca *Keras*, é preciso configurar o modelo para treinamento através da função `compile()`, definindo, por exemplo, qual a função de perda (*loss*), otimizador e métricas que serão utilizadas. A entropia cruzada (do inglês, *cross-entropy*) foi utilizada no Programa 1.2.2 como função de perda, pois as classes possíveis de saída são discretas e exclusivas. Para treinar o modelo, é usado um gradiente descendente estocástico (parâmetro "sgd"). Por fim, como métrica de avaliação, utiliza-se nesse exemplo a acurácia ("accuracy"), finalizando a etapa de *backward pass*. Para treinar o modelo criado, basta chamar a função `fit()`, passando os dados utilizados e a quantidade de épocas que o modelo será otimizado. Um ciclo que passa pelos exemplos de treinamento da rede é chamada de época. As épocas são repetidas até ocorrer uma convergência, isto é, um ponto no treinamento após o qual as mudanças calculadas com a métrica de avaliação tornam-se menores e os erros produzidos pelo modelo (computados pela função de perda) são mínimos.

O *Perceptron* multicamada é indicado para tarefas onde têm-se, essencialmente, entradas de tamanho fixo e que resultam em saídas discretas. Para problemas com entradas em forma de sequências, como áudios, textos ou vídeos, essas arquiteturas mais simples não são as mais indicadas. Para esse tipo de problema com entradas muito longas, é comum o uso de arquiteturas chamadas de Redes Neurais Recorrentes.

A semelhança entre MLPs tradicionais e RNNs é que estas últimas também possuem conexões para neurônios de camadas anteriores. Essas conexões são capazes de modelar informações anteriores de uma determinada sequência de entrada. A Figura 1.6 (à esquerda) mostra um único neurônio, com uma ligação adicional da sua saída para sua entrada. Isso significa que a cada passo de tempo t , este neurônio recebe como entrada o trecho da sequência x_t e o trecho anterior x_{t-1} que vem do passo de tempo anterior ($t - 1$). Esse processo é realizado recursivamente até o passo de tempo $t = 0$. A Figura 1.6 (à direita) também ilustra como as autoconexões de um neurônio funcionam ao longo do tempo.

Da mesma forma em que é criada uma camada de neurônios “convencionais” em

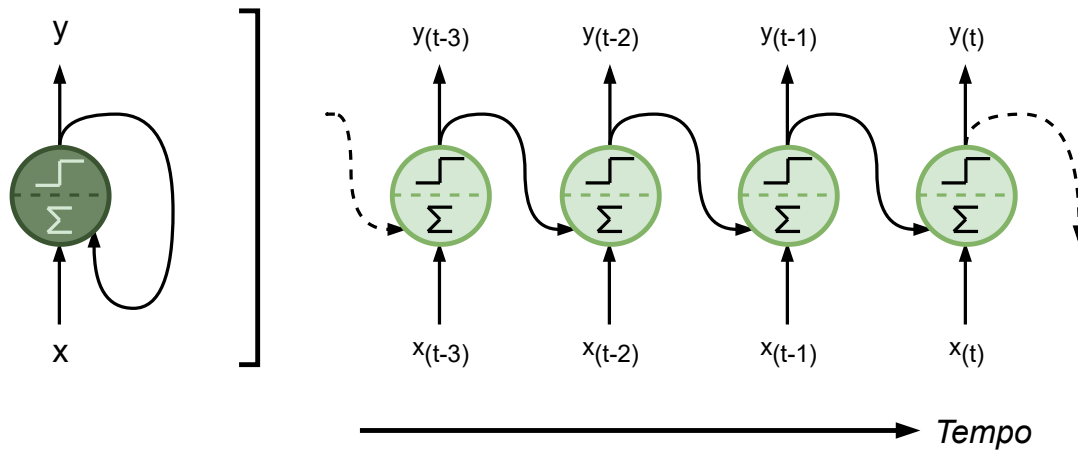


Figura 1.6. Unidade RNN (à esquerda) e unidade RNN desenrolada (à direita). Adaptado de [Géron 2019].

uma MLP, é possível criar uma camada de neurônios com novas autoconexões, como exemplificado na Figura 1.7.

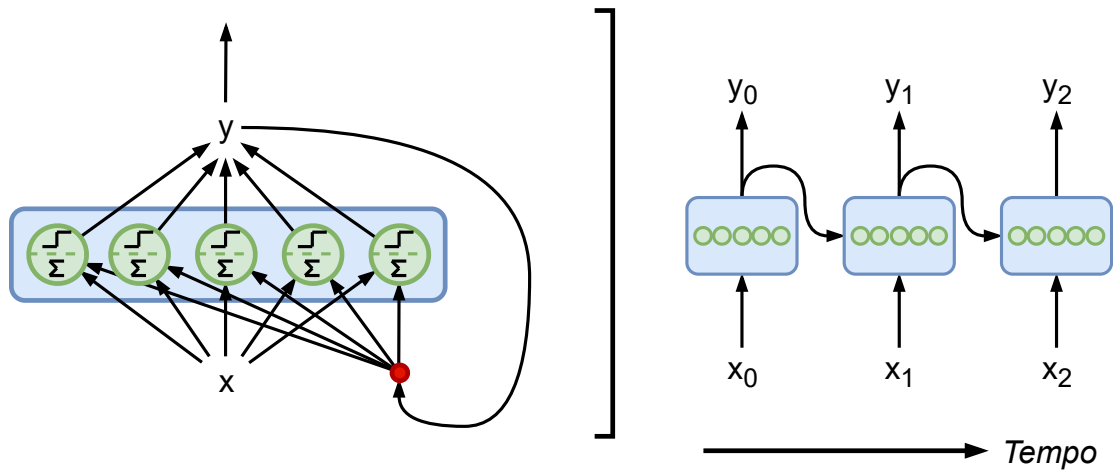


Figura 1.7. Rede neural com uma camada de unidades RNN (à esquerda) e a mesma rede desenrolada ao longo do tempo (à direita). Adaptado de [Géron 2019].

Para treinar uma RNN, deve-se aplicar a seguinte regra de aprendizado após fornecer todas as entradas do intervalo de tempo t dentro de uma matriz X_t :

$$Y_t = \phi([X_t Y_{t-1}]W + b) \quad (2)$$

onde

- Y_t é a matriz de saídas da RNN no intervalo de tempo t ,

- X_t é a matriz de entradas da RNN no intervalo de tempo t ,
- Y_{t-1} é a matriz de saídas da RNN no intervalo de tempo $t - 1$,
- $W = \begin{bmatrix} W_x \\ W_y \end{bmatrix}$ é o empilhamento dos pesos relacionados às entradas X_t e às saídas Y_{t-1} ,
- b é o vetor de neurônios de viés.

Quando é preciso lidar com sequências longas, células RNN tendem a crescer bastante ao longo do tempo. Isso é um potencial problema, dado que torna o treinamento computacionalmente custoso, demorado e instável. Um outro problema que surge quando lida-se com sequências longas é a perda de informações do começo da sequência por parte da RNN. Um exemplo real disso ocorre quando uma rede tenta traduzir uma sentença muito longa e perde informações do começo da frase.

Para algumas tarefas onde a ordem das entradas (ordem das palavras em um texto, por exemplo) são importantes, o uso de RNNs bidirecionais (BiRNNs) podem trazer um ganho significativo à performance do modelo. Uma BiRNN é composta por duas RNNs convencionais, em que uma processa a sequência de entrada em sua ordem natural, e a adicional processa a sequência em sua forma inversa. No final, é feita uma operação de concatenação das representações das duas RNNs. Dessa forma, uma BiRNN consegue capturar padrões que uma RNN convencional não consegue sozinha.

Alguns novos tipos de células baseadas em RNNs foram desenvolvidas para mitigar esses problemas. Dentre elas, a mais famosa é a célula de LSTM. A Figura 1.8 apresenta, de forma simplificada, uma célula LSTM. Comparando-a com uma célula RNN, como a da Figura 1.6, células LSTM possuem algo extra, visto que adicionam uma divisão em seu estado: o vetor $h_{(t)}$ representa o estado de curto prazo e o vetor $c_{(t)}$ representa o estado de longo prazo.

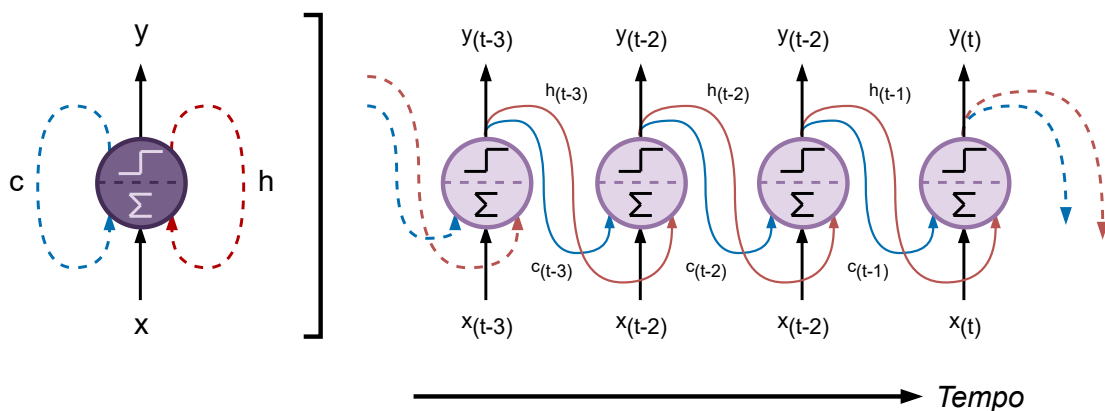


Figura 1.8. Unidade LSTM (à esquerda) e unidade LSTM desenrolada (à direita).

1.3. Preparação de Dados Textuais

Como o texto, em particular, origina a maior parte das comunicações e da produção cultural da humanidade, não é coincidência que ele também sirva de base para a criação de conjuntos de dados textuais¹³, os quais são chamados de **corpus linguísticos**. Assim, em PLN, os modelos de Aprendizagem Profunda reconhecem padrões que possam ser aplicados a palavras, frases e/ou parágrafos inteiros.

Porém, não é possível alimentar modelos com palavras ou textos. Informações algébricas (ou seja, vetores numéricos) é que normalmente são extraídas de um corpus como forma de representar unidades linguísticas, especialmente, palavras. A representação vetorial de uma palavra pode ser obtida mediante um procedimento chamado de *text-vectorization* (vetorização do texto), que pode ser feito de várias maneiras. Dentre elas, segmentar o texto em palavras ou caracteres, ou até mesmo extrair *n-grams*¹⁴, e transformar essas “partes” do texto em vetores [Chollet 2021].

Embora existam diversas formas de processar e preparar dados textuais para serem utilizados em modelos de Aprendizagem Profunda, quase todos seguem um mesmo padrão:

- Primeiramente, o texto deve ser **padronizado**, convertendo todas as letras em minúsculas¹⁵ e removendo pontuações;
- Em seguida, é necessário dividir o texto em várias partes, chamados de *tokens*. Esse processo é conhecido como **tokenização**;
- Por fim, os *tokens* devem ser transformados em representações numéricas. É comum que sejam utilizadas estratégias de **indexação** e de *embeddings*.

Seguindo esse roteiro, os dados em formato numérico podem ser usados para alimentar o modelo neural a ser treinado para alguma tarefa de PLN. A seguir, é explicada de forma mais detalhada cada caso.

1.3.1. Padronização

É comum que existam sentenças que apresentam uma mesma informação, mas que possuem estruturas diferentes. Suponha as seguintes frases: “*Olha só! Como PLN é legal! Estava estudando esses dias*”, “*Olha só, como PLN é legal! Estive estudando esses dias*”, e “*olha so como pln eh legal, estudei esses dias*”. Apesar da informação passada ser a mesma, as frases podem muito bem ter sido escritas por três pessoas diferentes: uma mais animada, outra cética e outra mais informal, respectivamente.

Ainda utilizando o exemplo anterior, daria mais trabalho para um modelo aprender que *PLN* e *pln* tratam-se da mesma sigla, ou que *só* e *so* na verdade são um mesmo termo.

¹³Mesmo se a tarefa de PLN envolva fala ou áudios, uma transcrição é feita para que os arquivos de sons sejam convertidos em documentos textuais.

¹⁴*N-grams* (N-gramas) são grupos formados por *n* palavras ou caracteres consecutivos de uma determinada amostra de texto.

¹⁵Por convenção, a transformação de todas as letras em minúsculas é utilizada, mas a conversão para letras maiúsculas também é aceita.

A ideia principal da padronização é diminuir as diferenças entre textos que apresentam informações similares, permitindo assim que modelos não se preocupem em aprender tais detalhes.

Uma das estratégias mais básica e direta é, simplesmente, remover a pontuação e colocar os caracteres para minúsculo. Tal modificação tornam as frases bem mais semelhantes. Outra estratégia também comum é trocar as letras e vogais com acentos por suas versões sem acento, o que auxilia quando é trabalhado com dados de redes sociais, por exemplo, onde não é comum a formalidade de escrita. Uma forma simples de aplicação do processo de padronização é apresentada no Programa 1.3.1.

Programa 1.3.1. Padronização da sentença “O gato caçou o rato!”.

```
import string
from unidecode import unidecode

def standardize(text):
    text = unidecode(text.lower())
    return "".join(char for char in text
                    if char not in string.punctuation)

standardize("O gato caçou o rato!")
# 'o gato caçou o rato'
```

1.3.2. Tokenização

Uma vez padronizado, é necessário quebrar o texto em unidades a serem vetorizadas. Dividir um texto em unidades distintas (palavras, caracteres ou *n-grams*) nada mais é do que **tokenizar** este texto [Webster and Kit 1992]. Em geral, é mais comum dividir o texto em palavras ou *n-grams*.

Existem dois tipos principais de estratégias para formar *embeddings* (vetores): aquelas que utilizam a sequência de *tokens* para agregar uma informação de contexto e ordem (chamados de modelos sequenciais), e aqueles que importam-se apenas com a ocorrência e frequência dos *tokens* (chamados de modelos *bag-of-words*). Essas estratégias são apresentadas e explicadas com mais detalhes na Seção 1.4.

Para os modelos sequenciais, é comum que os *tokens* sejam criados a nível das palavras, para poder processar a formação do texto. Em contrapartida, para os modelos de *bag-of-words*, é comum que a tokenização seja feita a nível de *n-grams* a fim de contar/identificar a presença de termos e expressões. O processo de tokenização é representado de forma simplificada no Programa 1.3.2.

Entretanto, vale ressaltar que existem palavras que diferem de outras apenas por sufixos flexionais, modificando seu gênero, número, grau, etc., como em *presidente* e *presidenta*, *estava* e *estive*, *casa* e *casas*. Removendo esses elementos linguísticos e reduzindo as palavras aos seus radicais (*stem*), tem-se o processo chamado de *stemming* ou estemização. Para esse procedimento, existem funções prontas da biblioteca NLTK, conforme o código exemplo no Programa 1.3.3. Note como as frases distintas parecem

Programa 1.3.2. Padronização e tokenização da sentença “O gato caçou o rato!”. Os *tokens* obtidos são a nível das palavras.

```
def tokenize(text):
    text = standardize(text)
    return text.split()

tokenize("O gato caçou o rato!")
# ['o', 'gato', 'caçou', 'o', 'rato']
```

ainda mais semelhantes se comparados os *stems* ao invés das próprias palavras.

Programa 1.3.3. Estemização das sentenças “Os gatos caçaram os ratos” e “O gato caçou o rato”. Ambas retornam os mesmos *stems* quando aplicada a função *stem*: o, gat, cac, o, e rat.

```
from nltk.stem.snowball import PortugueseStemmer
stemmer = PortugueseStemmer()

def stem(text):
    return [stemmer.stem(w) for w in tokenize(text)]

stem("Os gatos caçaram os ratos")
# ['os', 'gat', 'cac', 'os', 'rat']

stem("O gato caçou o rato")
# ['o', 'gat', 'cac', 'o', 'rat']
```

Às vezes torna-se necessário não só remover sufixos flexionais, como também manter algumas dessas informações (tempo verbal, por exemplo) ou, até mesmo, mapear os *stems* para uma outra palavra que melhor os represente (o *stem* da palavra *casaram* poderia ser, por exemplo, *casar*, dando uma resposta diferente do *stem* de *casas*, que seria *casa*). Esse processo é conhecido como lematização (*lemmatization*). Para conseguir realizar esse processamento, são necessários modelos ou técnicas mais complexas. É possível realizar a lematização em Português com a biblioteca *spaCy* (Programa 1.3.4). Lembre-se que tanto *stems* como *lemmas* funcionam no final como *tokens*.

1.3.3. Indexação do Vocabulário

Uma vez que os *tokens* do texto são encontrados, é necessário codificá-los em uma representação numérica. Embora seja possível aplicar alguma função *hash* para os *tokens* e encontrar um número que os represente, é mais comum optar por construir um vocabulário de todos os termos presentes nos textos processados. Para cada *token*, atribui-se um número que serve como seu identificador (ou índice) dentro do vocabulário. Uma forma de implementar como obter um vocabulário pode ser encontrada no Programa 1.3.5.

Com o vocabulário construído, é possível utilizar o índice de cada termo e codificá-lo de forma vetorial utilizando uma estratégia chamada de codificação *one-hot* (também

Programa 1.3.4. Lematização das sentenças “Amigos, amizade e carreira são importantes” e “Amigas, amigas e carreira é importante”. Os tokens obtidos com a função lemma para ambas as frases são amigo, amizade, e, carreira, ser, e importante.

```
import spacy

nlp = spacy.load('pt')

def lemma(text):
    doc = nlp(vectorizer.standardize(text))
    return [token.lemma_ for token in doc]

lemma("Amigos, amizade e carreira são importantes")
# ['amigo', 'amizade', 'e', 'carreira', 'ser',
  ↪ 'importante']

lemma("Amigas, amigas e carreira é importante")
# ['amigo', 'amizade', 'e', 'carreira', 'ser',
  ↪ 'importante']
```

Programa 1.3.5. Criação de vocabulário para um pequeno corpus formado pelas sentenças “O gato caçou o rato” e “Gato e cachorro são animais”.

```
def make_vocab(texts):
    # O vocabulário é iniciado com 0 que indica strings
    ↪ vazias, e 1 que indica palavras desconhecidas
    vocab = {"": 0, "[UNK]": 1}
    for text in texts:
        text = standardize(text)
        tokens = tokenize(text)
        for token in tokens:
            if token not in vocab:
                vocab[token] = len(vocab)
    inverse_vocab = dict(
        (v, k) for k, v in self.vocab.items())

    return vocab, inverse_vocab

make_vocab(["O gato caçou o rato",
           "Gato e cachorro são animais"])
# {'': 0, '[UNK]': 1, 'o': 2, 'gato': 3, 'caçou': 4, 'rato':
  ↪ 5, 'e': 6, 'cachorro': 7, 'são': 8, 'animais': 9}
```

conhecida como *dummy*). Um vetor *one-hot* possui o mesmo tamanho do vocabulário (ou seja, sua dimensão é igual a quantidade de *tokens*) e é preenchido com zeros, exceto na posição referente ao índice da palavra, onde contém o valor 1. O método de codificação *one-hot* é útil por alguns motivos: (i) todos os vetores estão normalizados; (ii) é fácil de descobrir a palavra que um vetor representa; e (iii) as operações vetoriais são mais

simples, permitindo trabalhar com matrizes esparsas¹⁶. Ainda assim, o uso de vetores *one-hot* não são a melhor forma de como representar palavras e textos.

O Programa 1.3.6 mostra como é possível implementar a construção dos vetores *one-hot* a partir de uma sentença exemplo.

Programa 1.3.6. Codificação *one-hot* da sentença “O gato caçou o rato”.

```
import numpy as np

def one_hot_encoder(text, vocab):
    tokens = tokenize(text)
    vectors = np.zeros((len(tokens), len(vocab)))
    for i, token in enumerate(tokens):
        token_idx = vocab.get(token, 1)
        vectors[i, token_idx] = 1
    return vectors

one_hot_encoder("O gato caçou o rato", vocab)
# array([[0., 0., 1., 0., 0., 0., 0., 0., 0., 0.],
#        [0., 0., 0., 1., 0., 0., 0., 0., 0., 0.],
#        [0., 0., 0., 0., 1., 0., 0., 0., 0., 0.],
#        [0., 0., 1., 0., 0., 0., 0., 0., 0., 0.],
#        [0., 0., 0., 0., 0., 1., 0., 0., 0., 0.]])
```

1.4. Representação Textual

O conceito de representação textual não baseia-se apenas em vetores, muito menos em vetores de palavras *one-hot*. Primeiramente, deve-se analisar como palavras formam uma sentença. Línguas diferentes ordenam palavras semelhantes ou não de maneiras diferentes e, apesar da ordem importar, sua relação com o significado não é tão direta. Como representar as palavras é o principal motivo pelo qual surgem diferentes tipos de arquiteturas de PLN, que podem ser agrupadas em três categorias principais:

- 1. Representando palavras como conjuntos.** Trata-se da representação mais simples. Aqui a ordem das palavras simplesmente é descartada e o texto é tratado como um conjunto não ordenado de palavras, obtido essencialmente por modelos *bag-of-words*.
- 2. Representando palavras como sequências.** Quando a ordem das palavras é considerada, elas devem ser processadas estritamente na ordem em que aparecem, uma de cada vez. Quanto a isto, os já citados vetores de palavras, ou *word embeddings*, são amplamente utilizados. Métodos específicos (*Word2Vec* e *GloVe*, por exemplo) e/ou arquiteturas de redes neurais (como RNNs e *Transformers*) são modelos de sequência de palavras cujas saídas geram *word embeddings*.

¹⁶Matriz esparsa é uma matriz onde a maioria dos valores é 0.

3. Modelando as palavras. Por possuir diferentes arquiteturas, a segunda abordagem ainda pode ser fragmentada em diversas outras. Uma delas, talvez a mais popular atualmente, relaciona-se a um conceito um pouco abstrato: o de como modelar uma linguagem.

Mas, o que significa “*modelar uma linguagem*”? Ou, ainda, o que significa “*modelar algo*”? Um bom modelo deve simular o comportamento do mundo real, entendendo quais eventos são mais prováveis de acontecer, isto é, prever o que acontece a seguir dada alguma **descrição de contexto**. Um modelo de linguagem não é diferente: um evento pode ser uma unidade linguística (texto, frase, *token*, ou símbolo), onde o objetivo final do modelo é estimar as probabilidades de cada unidade linguística ocorrer.

Assim, os chamados *Language Models* estimam a probabilidade de diferentes unidades linguísticas, como símbolos, *tokens*, e sequências de *tokens*. Algumas arquiteturas que usam *Transformers*, por exemplo, são LMs, podendo ou não funcionar do jeito tradicional, ao computar probabilidades. Na verdade, esse conceito é bastante conhecido e LMs estão em ação todos os dias, sejam em *sites* de busca, serviços de tradução, teclados de celulares, e até mesmo na correção de escrita feita em serviços de *e-mail*. A Figura 1.9 ilustra de forma didática o funcionamento de um LM.

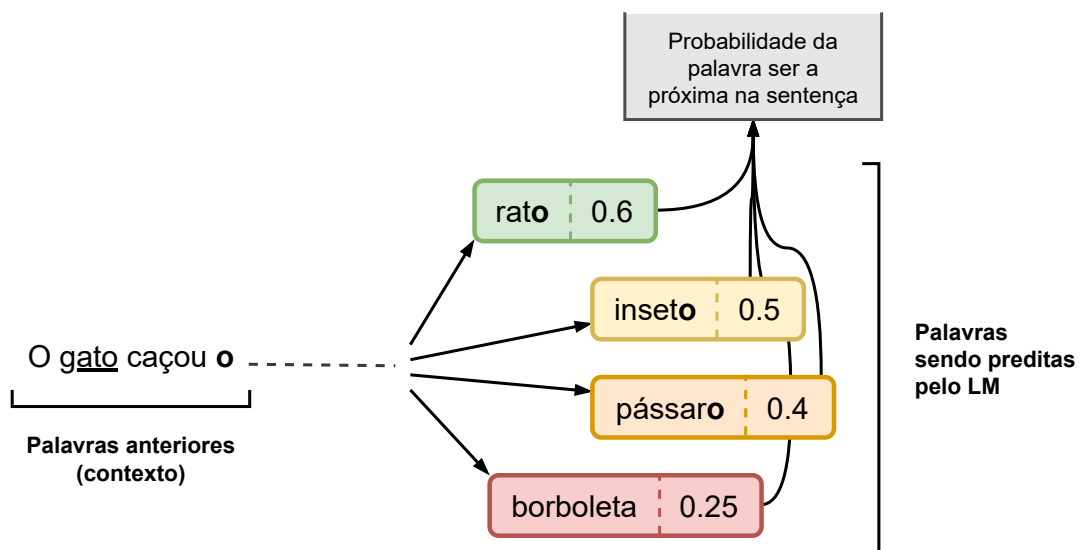


Figura 1.9. LM identificando qual a próxima palavra na frase “*O gato caçou o...*”. À direita, estão listadas algumas das possíveis palavras juntamente com suas probabilidades. À esquerda, a palavra *gato* e o artigo *o* destacados indicam os termos mais importantes do “contexto” para predição. Nesse caso, a próxima palavra da sentença seria *rato*.

O escopo deste curso não inclui o aprofundamento dos tipos de LMs existentes utilizados para estimar probabilidades de palavras, pois seria necessário o estudo de diversos modelos estatísticos, além de conceitos da Teoria da Probabilidade, como Probabilidade Condicional. O que é importante que o leitor saiba¹⁷ é que, dada uma palavra, o objetivo

¹⁷As informações apresentadas até aqui sobre LMs serão utilizadas nas próximas seções.

de um LM é identificar qual seria a próxima palavra, de acordo com o contexto da sentença. Porém, não só um LM pode atribuir probabilidades para próximas palavras, como também pode dar probabilidades a frases, como representado na Figura 1.10.

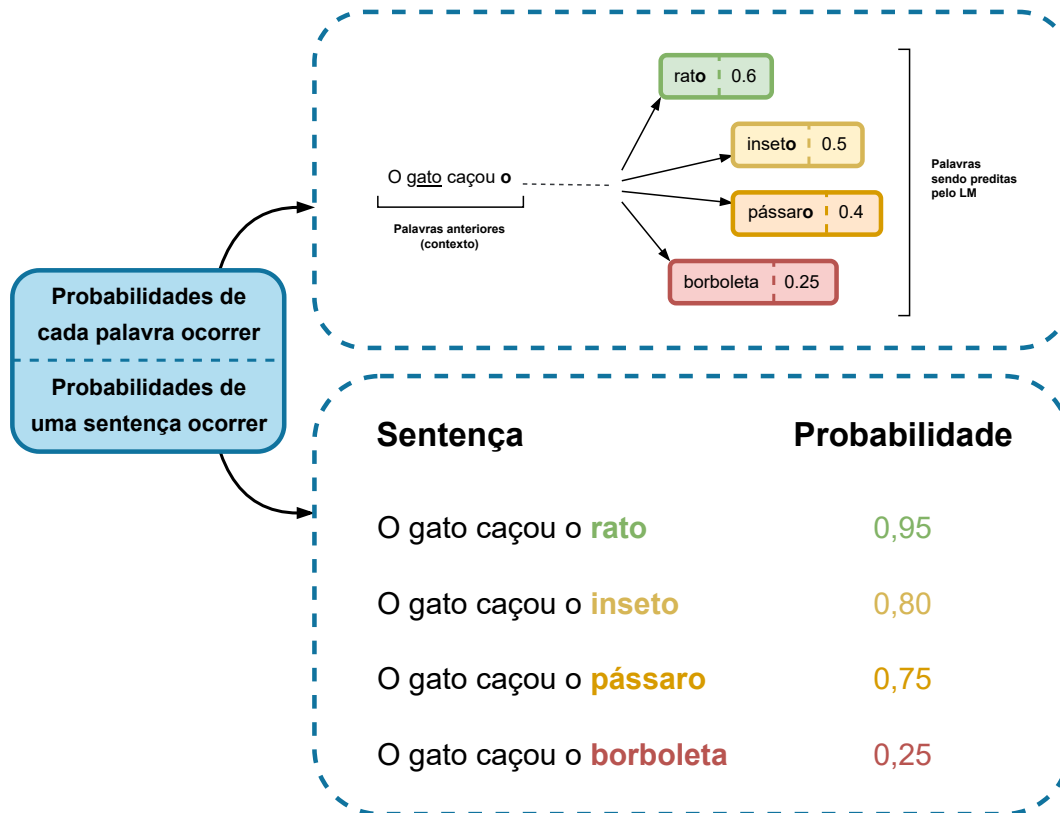


Figura 1.10. Além de palavras, LMs determinam qual sentença é mais adequada que outras com base no contexto.

Mesmo que modelos *bag-of-words* pareçam obsoletos se comparados com as soluções sofisticadas dos modelos de sequência não importa para qual tarefa ou conjunto de dados, isso definitivamente não é o caso: ambas as abordagens de como formar vetores de palavras descritas brevemente acima permanecem igualmente relevantes [Chollet 2021].

1.4.1. Representação por *Bag-of-Words*

A maneira mais simples de representar um texto é **descartar a ordem** e tratá-lo como um **conjunto de tokens**. Existem diferentes formas de codificação com modelos *bag-of-words*, desde considerar palavras individuais (*unigrams*), até recuperar informações locais examinando grupos de sequências de *tokens* (*n-grams*).

Unigrams com codificação multi-hot. Para exemplificação de um conjunto de *unigrams* (*bag-of-unigrams*), observe o Programa 1.4.1 que transforma a sentença “o gato caçou o rato” já padronizada.

A função `make_ngrams` presente no trecho de código do Programa 1.4.1 possui um nome genérico e para o parâmetro `n` da função está atribuído o valor 1. Repare que

Programa 1.4.1. *Bag-of-unigrams* da frase “o gato caçou o rato”. Atente para o uso da função `make_ngrams`, que inicializa qualquer conjunto de *n-grams*.

```
sent = "o gato caçou o rato"

def make_ngrams(text, n=1):
    words = text.split()
    ngrams = [tuple(words[i:i + n]) for i in range(len(words) -
    → n + 1)]
    return ngrams

unigrams = make_ngrams(sent, n=1)

print(unigrams)
# [('o',), ('gato',), ('caçou',), ('o',), ('rato',)]
```

unigrams nada mais são do que palavras únicas sem nenhuma ordem, e que a intenção é formar um conjunto apenas com as palavras distintas do texto.

Obtidos os *unigrams* da sentença exemplo, falta apenas aplicar a codificação *multi-hot*, também chamada de binária (*multi-hot binary encoding*). Diferentemente da codificação *one-hot*, a *multi-hot* considera como entrada mais de um valor (no caso, mais de uma palavra). Sua funcionalidade será melhor explicada adiante porque, para *bag-of-unigrams*, o vetor resultante acaba sendo o mesmo que o da codificação *one-hot*.

Programa 1.4.2. Codificando os *unigrams* da sentença “o gato caçou o rato” com codificação *multi-hot* (função `multihot_encoder`).

```
from sklearn.preprocessing import MultiLabelBinarizer

def multihot_encoder(ngrams):
    mlb = MultiLabelBinarizer()
    binary_format = mlb.fit_transform(ngrams)
    print("Índices correspondentes de cada palavra =
    → {} \n".format({k: v for k, v in
    → enumerate(mlb.classes_)}))
    return binary_format

unigrams_vectors = multihot_encoder(unigrams)

print(unigrams_vectors)
# Índice correspondente de cada palavra = {0: 'caçou', 1:
→ 'gato', 2: 'o', 3: 'rato'}

# array([[0, 0, 1, 0],
#        [0, 1, 0, 0],
#        [1, 0, 0, 0],
#        [0, 0, 1, 0],
#        [0, 0, 0, 1]])
```

Para facilitar, a função `multihot_encoder` no Programa 1.4.2 faz uso do método `MultiLabelBinarizer`¹⁸ da biblioteca *Scikit-learn*, comumente utilizado para codificação de *bag-of-words*. Em suma, o trecho de código do Programa 1.4.2 demonstra que a sentença seguiu com o padrão da codificação *one-hot*, onde cada palavra é representada como um único vetor preenchido por zeros, exceto as posições que indicam o índice correspondente, ditadas pelo valor 1.

***N-grams* com codificação *multi-hot*.** Dependendo do contexto, descartar a ordem das palavras pode não contribuir para o seu entendimento. Por exemplo, o termo *Bela Cruz* em um texto com descrições sobre cidades do estado do Ceará transmite um conceito bastante distinto do que o significado das palavras *bela* e *cruz* separadamente. Faz-se necessário inserir informações extras para uma representação de *bag-of-words* olhando para os *n-grams* em vez de palavras únicas (mais comumente, são utilizados *bigrams*). Com *bigrams* (*bag-of-bigrams*), a sentença do exemplo anterior no Programa 1.4.3 origina um diferente conjunto e diferentes vetores codificados.

Programa 1.4.3. *Bag-of-bigrams* e vetores resultantes da sentença “o gato caçou o rato” com codificação *multi-hot*.

```
bigrams = make_ngrams(sent, n=2)

print(bigrams)
# [('o', 'gato'), ('gato', 'caçou'), ('caçou', 'o'), ('o',
  → 'rato')]

bigrams_vectors = multihot_encoder(bigrams)

print(bigrams_vectors)
# Índices correspondentes de cada palavra = {0: 'caçou', 1:
  → 'gato', 2: 'o', 3: 'rato'}

# array([[0, 1, 1, 0],
#        [1, 1, 0, 0],
#        [1, 0, 1, 0],
#        [0, 0, 1, 1]])
```

O uso da codificação *multi-hot* torna-se mais intuitivo no Programa 1.4.3. O conjunto de *bigrams*, formado pelas tuplas (o, gato), (gato, caçou), (caçou, o), e (o, rato), e o vetor de saída resultante possuem quatro itens, e os índices de cada vetor seguem as mesmas posições que as palavras únicas da sentença (valor 0 para *caçou*, 1 para *gato*, e assim sucessivamente). Por isso que, para o primeiro vetor [0, 1, 1, 0], correspondente à tupla (o, gato), o valor 1 está presente nas posições 1 e 2, já que estas dizem respeito aos índices dos termos *gato* e *o*.

Por conta do parâmetro `n` em `make_ngrams`, é possível configurá-lo para retornar *n-grams* arbitrários além de *bigrams*: *trigrams*, *4-grams*, *5-grams*, etc.

¹⁸<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MultiLabelBinarizer.html>

***N-grams* com codificação TF-IDF.** Também é possível adicionar mais informações à representação por *bag-of-words* ao elencar quantas vezes cada palavra ou *n-grams* ocorrem no texto. Essa abordagem é possível pois, conforme um termo aparece abundantemente em um texto, entende-se que ele é crucial para o seu entendimento. Contar a frequência de palavras ou *n-grams* em um corpus também é importante, uma vez que é possível saber quais palavras repetem-se mais e quais pertencem a pequenos subconjuntos. Basicamente, o TF-IDF (*term frequency, inverse document frequency*) é uma métrica que une essas duas ideias.

No Programa 1.4.4, com a codificação TF-IDF, um termo foi “ignorado”: a letra *o*. Isso ocorre porque o método `TfidfVectorizer`¹⁹ usado possui uma propriedade oriunda desse tipo de codificação chamada de normalização. É por esse motivo que esse tipo de codificação também é conhecida como TF-IDF *normalization*. Como citado anteriormente, se um termo aparecer muitas vezes em um texto, ele provavelmente deve ser importante, mas será mesmo? Termos que aparecem em quase todos os textos (como “*o*” ou “*a*”, considerados *stopwords*²⁰, não são particularmente informativos, enquanto os que são distintos ou que aparecem apenas em um pequeno subconjunto são, portanto, importantes. A normalização é efetuada porque o cálculo do TF-IDF pondera um determinado termo tomando como base a sua frequência (quantas vezes ele aparece em um texto), e dividindo-a por uma medida de frequência do documento, que estima a frequência com que o termo aparece em todo o corpus.

Programa 1.4.4. Codificação TF-IDF para os *bigrams* da sentença “*o gato caçou o rato*”.

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf_encoder = TfidfVectorizer(ngram_range=(1, 2))
# Limite inferior e superior do intervalo de n valores para os
  → diferentes n-grams a serem extraídos (bigrams)

tfidf_ngrams_vectors = tfidf_encoder.fit_transform([sent])

feature_names = tfidf_encoder.get_feature_names_out()
dense = tfidf_ngrams_vectors.todense()

print("Índices correspondentes de cada palavra =
  → {}".format({k: v for k, v in enumerate(feature_names)}))
# Índices correspondentes de cada palavra = {0: 'caçou', 1:
  → 'caçou rato', 2: 'gato', 3: 'gato caçou', 4: 'rato'}

print("Codificação por TF-IDF = {}".format(dense.tolist()))
# Codificação por TF-IDF = [[0.4472136, 0.4472136, 0.4472136,
  → 0.4472136, 0.4472136]]
```

Por isso que no Programa 1.4.4 a letra *o* não foi considerada. Note que o vetor

¹⁹https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

²⁰Lista de palavras muito comuns de um idioma, mas pouco informativas, que podem ser ignoradas.

resultante possui valores iguais para cada *bigram*. Isso ocorre porque o método foi originalmente pensado para ser aplicado em um conjunto de dados completo, não apenas em uma frase. Veja a diferença da representação obtida no Programa 1.4.5 para um pequeno corpus.

Programa 1.4.5. Codificação TF-IDF para *unigrams* de um corpus exemplo com apenas duas frases. Observe que os termos “o” e “e” não foram considerados por conta da normalização. As posições dos que possuem valor 0 nos vetores indicam que o *token* não aparece na frase.

```
corpus = ["o gato caçou o rato",
          "gato e cachorros são animais"]

tfidf_encoder = TfidfVectorizer()
tfidf_corpus_vectors = tfidf_encoder.fit_transform(corpus)

feature_names = tfidf_encoder.get_feature_names_out()
dense = tfidf_corpus_vectors.todense()

print("Índices correspondentes de cada palavra =
→ {}".format({k: v for k, v in enumerate(feature_names)}))
# Índices correspondentes de cada palavra = {0: 'animais', 1:
→ 'cachorros', 2: 'caçou', 3: 'gato', 4: 'rato', 5: 'são'}

print("Codificação por TF-IDF = {}".format(dense.tolist()))
# Codificação por TF-IDF = [
→ [0.0, 0.0, 0.6316672017376245, 0.4494364165239821,
→ 0.6316672017376245, 0.0],
→ [0.534046329052269, 0.534046329052269, 0.0,
→ 0.37997836159100784, 0.0, 0.534046329052269]]
```

1.4.2. Representação por *Word Embeddings*

Diferentemente da representação por *bag-of-words*, *word embeddings* trabalham com **vocabulários** e consideram a **ordem das palavras**. Tal como abordado inicialmente em **Indexação do Vocabulário**, em PLN, um vocabulário contém palavras pré-selecionadas, as quais podem ser obtidas reunindo todas as palavras únicas de um corpus ou mesmo de um vocabulário preexistente. As palavras do vocabulário ficam dispostas em uma espécie de tabela de consulta (uma matriz, chamada de *look-up table*) que contém vetores de *word embeddings*. Como ilustrado na Figura 1.11, uma *word embedding* pode ser encontrada usando o índice da palavra no vocabulário ao procurar pelo mesmo índice na *look-up table*.

Out-of-Vocabulary (OOV). Imagine que é preciso usar o vocabulário ilustrativo da Figura 1.11 para um corpus apenas sobre plantas. Isso é altamente comum no PLN²¹, mas o que acontece caso o nome de uma planta não esteja no vocabulário sobre animais? Para dar conta de palavras desconhecidas, ou seja, que não estão no vocabulário (*out-of-vocabulary* ou OOV), geralmente é reservado um *token* especial UNK (de *unknown*, desconhecido).

²¹Justificativas do porque isso acontece serão explicadas mais a frente no texto.

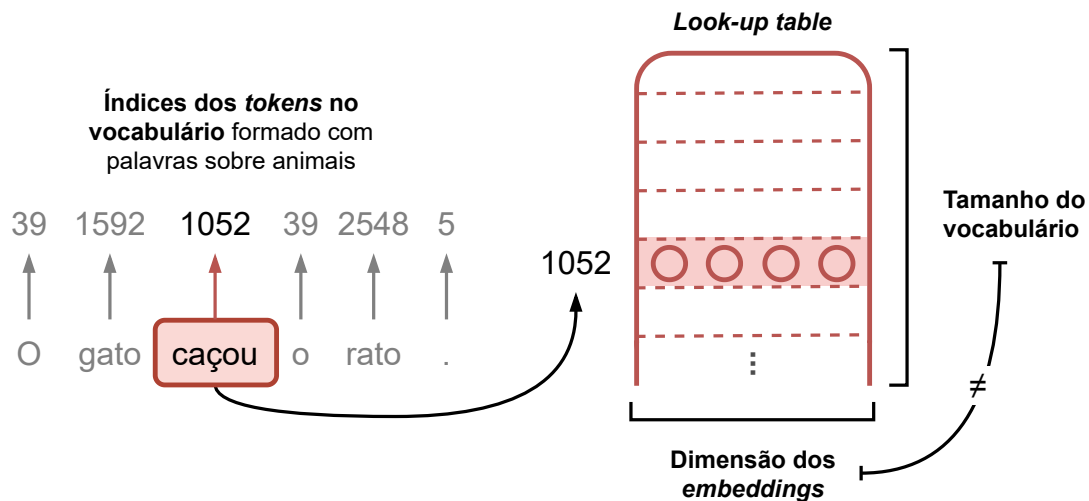


Figura 1.11. Ilustração que representa a relação dos índices de um vocabulário com *word embeddings*. O vocabulário em questão é formado com palavras que possuem nomes, características e comportamentos de animais. Os índices do vocabulário para os *tokens* da frase “O gato caçou o rato.” estão elencados à esquerda, juntamente com a palavra *caçou* e seu índice estando em destaque. À direita, é demonstrado que para o índice 1052, que representa a palavra *caçou*, existe uma *word embedding*. Neste cenário, o tamanho do vocabulário não é igual à dimensão (tamanho dos vetores) das *word embeddings*. Adaptado de [Voita 2020].

Alternativamente, *tokens* desconhecidos podem ser ignorados ou atribuídos para *embeddings* compostos apenas por zeros. Contudo, essa não é uma solução adequada para o problema de OOV [Torregrossa et al. 2021].

Word embeddings como símbolos discretos. A maneira mais simples de obter *word embeddings* é representando palavras como vetores *one-hot*. No entanto, para vocabulários grandes, esses vetores também serão grandes: a dimensionalidade/tamanho do vetor é igual ao tamanho do vocabulário. Isso é indesejável na prática, uma vez que fornece vetores muito esparsos.

Neste curso, a codificação *one-hot* foi usada apenas para demonstração de exemplos simples de como representar palavras. De agora em diante, o foco será nos tipos de representações mais efetivas e bem estabelecidas encontradas na literatura, uma vez que vetores *one-hot* são agnósticos às palavras.

Word embeddings como um espaço vetorial semântico. Partindo do princípio de que palavras formam um espaço estruturado, isto é, que compartilham informações entre si, a hipótese distributiva (*distributional hypothesis*) define que palavras que frequentemente aparecem em contextos semelhantes (que possuem um relacionamento semântico) também possuem significados semelhantes [Voita 2020]. Essa hipótese estabelece que capturar significado e capturar contextos são inerentemente a mesma coisa.

Técnicas de modelos de sequência inserem informações sobre contextos em *word embeddings*, e são capazes de identificar semelhanças entre palavras e mapeá-las em um

espaço vetorial de dimensão finita (espaço Euclidiano) [Chollet 2021]. Nesse espaço, a proximidade entre os vetores podem representar relações semânticas e sintáticas entre palavras de forma linear. Por exemplo, na Figura 1.12, a distribuição das palavras *rainha* e *rei* no espaço é (quase) a mesma que entre *mulher* e *homem*.

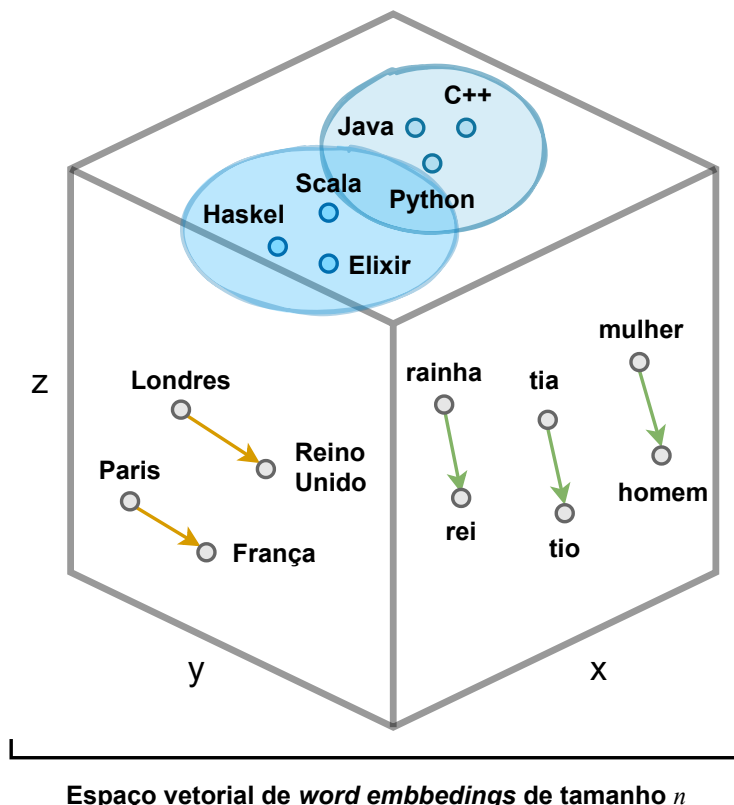


Figura 1.12. *Word embeddings* permitem que o significado seja representado quantitativamente através de um espaço vetorial. Quanto mais semelhante for o significado entre palavras, mais próxima será a representação no espaço, criando na figura, por exemplo, um *cluster* (grupo) de linguagens de *software* (em azul), que por sua vez contém subgrupos de linguagens de Programação Funcional (azul mais escuro) e Programação Orientada a Objetos (azul mais claro). Outra característica das *word embeddings* é que sua distância e direção no espaço mapeiam o significado: neste desenho, os vetores verdes representam o gênero, enquanto que os amarelos representam as relações entre país e capital.

O exemplo *mulher-homem* \approx *rainha-rei* é provavelmente o mais popular, mas também existem muitas outras relações. Na Figura 1.13 estão alguns exemplos de relações similares que formam uma estrutura linear entre *country-capital* (país-capital, diagrama à esquerda) e relações sintáticas (gramaticais, diagrama à direita).

Word embeddings são representações vetoriais de palavras que conseguem exatamente isso: mapeiam a linguagem humana para um espaço geométrico estruturado. A representação obtida são de vetores densos, de baixa dimensionalidade e aprendidos através dos dados. Por fornecerem representações textuais eficientes, modelos que geram *word embeddings* semânticas são atualmente os mais utilizados no PLN. Atente-se apenas

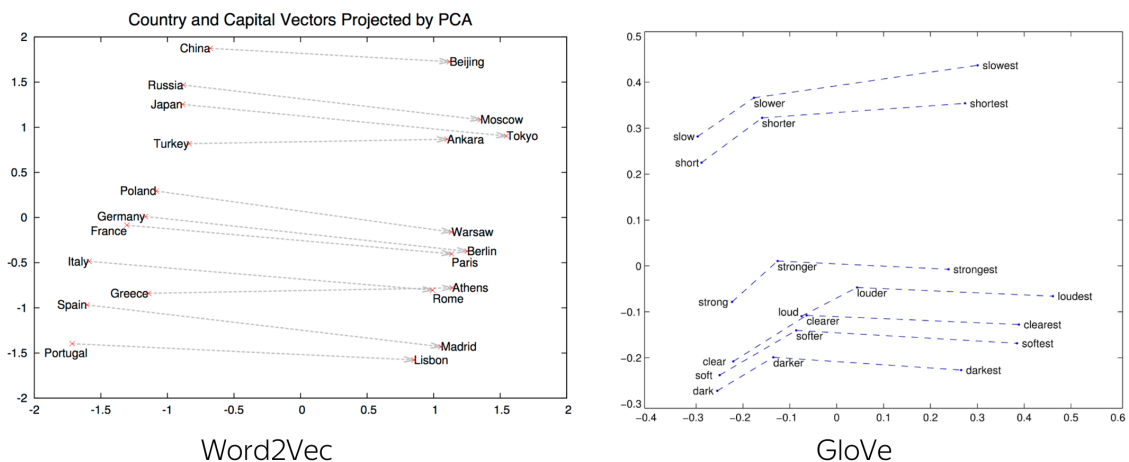


Figura 1.13. Exemplos ilustrativos em 2D de espaços vetoriais que mostram tipos diferentes de relações entre *word embeddings*. À esquerda, pares de palavras *country-capital* (país-capital) semanticamente similares computadas usando a representação do *Word2Vec*. À direita, palavras sintaticamente (variações de uma mesma palavra) similares computadas usando a representação do *GloVe*. Fonte: [Voita 2020].

para o fato de que esses modelos de sequência funcionam melhor quando muitos dados de treinamento estão disponíveis e quando cada texto é relativamente curto.

Os espaços vetoriais formados por *word embeddings* geralmente apresentam milhares de vetores interpretáveis e potencialmente úteis. Para técnicas de *word embeddings* que usam a hipótese distributiva, estas ainda podem ser separadas em dois grandes grupos: *word embeddings* estáticas (ou clássicas), e *word embeddings* contextuais. Antes de entender qual a diferença principal desses dois grupos, saiba que existem duas maneiras de obter *word embeddings*:

- **Aprender *word embeddings* em conjunto com a tarefa principal.** Nessa configuração, os vetores de palavras são iniciados de forma aleatória e, em seguida, aprendidos da mesma forma que são atualizados os pesos de uma rede neural. A Seção 1.6 possui exemplos práticos com problemas reais que demonstram essa forma em específico de gerar *word embeddings*.
- **Carregar *word embeddings* pré-treinadas.** Refere-se aos vetores que são pré-computados usando um modelo diferente do que o usado pela tarefa que se está tentando resolver, como também um corpus diferente com (provavelmente) muitos textos. Os nomes *Word2Vec* e *GloVe* presentes na parte inferior da Figura 1.13 tratam-se de dois modelos prestigiados que geram *word embeddings*. Em particular, vetores pré-treinados serão usados para demonstrarem a diferença entre *word embeddings* estáticas e contextuais nas próximas seções. Por tratar-se de uma forma eficaz e popular de como obter *word embeddings*, esses vetores pré-treinados são uma das principais causas de ocorrerem problemas de palavras OOV em domínios específicos, por conta da divergência existente entre vocabulários.

1.4.3. Representação por *Word Embeddings* Estáticas

Como citado anteriormente, é possível aprender *word embeddings* juntamente com a tarefa de PLN, como também carregar vetores provenientes de um espaço de *word embeddings* pré-treinado. A principal lógica por trás do uso de *word embeddings* pré-treinadas no PLN²² é que o espaço pré-formado é altamente estruturado e compõe-se de aspectos genéricos da estrutura da linguagem. Tais *word embeddings* são geralmente computadas usando estatísticas das ocorrências das palavras (observações sobre quais palavras co-ocorrem em sentenças ou documentos), usando uma variedade de técnicas, algumas envolvendo redes neurais, outras não.

A ideia de um espaço vetorial de *word embeddings* denso e de baixa dimensão, computado de forma não supervisionada, foi inicialmente explorada no início dos anos 2000 [Bengio et al. 2000a]. Embora só começou a popularizar-se em pesquisas e aplicações industriais após o lançamento de um dos mais famosos e bem-sucedidos modelos de *word embeddings*: o algoritmo *Word2Vec*²³ (citado na Figura 1.13), desenvolvido por Tomas Mikolov em 2013 [Mikolov et al. 2013].

Vetores de *word embeddings* pré-treinados estão disponíveis gratuitamente para carregamento e/ou *download* em diversos idiomas. *Word2Vec* é um deles. Outro modelo também renomado e presente na Figura 1.13, chamado de vetores globais para representação de palavras ou apenas *GloVe*²⁴, foi desenvolvido por pesquisadores da Universidade de *Stanford* em 2014 [Pennington et al. 2014].

Os vetores pré-treinados usados nesta seção para abordar assuntos previamente apresentados, como *look-up table* e OOV, além de explicar o que são *word embeddings* estáticas, originaram-se a partir de um outro modelo bem estabelecido: o *fastText*. Normalmente, uma *look-up table* atribui cada palavra para um vetor distinto. Por padrão, essas *word embeddings* não têm nenhuma ideia sobre suas sub-palavras. Por exemplo, a palavra *jogar* não “sabe” que possui relação com *jogador*, *jogando*, *jogado*, dentre outras flexões, pois todas as informações que esses vetores têm são do que aprenderam dos contextos. Uma das possíveis abordagens é compor um vetor de palavras a partir de vetores das suas sub-palavras.

Para isso, o modelo *fastText*²⁵ retorna, além de *word embeddings*, *embeddings* para *n-grams* de caracteres que também estão no vocabulário [Bojanowski et al. 2017]. Fornecer informações sobre sub-palavras possui algumas vantagens: (i) permite que o modelo saiba que diferentes *tokens* podem ser variações da mesma palavra; (ii) ajuda a representar palavras OOV com base em sua ortografia; e (iii) manipula erros ortográficos.

Note no Programa 1.4.6 como vetores pré-treinados com o *fastText* para o Português podem ser carregados facilmente com ajuda da biblioteca *flairNLP*, um *framework* de PLN projetado para facilitar o treinamento e a distribuição de modelos de rotulagem de sequência, classificação de texto e LMs [Akbik et al. 2019]. Inclusive, o *flairNLP* geralmente recomenda o uso de *word embeddings* do *fastText* ou *GloVe*.

²²Mais detalhes são explanados na Seção 1.5.

²³<https://code.google.com/archive/p/word2vec/>

²⁴<https://nlp.stanford.edu/projects/glove>

²⁵<https://fasttext.cc/>

Programa 1.4.6. Inicializando *word embeddings* estáticas pré-treinadas do *fastText* para o Português com a biblioteca *flairNLP*.

```

from flair.embeddings import WordEmbeddings

# Inicializando as word embeddings estáticas pré-treinadas
# 'pt' = Português
ftext_embedding = WordEmbeddings('pt')

print("São embeddings estáticos? =
→ {}".format(ftext_embedding.static_embeddings))
print("Tamanho do vocabulário = {}
→ tokens".format(len(ftext_embedding.vocab)))
print("Dimensão dos embeddings =
→ {}".format(ftext_embedding.embedding_length))
print("Tipos dos embeddings =
→ {}".format(ftext_embedding.embedding_type))
# São embeddings estáticos? = True
# Tamanho do vocabulário = 592107 tokens
# Dimensão dos embeddings (tamanho) = 300
# Tipos dos embeddings = word-level

```

Dentro da variável `ftext_embedding` estão as *word embeddings* geradas a partir de milhões de *tokens* em Português, resultantes do corpora (conjunto de corpus) formado pelos dados do Wikipédia e *Common Crawl*²⁶. O Programa 1.4.6 também possui algumas informações sobre os vetores carregados: tratam-se de *word embeddings* estáticas, com vocabulário extenso, de mais de 50.000 *tokens*, e com vetores de tamanho 300. É comum ver *word embeddings* estáticas de 256 dimensões, 512 ou até 1.024 dimensões ao lidar com vocabulários muito grandes [Chollet 2021]. O vocabulário do `ftext_embedding` pode ser facilmente acessado com o comando `ftext_embedding.vocab`, em que é retornado um dicionário com todos os *tokens*, iniciando do valor 0. Esse vocabulário serve apenas como base, e os valores das chaves (*tokens*) podem ser facilmente editados.

No *flairNLP*, é simples inicializar uma sentença e atribuir os *embeddings* correspondentes para cada palavra. A última linha de código no Programa 1.4.7 retorna um vetor de tamanho 300 que representa a palavra `gato`. O valor dois (2) reflete que `gato` é o segundo *token* da sentença “*O gato caçou o rato.*”. Repare que a sentença não foi padronizada porque o foco aqui é verificar apenas a existência de *word embeddings* e de possíveis termos OOV, e não realizar um procedimento completo para treinamento de um modelo. Deste modo, nenhum dos *tokens* da sentença “*O gato caçou o rato.*” são OOV, ou seja, todos os *tokens* fazem parte do vocabulário do *fastText*.

Mas, afinal, o que são *word embeddings* estáticas? Basicamente, uma ***word embedding* estática** trata-se de uma representação de palavra que não varia entre contextos. Por exemplo, a palavra *bolo* em “*Pesquisei sobre receitas de bolo de laranja*” e “*Há um bolo de coisas sobre a mesa*” tem dois significados diferentes: a primeira indica uma comida e a segunda um amontoado de itens. Porém, uma *word embedding* estática for-

²⁶<https://fasttext.cc/docs/en/crawl-vectors.html>

Programa 1.4.7. Trecho de código em que a frase “O gato caçou o rato.” é tokenizada e o *embedding* da palavra `gato` é acessado para o *fastText*.

```
from flair.data import Sentence

sent = Sentence("O gato caçou o rato.")

sent.tokens
# [Token[0]: "O",
#  Token[1]: "gato",
#  Token[2]: "caçou",
#  Token[3]: "o",
#  Token[4]: "rato",
#  Token[5]: "."]

ftext_embedding.embed(sent)

sent.get_token(token_id=2).embedding
```

neca a mesma representação para a palavra *bolo* em ambas as frases. O termo “estático” refere-se a esta particularidade: um vetor de palavras é sempre o mesmo, e pode ser *estaticamente* pré-computado e armazenado em uma *look-up table*. O Programa 1.4.8 realiza a comparação da palavra `bolo` nos diferentes contextos e demonstra que a *word embedding* é a mesma.

Sentence embeddings. Uma alternativa a limitação mencionada anteriormente é usar *sentence embeddings* [Feng et al. 2020, Cer et al. 2018, Le and Mikolov 2014] (veja alguns exemplos na Figura 1.14). Tais modelos tomam o texto como entrada, computam suas *word embeddings*, e geram uma saída com um único vetor de tamanho fixo. Isso não só ajuda o modelo a entender o contexto, como também a intenção e outras nuances presentes. Uma estratégia comum é usar uma operação de *pooling* para realizar a média das *word embeddings* de cada palavra presente numa sentença. Como nenhuma das tarefas de PLN abordadas neste curso presentes na Seção 1.6 fazem uso de *sentence embeddings*, para saber detalhes de quais são as possíveis técnicas para obtenção desses tipos de vetores, a biblioteca *flairNLP* ilustra e resume algumas delas²⁷.

1.4.4. Representação por *Word Embeddings* Contextuais

Sentence embeddings que derivam de representações fixas possuem alguns fatores que podem afetar sua efetividade, como o tamanho da sentença, a ordem das palavras, a presença de palavras que são dispensáveis ao contexto (como *stopwords*), e possíveis vieses presentes nos dados [Torregrossa et al. 2021]. Uma solução para esse problema e para *word embeddings* estáticas é computar representações que possam ser adaptáveis.

Em particular, ***word embeddings contextuais*** possuem propriedades variáveis e são diretamente obtidas a partir do contexto²⁸. As informações úteis que formam uma

²⁷https://github.com/flairNLP/flair/blob/master/resources/docs/TUTORIAL_5_DOCUMENT_EMBEDDINGS.md

²⁸O contexto de uma palavra geralmente é composto pelas palavras que a cercam.

Programa 1.4.8. Demonstração simples de como *word embeddings* estáticas não variam entre contextos. Primeiramente, são instanciadas duas sentenças que possuem a palavra `bolo`, mas com significados diferentes: “*Pesquisei sobre receitas de bolo de laranja*” e “*Há um bolo de coisas sobre a mesa*”. Os índices para a palavra `bolo` em ambas as sentenças são cinco (5) e três (3), respectivamente. A partir dos índices, os *embeddings* são acessados e comparados, e é demonstrado que os vetores para a palavra `bolo` são os mesmos.

```
sent_1 = Sentence("Pesquisei sobre receitas de bolo de
→ laranja")
sent_2 = Sentence("Há um bolo de coisas sobre a mesa")

ftext_embedding.embed([sent_1, sent_2])
# [Sentence: "Pesquisei sobre receitas de bolo de laranja",
#  Sentence: "Há um bolo de coisas sobre a mesa"]

sent_1.get_token(5), sent_2.get_token(3)
# (Token[4]: "bolo", Token[2]: "bolo")

bolo_emb_sent_1 = sent_1.get_token(token_id=5).embedding
bolo_emb_sent_2 = sent_2.get_token(token_id=3).embedding

print("Os embeddings são os mesmos? {}".format(bolo_emb_sent_1
→ == bolo_emb_sent_2))
print("Índice da palavra 'bolo' =
→ {}".format(ftext_embedding.vocab["bolo"]))
# Os embeddings são os mesmos? True
# Índice da palavra 'bolo' = 12713
```

word embedding contextual, em geral, são extraídas a partir de *Language Models* e lidam com os três problemas principais das *word embeddings* estáticas [Liu et al. 2019]: (i) *word embeddings* estáticas não tratam a polissemia (vide o exemplo anterior com a palavra polissêmica *bolo*); (ii) *word embeddings* contextuais permitem o uso de representação por *n-grams* de palavras; e (iii) *word embeddings* estáticas assumem que o significado de uma palavra depende de contextos similares, o que não é totalmente correto. Uma observação importante é que *word embeddings* contextuais ainda conseguem projetar palavras em um espaço vetorial semântico e finito.

Word embeddings contextuais são obtidas através de modelos pré-treinados (em sua maioria, *Language Models* pré-treinados), e em vez de representar palavras individualmente, representam palavras juntamente com o contexto em que são usadas. Ao treinar um LM, é obtido muito mais do que treinar apenas *word embeddings* clássicas, já que LMs processam não apenas palavras, mas frases, parágrafos, e textos completos, como ilustrado anteriormente na Figura 1.10 no início da seção sobre **Representação Textual**. Para verificar essa característica intrínseca e poderosa dos LMs, a mesma experimentação feita na Seção 1.4.3 é adaptada para uma das *word embeddings* contextuais mais conceituadas hoje: o modelo BERT.

O *Bidirectional Encoder Representations from Transformers* ou apenas BERT, foi introduzido em [Devlin et al. 2018] e, como o próprio nome diz, usa internamente a ar-

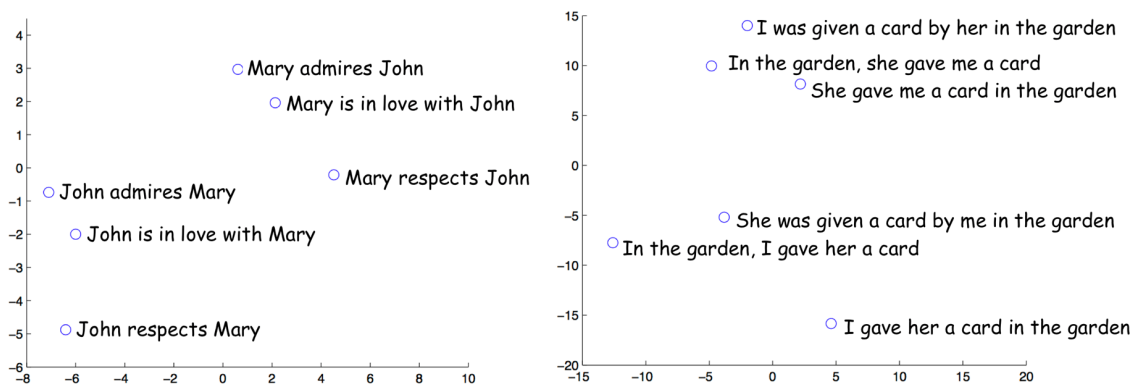


Figura 1.14. Representações das proximidades de *sentence embeddings* com significados semelhantes, mas estruturas diferentes. Fonte: [Voita 2020].

quitetura *Transformer*, um mecanismo de atenção que aprende relações contextuais entre palavras (ou sub-palavras) em textos. Em sua forma simples, o *Transformer* possui uma estrutura de codificador-decodificador. Como o objetivo do BERT é funcionar como um LM, apenas o codificador do *Transformer* é necessário. Ao contrário dos LMs tradicionais, que leem textos sequencialmente (da esquerda para a direita e depois da direita para a esquerda)²⁹, o codificador *Transformer* lê toda a sequência de palavras de uma só vez. Portanto, é considerado bidirecional, embora seja mais correto dizer que é não direcional.

Para contornar alguns outros problemas dos LMs tradicionais, BERT utiliza duas estratégias de treinamento: *Masked Language Modeling* (MLM ou LM Mascarado) e *Next Sentence Prediction* (NSP). Ambos MLM e NSP são treinados juntos, tendo como objetivo minimizar a função de perda. Por conta dessas características, em contraste com outros LMs, BERT atua não só como um substituto para *word embeddings* estáticas mas, também, dependendo da tarefa, pode ser usado como substituto para toda uma arquitetura específica.

Seguindo os passos do experimento feito anteriormente com *word embeddings* estáticas para as sentenças “*Pesquisei sobre receitas de bolo de laranja*” e “*Há um bolo de coisas sobre a mesa*”, em primeiro lugar, os vetores pré-treinados do BERT foram carregados. No Programa 1.4.9, são carregadas as representações do modelo BERT para o Português Brasileiro, ou BERTimbau³⁰ [Souza et al. 2020], com a classe `TransformerWordEmbeddings`³¹.

[Souza et al. 2020] desenvolveram e disponibilizaram dois modelos BERTimbau: um chamado de *base*, com 12 camadas, dimensão de tamanho 768, 12 *attention heads* e 110 milhões de parâmetros; e o outro chamado de *large*, possuindo 24 camadas, dimensão

²⁹Esse processo consiste no tradicional usado por BiRNNs e é importante porque o LM é testado a prever palavras considerando a ordem natural da sentença e depois a sua forma “inversa”. Para esses dois caminhos, as representações obtidas das palavras são concatenadas em uma representação final.

³⁰<https://github.com/neuralmind-ai/portuguese-bert/>

³¹https://github.com/flairNLP/flair/blob/master/resources/docs/embeddings/TRANSFORMER_EMBEDDINGS.md

Programa 1.4.9. Inicializando *word embeddings* contextuais pré-treinadas do BERTimbau *base* para o Português Brasileiro com a biblioteca *flairNLP*.

```
# Observe que a classe carregada é diferente da utilizada para
→ as word embeddings estáticas
from flair.embeddings import TransformerWordEmbeddings

# Inicializando as word embeddings contextuais pré-treinadas
bert_embedding = TransformerWordEmbeddings(
    'neuralmind/bert-base-portuguese-cased')

print("São embeddings estáticos? =
→ {}".format(bert_embedding.static_embeddings))
print("Dimensão dos embeddings =
→ {}".format(bert_embedding.embedding_length))
print("Tipos dos embeddings =
→ {}".format(bert_embedding.embedding_type))
# São embeddings estáticos? = False
# Dimensão dos embeddings = 768
# Tipos dos embeddings = word-level
```

de tamanho 1.024, 16 *attention heads* e 330 milhões de parâmetros³².

Observe que esses dois modelos possuem um número consideravelmente grande de parâmetros. Ao mesmo tempo, exigiram uma enorme quantidade de dados de treinamento e poderosos recursos de computação para garantir resultados promissores para o Português Brasileiro. BERTimbau foi treinado com o brWaC corpus³³, que contém 2,68 bilhões de *tokens* de mais de 3 milhões de textos, e é o maior corpus em Português disponível [Souza et al. 2020]. Além de seu tamanho, o brWaC é composto de documentos inteiros e sua metodologia garante alta diversidade de domínio e qualidade de conteúdo. A quantidade de parâmetros e dados, e o fato de precisarem de processadores avançados para treino do BERT e de outros LMs, tratam-se das principais limitações existentes hoje para obtenção de *word embeddings* contextuais [Neves Oliveira et al. 2022].

As representações carregadas no Programa 1.4.9 foram do modelo BERTimbau *base* (*bert-base-portuguese-cased*³⁴), com dimensão 768. Os vetores resultantes das *word embeddings* dizem respeito a média das saídas das 12 camadas do *base*, conforme explicado na documentação do *flairNLP*. Uma restrição do BERT é que ele processa textos de duas formas distintas, com ou sem capitalização (*cased* ou *uncased*). Dependendo da tarefa, o uso de capitalização pode ser importante, como para o NER, já que entidades (nomes próprios que identificam algo ou alguém) tendem a aparecer com a letra inicial maiúscula. No caso, os modelos BERTimbau são apenas do tipo *cased*. Outra restrição do BERT é que ele apenas processa sentenças de até 512 *tokens*. Isso implica no fato de que ele não processa textos muito extensos.

³²As variações do BERT em *base* e *large* foram introduzidas em seu *paper* [Devlin et al. 2018]. Em resumo, a diferença entre os dois está no número de camadas do codificador, sendo 12 para o *base* e 24 para o *large*.

³³<https://www.inf.ufrgs.br/pln/wiki/index.php?title=BrWaC>

³⁴<https://huggingface.co/neuralmind/bert-base-portuguese-cased>

Após o carregamento, seguindo o fluxo de experimentação, é verificado a *word embedding* da palavra `gato` presente na sentença “*O gato caçou o rato.*” (veja o Programa 1.4.10), sem padronização. Na verdade, não é recomendado a realização de padronização de texto quando se usa *word embeddings* contextuais, já que é necessário saber qualquer tipo de informação que possua dependência de domínio e que seja útil para predição do próximo termo, desde *stopwords* até mesmo pontuações.

Programa 1.4.10. Trecho de código em que a frase “*O gato caçou o rato.*” é tokenizada e o *embedding* da palavra `gato` é acessado para o BERTimbau base.

```
print(sent)
# Sentence: "O gato caçou o rato ."
```



```
bert_embedding.embed(sent)
gato_emb = sent.get_token(token_id=2).embedding
```



```
print("Tamanho do embedding da palavra gato =
↪ {}".format(len(gato_emb)))
# Tamanho do embedding da palavra gato = 768
```

Por fim, o Programa 1.4.11 verifica se as representações para a palavra polissêmica `bolo` são diferentes, o que é retornado que sim: as *word embeddings* são diferentes. Na literatura, os dois vetores distintos da palavra `bolo` são chamados de vetores de conceito (*concept vectors*) porque representam o conceito particular de uma palavra [Torregrossa et al. 2021]. De fato, BERT funciona a nível da palavra (*word-level*, Programa 1.4.9), e pode incorporar informações de sub-palavras, o que implica que lida com palavras OOV. Entretanto, somente se as palavras de contexto forem bem conhecidas, isto é, caso não apareçam raramente no corpus ou corpora usado para treinamento.

Programa 1.4.11. Demonstração simples de como *word embeddings* contextuais variam entre contextos. É demonstrado que os vetores para a palavra polissêmica `bolo` são diferentes para as sentenças “*Pesquisei sobre receitas de bolo de laranja*” e “*Há um bolo de coisas sobre a mesa*”.

```
bert_embedding.embed([sent_1, sent_2])
# [Sentence: "Pesquisei sobre receitas de bolo de laranja",
# Sentence: "Há um bolo de coisas sobre a mesa"]
```



```
sent_1.get_token(5), sent_2.get_token(3)
# (Token[4]: "bolo", Token[2]: "bolo")
```



```
bolo_emb_sent_1 = sent_1.get_token(token_id=5).embedding
bolo_emb_sent_2 = sent_2.get_token(token_id=3).embedding
```



```
print("Os embeddings são os mesmos? {}".format(bolo_emb_sent_1
↪ == bolo_emb_sent_2))
# Os embeddings são os mesmos? False
```

1.5. O Que Faz uma Boa Representação Textual?

O fato de existirem dois grandes grupos principais de *word embeddings* trazem o seguinte questionamento: existe algum algoritmo para criação de *word embeddings* ideal que mapeie perfeitamente a linguagem humana e que possa ser usado para qualquer tarefa do Processamento de Linguagem Natural?

A resposta curta para essa pergunta é *não*. Como existem muitas linguagens diferentes, que não são semelhantes umas às outras, visto que uma língua é o reflexo de uma cultura e um contexto específico, pragmaticamente, o que faz um bom espaço vetorial de *word embeddings* depende muito da tarefa de interesse. A título de exemplo, *word embeddings* que funcionam bem para um modelo de classificação de resenhas de filmes em Inglês podem parecer totalmente diferentes das *word embeddings* ideais para um modelo de classificação de documentos jurídicos em Português, porque a importância de certas relações semânticas varia de tarefa para tarefa.

Isto posto, conclui-se que alguns fatores influenciam intrinsecamente nessa escolha:

- **Contexto ou domínio dos dados.** Realize os seguintes questionamentos: sobre o que é o corpus/corpora? É formado por resenhas de filmes, textos com *fake news*, ou são boletins policiais sobre roubos?
- **Tarefa de PLN.** Que tarefa e, conseqüentemente, que arquitetura será usada? Dependendo da tarefa, *word embeddings* podem ser unicamente essenciais para codificar as informações pertinentes, como *word embeddings* contextuais, ou podem não possuir por si só a capacidade de extrair informações, necessitando de um processo de codificação extra.
- **Quantidade dos dados.** Qual a quantidade de dados disponível? Caso exista uma quantidade razoavelmente grande de dados, é possível usar diretamente o corpus para gerar *word embeddings* específicas do domínio e, em seguida, aplicá-las na tarefa final. Agora, se não há uma quantidade suficiente de dados, é necessário carregar vetores originados previamente (*word embeddings* pré-treinadas), geralmente de um domínio genérico (como artigos de notícias), que é altamente estruturado e que exhibe propriedades úteis ao capturar aspectos genéricos da estrutura da linguagem.

Sobre o último ponto elencado acima, vale destacar que a lógica por trás do uso de *word embeddings* ou modelos pré-treinados no PLN é de que, caso não haja dados disponíveis suficientes que possam ser usados para obter conhecimento linguístico por conta própria sobre os textos de um domínio, espera-se então que as informações necessárias sobre estes textos sejam genéricas. Nesse caso, faz sentido reutilizar o que foi aprendido pelas *word embeddings* pré-treinadas. Porém, deve-se sempre levar em consideração o tipo de corpus ou corpora usado para obter as *word embeddings* pré-treinadas, visto que nem sempre é tão “genérico” assim. Por exemplo, caso o conjunto de dados usado nas *word embeddings* pré-treinadas seja formado inerentemente por artigos de notícias, que de certa forma tendem a ser formais, e pretende-se usar esses vetores em mensagens de

textos que são mais informais, o modelo final pode não performar muito bem, visto que a linguagem é usada de forma distinta em cada um desses cenários.

Também é possível adaptar *word embeddings* pré-treinadas aos dados específicos de tarefas, realizando *fine-tuning* (ajuste) dos *embeddings* treinando-os com toda a rede responsável por resolver a tarefa. Isso pode trazer ganhos de desempenho, embora talvez não enormes [Voita 2020]. Todo esse processo tornou-se comum e possível graças a uma técnica chamada de *Transfer Learning* (Transferência de Aprendizagem). Atualmente, o *Transfer Learning* é a área de PLN mais popular tanto na academia quanto na indústria. A ideia geral por trás dessa técnica é de justamente “transferir conhecimento” de uma tarefa de origem para uma tarefa de interesse/destino [Voita 2020]. Por meio das *word embeddings* pré-treinadas, o conhecimento dos dados de treinamento é “transferido” para um modelo específico de uma certa tarefa.

A Tabela 1.1 descreve os diferentes métodos de representações por *bag-of-words* e *word embeddings* estáticas apresentados até aqui, juntamente com o BERT e outras *word embeddings* contextuais importantes, considerando cinco propriedades principais:

- 1. Considera a ordem das palavras.** Como visto na seção sobre **Representação por Bag-of-Words**, o modelo de *bag-of-bigrams* é bem limitado em relação a ordem das palavras num texto. A auto-atenção é um mecanismo de processamento de conjuntos e foca nas relações entre pares de elementos de sequência, assim, não sabe quais elementos ocorrem no início, no fim ou no meio de uma sequência. Apesar da auto-atenção fazer parte da composição de *Transformers*, e de outros modelos que usam tal arquitetura, como BERT e GPT-3 [Brown et al. 2020], todos estes consideram a ordem das palavras.
- 2. Lida com palavras OOV.** O símbolo — destaca os métodos que não tratam OOV completamente. De fato, ELMo e BERT tratam palavras OOV apenas se os *tokens* em questão não forem raros.
- 3. É contextual.** Propriedade que significa que a representação de uma palavra é obtida olhando para seu contexto, ou seja, quando é uma *word embedding* contextual.
- 4. Possui treinamento auto-supervisionado.** Comumente, a tarefa usada para treinar uma *word embedding* é auto-supervisionada. Isso significa que uma tarefa supervisionada é extraída automaticamente de um conjunto de dados não rotulado.

1.6. Além da Representação Textual: Problemas de Aprendizagem Profunda em PLN

Na prática, todas as ferramentas necessárias para lidar com a maioria das tarefas do Processamento de Linguagem Natural foram apresentadas. Contudo, elas não foram vistas em ação. Nesta seção, a experiência é aprofundada a partir da abordagem de problemas mais tradicionais e bem-sucedidos do PLN, apresentados a seguir.

Tabela 1.1. Propriedades alto nível de várias técnicas de representações textuais. Adaptado de [Torregrossa et al. 2021]. Significado dos símbolos: (✓), a propriedade é aplicável à técnica; (—), a propriedade possui certas limitações; (×), a propriedade não é aplicável à técnica.

<i>Técnica de word embedding</i>	Considera a ordem das palavras	Lida com palavras OOV	É contextual	Possui treinamento auto-supervisionado
<i>Bag-of-unigrams</i>	×	×	×	×
<i>Bag-of-bigrams</i>	—	×	×	×
GloVe [Pennington et al. 2014]	✓	×	×	✓
Word2Vec [Mikolov et al. 2013]	✓	×	×	✓
<i>fastText</i> [Bojanowski et al. 2017]	✓	✓	×	✓
ELMo [Neumann et al. 2018]	✓	—	✓	✓
<i>Flair</i> [Akbik et al. 2018]	✓	✓	✓	✓
BERT [Devlin et al. 2018]	✓	—	✓	✓
GPT-3 [Brown et al. 2020]	✓	—	✓	✓

1.6.1. Pipeline de PLN

O codificador-decodificador é o paradigma da modelagem padrão para tarefas de *sequence-to-sequence*, conforme explicado brevemente na Seção 1.2.1. Um modelo *sequence-to-sequence* usa uma sequência como entrada (geralmente frases ou parágrafos) e a transforma em uma sequência completamente diferente. Essa é a tarefa central de muitas das aplicações mais bem-sucedidas do PLN, como Tradução Automática, Sumarização de Sentenças, *chatbots*, Geração de Textos, etc. A estrutura dessa modelagem consiste em dois componentes principais:

- **Codificador**, que constrói uma representação a partir da sequência de entrada e a passa ao decodificador. Depois que os *tokens* de entrada são mapeados em um espaço vetorial (*word embeddings* ou outros tipos de *embeddings*), os vetores passam por uma camada de codificador para compactar todas as informações em um vetor de comprimento fixo. Normalmente, um codificador é feito de redes do tipo RNN, como LSTMs.
- **Decodificador** usa a representação precedente do codificador para gerar a sequência de saída. A camada do decodificador produz a sequência de saída de *tokens* com base no vetor codificado. Essa camada também é geralmente construída com arquiteturas RNN.

Mesmo que uma tarefa não seja um problema *sequence-to-sequence*, a modelagem ainda é relevante e pode ser utilizada para resolver outros problemas. Por exemplo, a tarefa de NER não se encaixa nesse tipo de abordagem, mas a estrutura de codificador-decodificador é usada para resolvê-la, sendo a saída uma sequência de entidades para cada *token* da sentença de entrada.

A Figura 1.15 demonstra como um *pipeline* de um modelo de Aprendizagem Profunda pode ser construído para o PLN, e deve ser lida de baixo para cima. Inicialmente, os textos de entrada (frases, parágrafos ou documentos completos) precisam ser convertidos em alguma abordagem de representação textual. Essa etapa no *pipeline* é tradicionalmente conhecida como camada de *embedding* (*embedding layer*), e a maioria das aplicações modernas de PLN iniciam com esta camada. Em seguida, a representação em forma

de vetores é passada ao codificador. Boa parte das tarefas simples de PLN terminam aí, como Classificação de Texto, uma vez que, depois do codificador, não é necessário entender mais nada sobre a sequência dos *tokens* de entrada, apenas converter a saída do codificador para um valor fixo (como 0 ou 1). O decodificador torna-se necessário caso seja preciso literalmente decodificar as representações vetoriais aprendidas pelo codificador em textos ou classes.

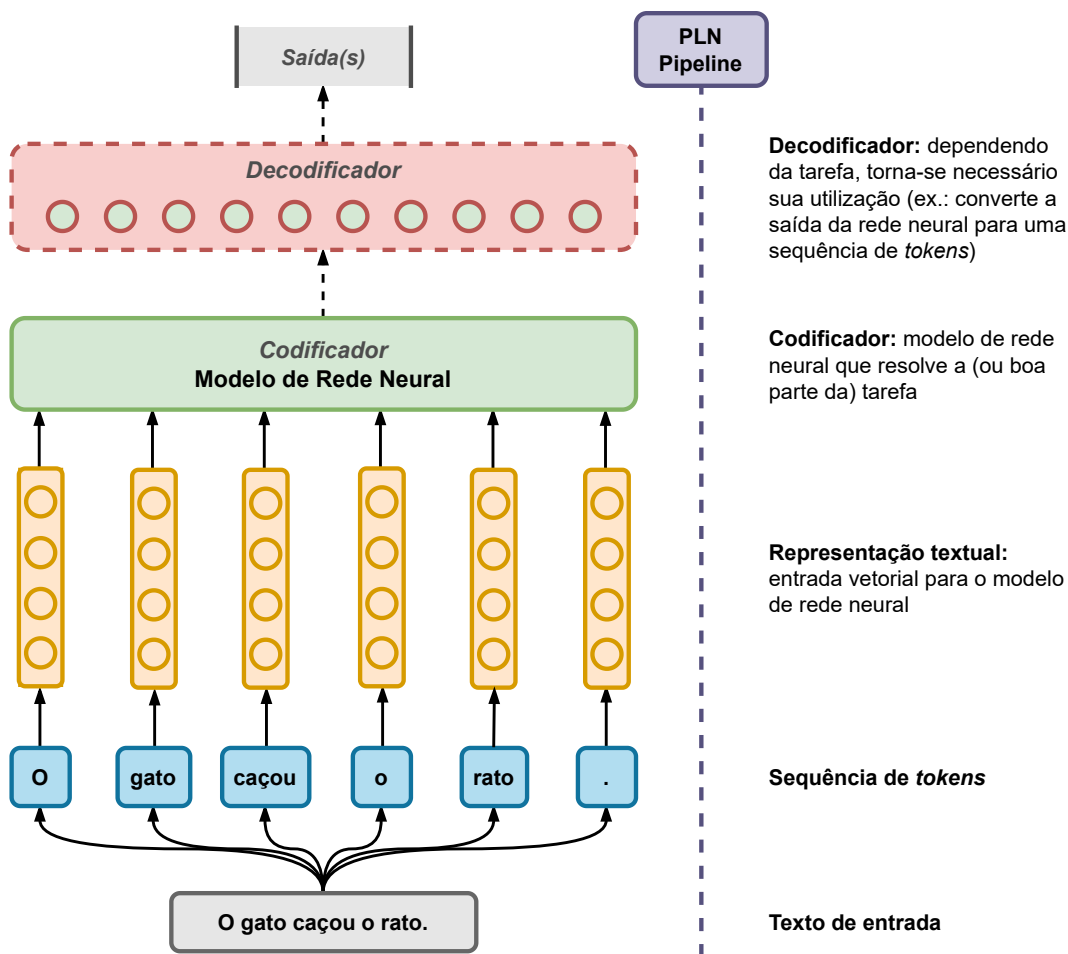


Figura 1.15. Pipeline geral de um modelo de Aprendizagem Profunda para o PLN.

Imagine o seguinte: o codificador recebe vetores que possuem internamente informações sobre as relações das palavras e usa-os por intermédio de uma rede neural para entender mais ainda sobre o seu significado. A saída desse codificador ainda serão vetores, e bem mais complexos. Então, como decodificar essa complexidade em algo que não seja vetorial, tal como aquele texto de entrada, e que ainda resolva a tarefa? Por isso que o decodificador é necessário. Um detalhe importante é que um decodificador precisa ser uma rede neural, diferente ou semelhante da usada pelo codificador, ou um modelo de Aprendizagem de Máquina. Isso explica porque, dependendo da tarefa de Classificação de Texto, um decodificador não torna-se necessário, já que não é fundamental o uso de

uma rede neural para converter vetores para valores inteiros.

Além da não utilização obrigatória do decodificador para determinadas tarefas, nem sempre essa estrutura é necessária quando a representação textual escolhida (vide o uso do BERT, explanado na Seção 1.4.4) é autossuficiente para conseguir resolver o problema. Porém, o *pipeline* ainda é válido para caso a representação textual aplicada não consiga obter informações específicas e suficientes sobre os textos de entrada. O *pipeline* apresentado é usado como base para resolução das tarefas de **Classificação de Textos** e **Sumarização de Sentenças**, descritas nas seções a seguir. Em cada tarefa, é destacado o que muda no *pipeline* geral, em relação aos tipos de *embeddings*, a rede que será usada no codificador, se possui um decodificador, etc.

1.6.2. Classificação de Textos

De certa forma, Classificação de Textos é uma das tarefas menos complexas do PLN: dado um texto, existe uma classe que pode ser atribuída a ele. Existem diversos exemplos para esse problema, como dizer se uma opinião sobre um produto é positiva ou negativa, identificar o tema de uma notícia, categorizar uma notícia como verdadeira ou falsa (*fake news*), etc.

Uma das aplicações mais recorrentes para o problema de Classificação de Textos é a Análise de Sentimentos, que busca identificar conteúdo de opinião, e determinar o sentimento, percepção ou atitude do público em relação ao alvo desta opinião. O IMDb (*Internet Movie Database*)³⁵ é um banco de dados *online* muito utilizado para o treinamento de modelos que resolvem esse tipo de problema. Um dos conjuntos de dados existentes baseados no IMDb é composto por resenhas de filmes em formato de texto e o sentimento associado à uma determinada resenha como classe (1 se for positivo e 0 se negativo).

A Figura 1.16 apresenta uma arquitetura simples para o treinamento de um modelo para a tarefa de Análise de Sentimentos, seguindo o *pipeline* básico da Figura 1.15. A sequência de *tokens* é codificada para *bag-of-bigrams*, em conformidade com o que foi explanado na Seção 1.4.2 e recomendado por [Chollet 2021].

Os *embeddings* gerados com *bag-of-bigrams* são utilizados por uma rede BiLSTM que fará o papel de codificador da resenha inicial. Uma vez que o modelo tenha sido treinado, uma etapa de decodificação é necessária para gerar a saída condizente com o problema inicial: como a definição da tarefa descreve que o sentimento é binário (positivo ou negativo), um *Perceptron* convencional é utilizado; caso a saída definida pertencesse a um conjunto finito de possibilidades, uma MLP poderia ser colocada como decodificador.

Todo o processo de leitura e padronização do conjunto de dados, montagem em código do modelo ilustrado na Figura 1.16, treinamento e avaliação, está documentado e descrito no repositório do curso <https://github.com/barbaraneves/nlp-course-jai22/tree/main/sec06-problemas-pln/classific-de-texto>.

³⁵<https://keras.io/api/datasets/imdb/>

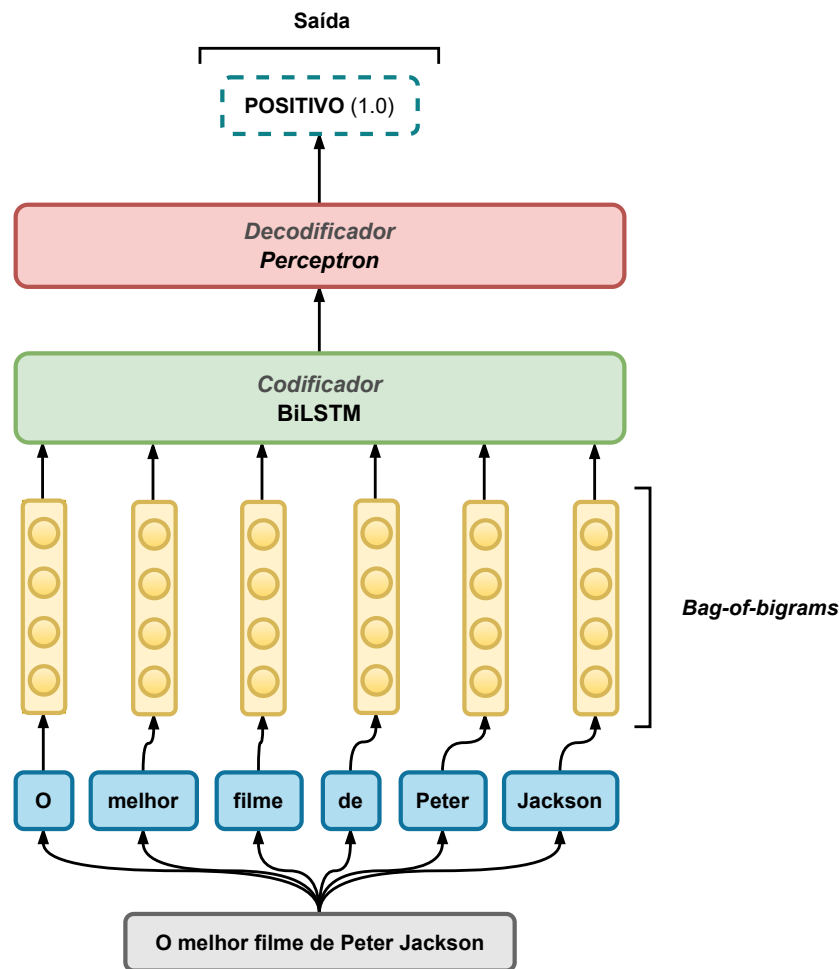


Figura 1.16. Modelo *baseline* para Classificação Binária de Textos. A sentença exemplo “O melhor filme de Peter Jackson” é classificada como uma opinião positiva (valor 1.0).

1.6.3. Sumarização de Sentenças

A tarefa de Sumarização de Sentenças [Wang et al. 2017] consiste em criar uma versão reduzida de uma determinada sentença original (com relação ao número de palavras) fornecida como entrada, de tal forma que a compressão gerada obedeça as duas seguintes regras: a sentença gerada deve manter todas as principais informações do texto original e deve ser gramaticamente correta. Chama-se de compressão válida as sentenças geradas que obedecem a essas regras.

Existem dois tipos distintos de sumarização de sentenças: extrativos ou abstrativos [Tas and Kiyani 2007]. A sumarização extrativa permite apenas a operação de remoção de palavras, sem alterar a ordem das palavras na compressão resultante com relação à sentença original. Por exemplo, a frase “Eu amo muito chocolate branco.” pode ser comprimida para “Eu amo chocolate”. Nessa compressão, as palavras *muito* e *branco* foram removidas e a posição das palavras restantes com relação a sentença original foi mantida. A sumarização abstrativa gera compressões por meio da geração de linguagem natural a

partir da sentença original, sem restrições com relação à remoção, ou reordenamento de palavras ou inclusão de novos termos. Por exemplo, a manchete “*Milhares de caxemires cantando slogans pró-Paquistão no domingo participaram de um comício para receber de volta um líder separatista linha-dura que passou por tratamento de câncer em Mumbai.*” pode ter como compressão a sentença “*Milhares participam de comício para linha-dura da Caxemira*”.

Com o crescimento do poder computacional nos últimos anos, o volume de textos disponíveis na *Web* e a popularização das redes neurais, diversos modelos de Aprendizagem Profunda foram propostos para resolver a tarefa de Sumarização de Sentenças de forma satisfatória. A Figura 1.17 apresenta uma das possíveis arquiteturas [Wang et al. 2017] que podem ser construídas para a tarefa de compressão de sentenças e que seguem o *pipeline* delineado na Figura 1.15. O conjunto de dados utilizado para resolução da tarefa de compressão é composto por 10 mil pares sentença/compressão extraídos do serviço *Google News* [Filippova et al. 2015].

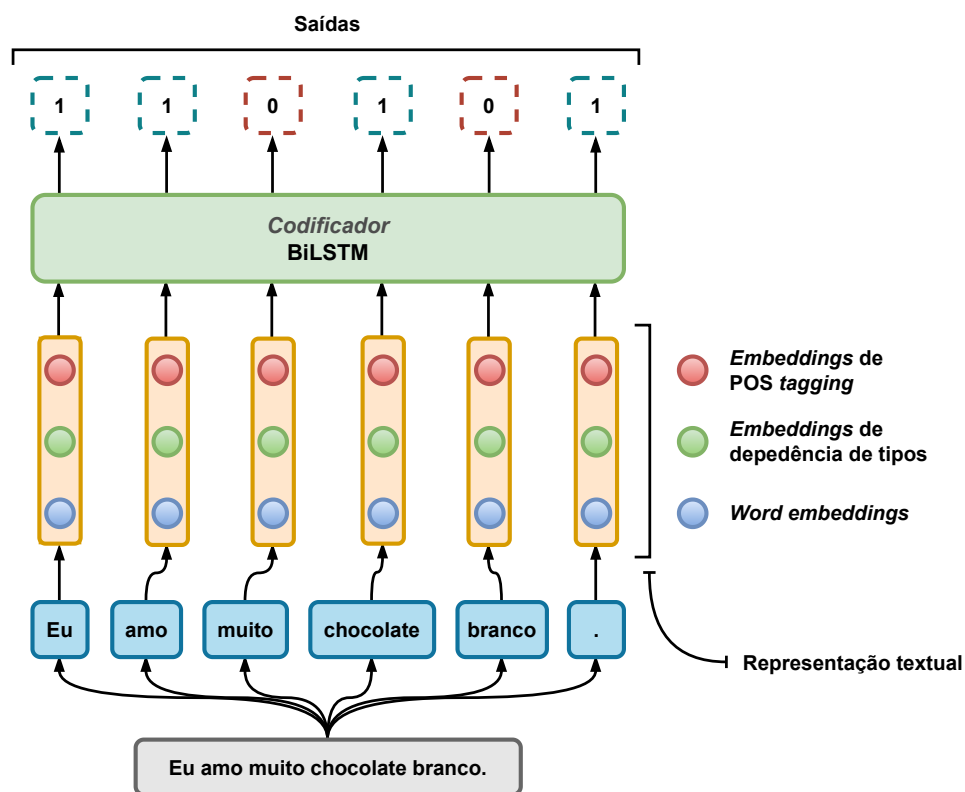


Figura 1.17. Arquitetura para o problema de Sumarização de Sentenças. É verificado como a sentença exemplo “*Eu amo muito chocolate branco.*” pode ser resumida: o valor 1 indica que o *token* deve permanecer, e o valor 0 indica que o *token* deve ser omitido da sumarização. Assim, “*Eu amo chocolate.*” trata-se da sentença resumida final.

Realizando a leitura de baixo para cima da Figura 1.17, tem-se a sentença “*Eu amo muito chocolate branco.*” que será resumida. A sentença passará por um processo de padronização (segundo o fluxo apresentado na Seção 1.3), gerando uma sequência de *to-*

kens. Cada um dos *tokens* gerados é representado vetorialmente para serem fornecidos ao codificador por uma pilha de *embeddings*, composta por: *word embeddings*, *embeddings* de POS *tagging*, e *embeddings* de tipos de dependências das palavras. O codificador usa uma BiLSTM padrão para processar os vetores de *embeddings* sequencialmente nas duas direções da sentença. Também é responsabilidade do codificador predizer a classe de cada um dos *tokens* da sequência: 0, para caso o *token* deva ser omitido da sumarização, e 1, caso deva permanecer.

Observe que esse problema de sumarização não necessita de um decodificador, apesar de retornar uma sequência de números. Todo o processo de leitura e padronização do conjunto de dados, montagem em código do modelo ilustrado na Figura 1.17, treinamento e avaliação, está documentado e descrito no repositório do curso <https://github.com/barbaraneves/nlp-course-jai22/tree/main/sec06-problemas-pln/sumarizacao-de-sent>.

1.7. Desafios de Pesquisa

Enquanto que o Processamento de Linguagem Natural tenta melhorar continuamente sua capacidade de compreender computacionalmente palavras e sentenças, a linguagem humana é imensamente complexa, fluida e inconsistente, apresentando grandes obstáculos.

Por ser uma área relativamente jovem, há muito espaço dentro do PLN para engenheiros e empresas enfrentarem os inúmeros problemas não resolvidos que surgem com a implantação destes sistemas. Ao longo deste curso foi levantado que na *Web* existe uma espécie de sobrecarga de informações, que representa vastos conjuntos de dados. Porém, isso não ocorre apenas na *Web*.

Setor público. A organização deste setor também possui uma geração de dados textuais contínua. Nos últimos três anos, a Universidade Federal do Ceará por meio do *Insight Data Science Lab* trabalhou em parceria com alguns departamentos de Segurança Pública no Brasil para construir e implantar soluções para problemas relacionados ao PLN. Muitos desses problemas envolviam a análise e entendimento de Boletins de Ocorrência (BOs). A frequência de geração desse tipo de texto, principalmente em locais com alta ocorrência de crimes, torna inviável que profissionais desse domínio leiam e categorizem manualmente todos os documentos relatados.

Os projetos de Pesquisa e Desenvolvimento em parceria com alguns departamentos de Segurança Pública envolviam: (i) tarefa de Reconhecimento de Entidade Nomeada (NER), para extração das informações relevantes das narrativas de BOs, como vítimas, criminosos e testemunhas; (ii) problema de Classificação de Textos a partir das narrativas dos BOs, para categorizar em um tipo de crime, como furto, fraude ou roubo; (iii) e, dado um BO, como identificar outros que apresentem narrativas semelhantes na base de dados da Segurança Pública, de forma que sejam identificados crimes com o mesmo *modus operandi*. [Neves Oliveira et al. 2022, Zschornack R. S. et al. 2020, da Silva et al. 2019] são alguns dos trabalhos publicados nessa temática.

Covid-19. A pandemia da Covid-19 exigiu dos governos medidas eficientes e ágeis para mitigar os efeitos causados pela doença. O Plantão Coronavírus foi uma das soluções desenvolvidas e disponibilizadas para a população no estado do Ceará. A plataforma

possui tecnologia de *chatbot*, permitindo que o usuário mantenha um bate-papo com um sistema de Inteligência Artificial, podendo ser redirecionado para um atendimento virtual com um humano (médico ou enfermeiro).

Ainda voltado para o setor público, o modelo *Sintomatic* [Da Silva et al. 2020] foi desenvolvido para processar textos em linguagem natural provenientes do Plantão Coronavírus e detectando sintomas. Devido a possíveis mutações do vírus e consequente aparecimento de novos sintomas, o *Sintomatic* proporcionou avanços para entender melhor a doença no Ceará. Por possuir a capacidade de reconhecer novos padrões, o modelo identificou uma alta frequência de comportamentos psicológicos alterados, como ansiedade, angústia e tristeza, em usuários positivos ou não para a Covid-19. A ferramenta mostrou, portanto, a necessidade do estado ampliar o atendimento através do canal para o cuidado com a saúde mental da população.

Impacto social. O caso de sucesso mencionado anteriormente sobre o Plantão Coronavírus e o *Sintomatic* foi publicado no livro *A Era dos Dados no Setor Público*³⁶, uma parceria da Universidade Federal do Ceará, Íris - Laboratório de Inovação Dados do Governo do Estado do Ceará, *Social Good Brasil* e *AWS Institute*. Outras publicações nessa temática foram [Júnior et al. 2021] e [Mendes de Melo et al. 2021].

Outros desafios. Independente da origem e temática dos textos produzidos, a extração de conhecimento em textos traz vários desafios, tais como: o tempo de desenvolvimento necessário para criação de um sistema de PLN; modelos que devem ser capazes de usar o contexto das sentenças para compreender ambiguidades; dependendo do idioma, modelos de PLN podem ser completamente diferentes; como identificar vieses presentes em textos; e, como tratar palavras que possuem significados diferentes dependendo do contexto.

Esses desafios têm sido amplamente investigados pela comunidade em geral de PLN.

1.8. Considerações Finais

O presente capítulo apresentou como técnicas de Aprendizagem Profunda podem ser utilizadas para resolver diversos tipos de tarefas de Processamento de Linguagem Natural. As seções apresentaram inicialmente os fundamentos necessários sobre Aprendizagem Profunda e como os textos utilizados como fontes de dados devem ser processados para que possa isolar suas características mais importantes. Em seguida, foram mostradas as diversas formas de representar, vetorialmente, textos e suas palavras. Por fim, foram apresentadas duas arquiteturas para resolver os problemas de Classificação de Textos e Sumarização de Sentenças baseadas em um *pipeline* genérico que tarefas de PLN tendem a seguir.

Dada a expressividade da área de Processamento de Linguagem Natural, vários assuntos deixaram de ser abordados neste capítulo. Um dos aspectos que muitos pesquisadores vêm investigando é o treinamento de modelos para tarefas de PLN em domínios específicos e que possuem uma pequena quantidade de dados anotados. Para isso, técnicas de *Transfer Learning* e *Meta-Learning* [Hospedales et al. 2020] têm sido desenvolvidas.

³⁶<https://irislab.ce.gov.br/wp-content/uploads/2021/09/LIVRO-DIGITAL-A-Era-dos-Dados-para-o-Setor-Pu%CC%81blico.pdf>

Algumas tarefas importantes também não foram abordadas, como Tradução de Textos, Reconhecimento de Entidades Nomeadas, e suas resoluções com arquiteturas mais modernas de representações textuais, como BERT, ELMo, e *Flair*.

Código do curso. Todos os códigos e ilustrações deste capítulo estão disponíveis no repositório <https://github.com/barbaraneves/nlp-course-jai22>.

Agradecimentos

Este trabalho foi apoiado pela Fundação Cearense de Apoio ao Desenvolvimento Científico e Tecnológico (FUNCAP), projeto Plataforma de Big Data para Acelerar a Transformação Digital do Estado do Ceará sob o número 04772551/2020 e Plataforma Cidadã sob o número 04772314/2020.

Referências

- [Akbik et al. 2019] Akbik, A., Bergmann, T., Blythe, D., Rasul, K., Schweter, S., and Vollgraf, R. (2019). FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.
- [Akbik et al. 2018] Akbik, A., Blythe, D., and Vollgraf, R. (2018). Contextual String Embeddings for Sequence Labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.
- [Bahdanau et al. 2014] Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- [Bengio et al. 2000a] Bengio, Y., Ducharme, R., and Vincent, P. (2000a). A Neural Probabilistic Language Model. *Advances in Neural Information Processing Systems*, 13.
- [Bengio et al. 2000b] Bengio, Y., Ducharme, R., and Vincent, P. (2000b). A neural probabilistic language model. *Advances in Neural Information Processing Systems*, 13.
- [Bojanowski et al. 2017] Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the association for computational linguistics*, 5:135–146.
- [Brown et al. 2020] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language Models are Few-Shot Learners. *Advances in neural information processing systems*, 33:1877–1901.
- [Cer et al. 2018] Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Céspedes, M., Yuan, S., Tar, C., et al. (2018). Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.
- [Chollet 2021] Chollet, F. (2021). *Deep Learning with Python*. Simon and Schuster.

- [Da Silva et al. 2020] Da Silva, T. L. C., Ferreira, M. G. F., Magalhaes, R. P., De Macêdo, J. A. F., and da Silva Araújo, N. (2020). Rastreador de sintomas da COVID19. In *SBBD*.
- [da Silva et al. 2019] da Silva, T. L. C., Magalhães, R. P., de Macêdo, J. A., Araújo, D., Araújo, N., de Melo, V., Olímpio, P., Rego, P. A., and Neto, A. V. L. (2019). Improving Named Entity Recognition using Deep Learning with Human in the Loop. In *EDBT*, pages 594–597.
- [Dai and Le 2015] Dai, A. M. and Le, Q. V. (2015). Semi-Supervised Sequence Learning. *Advances in neural information processing systems*, 28.
- [Devlin et al. 2018] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*.
- [Feng et al. 2020] Feng, F., Yang, Y., Cer, D., Arivazhagan, N., and Wang, W. (2020). Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.
- [Filippova et al. 2015] Filippova, K., Alfonseca, E., Colmenares, C. A., Kaiser, Ł., and Vinyals, O. (2015). Sentence compression by deletion with lstms. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, pages 360–368.
- [Géron 2019] Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. "O'Reilly Media, Inc."
- [Hochreiter and Schmidhuber 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.
- [Hospedales et al. 2020] Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. (2020). Meta-Learning in Neural Networks: A Survey. *arXiv preprint arXiv:2004.05439*.
- [Júnior et al. 2021] Júnior, V. O. D. S., Branco, J. A. C., De Oliveira, M. A., Da Silva, T. L. C., Cruz, L. A., and Magalhaes, R. P. (2021). A Natural Language Understanding Model COVID-19 based for chatbots. In *2021 IEEE 21st International Conference on Bioinformatics and Bioengineering (BIBE)*, pages 1–7. IEEE.
- [Le and Mikolov 2014] Le, Q. and Mikolov, T. (2014). Distributed representations of sentences and documents. In *ICML*, pages 1188–1196. PMLR.
- [Liu et al. 2019] Liu, N. F., Gardner, M., Belinkov, Y., Peters, M. E., and Smith, N. A. (2019). Linguistic knowledge and transferability of contextual representations. *arXiv preprint arXiv:1903.08855*.
- [Louis 2020] Louis, A. (2020). A Brief History of Natural Language Processing — Part 2.

- [McCulloch and Pitts 1943] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.
- [Mendes de Melo et al. 2021] Mendes de Melo, S., Lima Férrer de Almeida, A., Almada Cruz, L., and Linhares Coelho da Silva, T. (2021). A chat recommender system for covid-19 support based in textual sentence embeddings. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 248–252.
- [Mikolov et al. 2013] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*.
- [Neumann et al. 2018] Neumann, M. P. M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *arXiv preprint arXiv:1802.05365*.
- [Neves Oliveira et al. 2022] Neves Oliveira, B. S., Fernandes de Oliveira, A., Monteiro de Lira, V., Linhares Coelho da Silva, T., and Fernandes de Macêdo, J. A. (2022). Held: Hierarchical entity-label disambiguation in named entity recognition task using deep learning. *Intelligent Data Analysis*, 26(3):637–657.
- [Pennington et al. 2014] Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.
- [Rosenblatt 1958] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.
- [Rumelhart et al. 1985] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science.
- [Souza et al. 2020] Souza, F., Nogueira, R., and Lotufo, R. (2020). BERTimbau: Pre-trained BERT Models for Brazilian Portuguese. In *Brazilian Conference on Intelligent Systems*, pages 403–417. Springer.
- [Sutskever et al. 2014a] Sutskever, I., Vinyals, O., and Le, Q. V. (2014a). Sequence to Sequence Learning with Neural Networks. *Advances in neural information processing systems*, 27.
- [Sutskever et al. 2014b] Sutskever, I., Vinyals, O., and Le, Q. V. (2014b). Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.
- [Tas and Kiyani 2007] Tas, O. and Kiyani, F. (2007). A survey automatic text summarization. *PressAcademia Procedia*, 5(1):205–213.
- [Torregrossa et al. 2021] Torregrossa, F., Allesiardo, R., Claveau, V., Kooli, N., and Gravier, G. (2021). A survey on training and evaluation of word embeddings. *International Journal of Data Science and Analytics*, 11(2):85–103.

- [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. In *Advances in neural information processing systems*, pages 5998–6008.
- [Voita 2020] Voita, E. (2020). NLP Course For You.
- [Wang et al. 2017] Wang, L., Jiang, J., Chieu, H. L., Ong, C. H., Song, D., and Liao, L. (2017). Can syntax help? improving an lstm-based sentence compression model for new domains. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1385–1393.
- [Webster and Kit 1992] Webster, J. J. and Kit, C. (1992). Tokenization as the initial phase in NLP. In *COLING 1992 Volume 4: The 14th International Conference on Computational Linguistics*.
- [Zhang et al. 2018] Zhang, L., Wang, S., and Liu, B. (2018). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.
- [Zschornack R. S. et al. 2020] Zschornack R. S., F., Linhares Coelho da Silva, T., and Fernandes de Macêdo, J. A. (2020). Aspect term extraction using deep learning model with minimal feature engineering. In *International Conference on Advanced Information Systems Engineering*, pages 185–198. Springer.

Capítulo

2

Estratégias ágeis aplicadas à projetos de PD&I: da teoria à prática

Andre L L Aquino, Givanildo Nascimento-Jr., Fabiane Queiroz

Abstract

Research projects with collaboration between research organizations and the industry require new project management techniques that follow the evolution of research activities and how these activities meet the industry's innovation needs. The industry sector has already incorporated agile methodologies in its development processes in different contexts and competitive situations. However, scientific organizations still remain in their old ways of working, following research protocols, with little attention to the collaborative challenges involved in their processes. Recent studies assess how these methodologies contribute to collaborative management in scientific research processes. In this sense, we present the main challenges, concepts, and case studies of the adoption of agile strategies, in particular the framework Scrum, in collaborative and multidisciplinary research focused on technological innovation.

Resumo

Projetos de PD&I com colaboração entre organizações de pesquisa e a indústria, requerem técnicas de gerenciamento de projetos que acompanhem a evolução das atividades de pesquisa e o quanto estas atividades estão atendendo às necessidades de inovação do mercado. O setor produtivo, em contextos e situações competitivas totalmente diferentes, já incorporaram estratégias ágeis em seus processos de desenvolvimento. Contudo, organizações científicas ainda permanecem em suas antigas formas de trabalhar, seguindo protocolos de pesquisa, com pouca atenção para os desafios colaborativos envolvidos nos seus processos. Estudos recentes avaliam como a aplicação de estratégias ágeis contribui para a gestão colaborativa e coordenação de tarefas em processos de pesquisa científica. Neste sentido, apresentaremos os principais desafios, conceitos, relatos e estudos de caso da adoção de estratégias ágeis, em particular o framework Scrum, em pesquisa científica voltada para produção de inovação tecnológica.

2.1. Introdução

O Manual de Frascati [OECD 2015] é a principal referência para estudos, análises, levantamentos e comparações de competitividade entre empresas e países no que se refere a atividades de Pesquisa e Desenvolvimento experimental, frequentemente referidas como atividades de P&D. De acordo com o manual, estas atividades são definidas pelo trabalho criativo empregado de forma sistemática, com o objetivo de aumentar o volume de conhecimento do homem, da cultura e da sociedade.

O propósito da P&D não pode ser alcançado através da simples aplicação direta de um conhecimento previamente produzido. É necessário originar/disseminar conhecimento. A P&D é conduzida de forma sistemática, por meio de metodologias bem definidas, como o método científico, afim de gerar novas aplicações. Caso qualquer um desses fatores sejam ignorados, a atividade desenvolvida pode comprometer seu objetivo quanto a proposição de soluções de P&D.

O termo P&D abrange atividades de pesquisa básica e/ou aplicada e desenvolvimento experimental. A pesquisa básica é um trabalho experimental ou teórico realizado principalmente para adquirir novos conhecimentos sobre os fundamentos subjacentes dos fenômenos e fatos observáveis, sem qualquer aplicação ou uso particular em vista. A pesquisa aplicada também é uma investigação original empreendida com o objetivo de adquirir novos conhecimentos. No entanto, é dirigido principalmente para um fim ou objetivo prático específico. O desenvolvimento experimental é um trabalho sistemático, com base no conhecimento existente obtido em pesquisa e/ou experiência prática, que é direcionado para a produção de novos materiais, produtos ou dispositivos, para a instalação de novos processos, sistemas e serviços, ou para melhorar substancialmente aqueles já produzidos ou instalados.

Podemos conceituar a atividade de inovação tecnológica como a concepção de novo produto ou processo de fabricação, bem como a agregação de novas funcionalidades ou características ao produto ou processo, que implique em melhorias incrementais e efetivo ganho de qualidade ou produtividade, resultando em uma maior competitividade no setor produtivo. Segundo o manual de Frascati [OECD 2015], atividades de Pesquisa, Desenvolvimento e Inovação (PD&I) estão inseridas em um contexto de inovação tecnológica que incorpora, além de atividades de P&D, ações no âmbito organizacional, financeiro e comercial, que levam à implementação de produtos ou processos inovadores.

A colaboração baseada em equipes é um fator crítico no desenvolvimento de atividades de P&D multidisciplinar. A aplicação de boas práticas em pesquisa indica que o trabalho em equipe e a colaboração dominam a produção de conhecimento em organizações acadêmicas, como podemos observar nas diversas redes internacionais de pesquisa em larga escala [Cooke and Hilton 2015]. A colaboração em atividades de PD&I pode ser vista como um processo que não envolve apenas diferentes instituições de pesquisa, mas que se expandem para incluir colaboração com o setor produtivo (empresas públicas e privadas), governos que definem as políticas públicas de inovação e sociedade civil que se beneficia da inovação. Em PD&I, podemos ter empresas e equipes multidisciplinares de cientistas trabalhando colaborativamente em projetos compartilhados. O processo de profissionalização da PD&I em empresas, instituições e até mesmo nos países vem se tornando cada vez mais claro e objetivo.

Para as empresas absorverem efetivamente, elas podem adotar o conceito de escritório de projetos. Escritório de Projetos ou Project Management Office (PMO) é uma estrutura organizacional centralizada e responsável pelo gerenciamento de projeto em toda a organização ou em um departamento. Representa um corpo ou entidade organizacional à qual são atribuídas várias responsabilidades relacionadas ao gerenciamento centralizado e coordenado dos projetos sob seu domínio. Existem diferentes tipos de Escritórios de Projeto, com atribuições, responsabilidades e níveis de autoridade diferentes. Dentre as possíveis atividades do Escritório de Projetos, podemos citar: suporte material, padronização de processos e procedimentos, criação de *templates* e modelos de documentos, treinamento de gerentes e membros de equipes de projeto, realização de mentoria, criação de um repositório de informações históricas (inteligência organizacional), gestão centralizada de softwares de gerenciamento de projetos [Trentim 2017].

Os escritórios de projetos costumam ser classificados em três tipos:

- **Autônomo.** trata-se de um Escritório de Projetos criado para dar suporte apenas a um determinado Projeto ou Programa devido à importância estratégica, vulto de recursos ou mesmo exigência do cliente. Este PMO desapareceria ao final do projeto ou programa;
- **Setorial ou departamental.** existe apenas dentro de um Departamento ou Divisão e é responsável pelo gerenciamento de projetos nesse âmbito; e
- **Corporativo.** controla todos os projetos e programas da organização, orientando os processos e estabelecendo os padrões de gerenciamento de projetos.

O setor produtivo busca cada vez mais soluções para aumentar seu grau de inovação, sua produtividade e sua competitividade no setor produtivo. Existem diversos instrumentos de estímulo às atividades de PD&I, tais como o desenvolvimento de políticas públicas para a realização de atividades de *benchmarking* e o apoio nas definições de estratégias de competitividade para países e empresas. Por exemplo a Lei no 11.196, de 21 de novembro de 2005, conhecida como Lei do Bem, que institui incentivos fiscais a empresas que promovam pesquisa e desenvolvimento de inovação tecnológica.

A colaboração contínua e de longo prazo entre o setor produtivo e as instituições de pesquisa é crucial para desenvolver pesquisa de qualidade em áreas específicas, a exemplo do desenvolvimento de *softwares*, que encapsula soluções de pesquisa. Apesar dos muitos benefícios mútuos, essa colaboração é muitas vezes desafiadora, não apenas devido aos objetivos diferentes, mas também devido ao ritmo diferente na execução de processos e na entrega dos resultados. Diante do exposto, não é difícil imaginar que projetos de PD&I com colaboração baseada em equipes, que envolvam organizações de pesquisa e o setor produtivo, requerem novas técnicas de gerenciamento de projetos que acompanhem a evolução das atividades e analisem o quanto estas atividades estão atendendo às necessidades de inovação. O gerenciamento ágil de projetos ou metodologia de desenvolvimento ágil trata-se de uma abordagem iterativa para gerenciamento de projetos que ajuda as equipes a entregar valor aos seus contratantes mais rapidamente e com menos dores de cabeça. A indústria de desenvolvimento de *software* já possui como premissa a adoção de estratégias ágeis em seus processos de desenvolvimento e entrega.

Ao se deparar com a oportunidade de se desenvolver um projeto de PD&I, sempre temos como base a seguinte pergunta:

Como conduzir projetos de PD&I a fim de aumentar a sinergia e produtividade entre os envolvidos?

Nossa principal hipótese é a de que, por intermédio de métodos ágeis é possível melhorar a sinergia e aumentar a produtividade destes dois atores quanto a execução de projetos de PD&I. Para validar essa hipótese identificaremos e apresentaremos, neste trabalho, vários casos, bem como nossa experiência em adaptações das estratégias ágeis para o desenvolvimento de projetos de PD&I.

Os institutos de pesquisa são muitas vezes considerados como um farol para a inovação nas empresas. Parcerias entre os institutos de pesquisa e o setor produtivo são fortemente estimuladas, principalmente após a popularização da Ciência dos Dados e Inteligência Artificial, cujos problemas demandam soluções complexas e específicas, em detrimento das “soluções de prateleira” [Atta-Owusu et al. 2021]. Como forma de melhorar a qualidade do processo de inovação, produtividade, competitividade e as demandas do setor produtivo, as empresas precisam agregar pessoas especializadas em suas equipes (mestres ou doutores), ou fazer parcerias com os institutos de pesquisa por intermédio de projetos de PD&I. Essa parcerias com institutos de pesquisa geram resultados positivos e estimulam a criação e adoção de novos processos ou produtos. Diversos trabalhos empíricos estudaram a implicação de pesquisas externas nos resultados de inovação de uma empresa, concluindo que projetos de pesquisa realizados com instituições externas afetam positivamente o sucesso de inovações de produtos [Grimpe and Kaiser 2010, Lee et al. 2014, Weber and Heidenreich 2018].

Na proposição e execução de projetos de PD&I, três atores se fazem muito presentes: institutos de pesquisa, empresas públicas e privadas (representando o setor produtivo). Cada um possui funcionamento e dinâmica específicos que acarretam em divergências quanto a execução de projetos dessa natureza. Em particular nos institutos de pesquisa, existe uma desconexão entre o que é desenvolvido e as necessidades do setor produtivo. O objeto de estudo que envolve pesquisa científica é extremamente específico e não possui aplicabilidade direta no setor produtivo. Por outro lado, a necessidade do setor produtivo (empresas públicas e privadas) engloba questões, por vezes, de conhecimento comum nos institutos de pesquisa e que não geram engajamento por parte dos pesquisadores.

Contudo, a cada dia temos observado um crescente uso de soluções de Inteligência Artificial para a geração de conhecimento nas mais diversas aplicações e cenários. Com isso, o setor produtivo tem se posicionado cada vez mais para prover essas soluções que exigem uma especificidade cujas customizações e melhorias necessitam de conhecimentos de domínio restrito, muitas vezes encontrados apenas nos institutos de pesquisa. Para suprir essa necessidade, o setor produtivo tem buscado duas alternativas: i. Contratar profissionais extremamente qualificados, no caso, com mestrado e doutorado. Porém, a composição de uma equipe com esse perfil pode onerar consideravelmente os projetos desenvolvidos tornando-os difíceis de manter, uma vez que a solução pretendida pode ser algo pontual e específico; e ii. Contratar os institutos de pesquisa para desenvolver projetos de PD&I com o objetivo de gerar um protótipo funcional da solução pretendida.

Nesse cenário, ao se contratar uma equipe de pesquisadores surge um problema com a comunicação entre os envolvidos no projeto. Por um lado, temos os profissionais do setor produtivo que costumam trabalhar de forma célere e preocupado com a entrega, muito mais que a qualidade do que está sendo feito e por outro lado, temos os pesquisadores que trabalham considerando uma metodologia científica que permite o embasamento sólido para o que se pretende fazer, logo preocupa-se mais com a qualidade dos resultados do que com a celeridade da entrega.

Quando uma empresa pública é o elemento que busca soluções customizadas por intermédio de projetos de PD&I com os institutos de pesquisa, a complexidade em nível de relações interpessoais e interdisciplinares aumentam consideravelmente. Essas empresas em geral desenvolvem projetos com o setor produtivo regidos por licitações e normas que exigem um acompanhamento minucioso de tudo o que foi especificado e com prazos bem rígidos. O processo de P&D nem sempre ocorre como planejado, pois trata-se de originar algo novo e complexo e que pode ser ajustado entre a sua proposição até a sua transferência. Além disso, o tempo necessário para se gerar resultados satisfatórios pode ser de difícil dimensionamento ao se especificar previamente a solução pretendida.

Esses problemas são relatados por diferentes pesquisadores de diferentes áreas e localidades. Acreditamos que estes sejam os grandes gargalos no processo de desenvolvimento de pesquisa junto ao setor produtivo. Para mitigar esse problema, identificamos que é necessário unir o melhor dos dois mundos, no que diz respeito a gestão dos projetos, para que ambos os lados se beneficiem do processo e possam, assim, incrementar sua produtividade de forma célere e sinérgica. Nesse sentido, a utilização de métodos ágeis (prática usual no setor produtivo) aliada à metodologia científica (prática usual nos institutos de pesquisa) seja o elemento catalisador para se executar projetos de forma integrada e alcançar resultados satisfatórios. Ademais, para os cenários de projetos envolvendo institutos de pesquisa e empresas públicas, a utilização de métodos ágeis permite a execução de ciclos curtos, logo o rearranjo das especificações tornam-se mais flexíveis; e as entregas a cada ciclo permite a composição da solução de forma incremental e um melhor dimensionamento para o tempo de execução.

Há muito vivemos no Brasil e na América Latina numa repetição contínua de tentativas que produzem os mesmos erros. Políticas de curto prazo com consequências negativas no longo prazo parecem ser quase irresistíveis para os nossos políticos e tomadores de decisões. Há que romper este ciclo focando o que não podemos repetir [Mendes (org.) 2022]. A vontade política precisa ser sensível ao papel da inovação e aos desafios de médio e longo prazo.

Por fim, objetivamos aguçar a curiosidade dos leitores em torno do uso de estratégias ágeis dentro de projetos de pesquisa, mais especificamente em projetos de PD&I desenvolvidos em parceria com o setor produtivo. O restante do trabalho segue com as seguintes seções: A Seção 2.2 introduz conceitos importantes relativos à pesquisa científica aplicada, à relação instituto de pesquisa e o setor produtivo no desenvolvimento de projetos de PD&I, bem como aos seus desafios em ser ágil. A Seção 2.3 conceitua o gerenciamento ágil de projetos, e detalha algumas das mais populares metodologias de gerenciamento ágil. A Seção 2.4 mostra como as metodologias ágeis estão sendo cada vez mais utilizadas para fins organizacionais em projetos de PD&I. A Seção 2.5 discute

alguns exemplos da aplicação de metodologias no contexto de projeto de PD&I. Por fim, seção 2.6 apresenta as considerações finais.

2.2. Pesquisa científica e seus desafios em ser ágil

Diferentes organizações em contextos e situações competitivas adotaram firmemente metodologias ágeis de trabalho. O *Project Management Institute* (PMI)¹ relatou que a abordagem ágil está sendo amplamente utilizada para práticas de gerenciamento de projetos, não apenas na indústria de *software* e isso tem impacto significativo no crescimento do negócio e no desempenho de projeto de diversas naturezas [Raharjo and Purwandari 2020]. Mas as atividades de PD&I em negócios orientados pela ciência está notavelmente ausente dessa tendência. Embora práticas e conceitos ágeis estejam transformando corporações em todos os setores, departamentos científicos foram deixados, quase intocados, em suas antigas formas de trabalhar, seguindo protocolos rígidos de pesquisa científica com pouca atenção para os desafios colaborativos envolvidos nos seus processos [di Fiori et al. 2019]. O mesmo pode-se esperar dos centros de pesquisa das universidades, que encontram-se mais distantes ainda do ambiente dinâmico e competitivo do setor produtivo.

Contudo, vale observamos que a pesquisa colaborativa é uma realidade em um mundo globalizado. O uso de estratégias de colaboração em equipes de trabalho é um fator crítico em diversos campos e organizações de pesquisa, pois o conhecimento está sendo cada vez mais gerado por equipes de pesquisa multidisciplinares [Wagner et al. 2017]. Dados os desafios que a maioria dos grupos que desenvolvem projetos de PD&I estão enfrentando, as estruturas ágeis podem entregar um novo valor no suporte às equipes ao longo de sua jornada, desde descobertas inovadoras até implementações comerciais bem-sucedidas.

De forma geral, as metodologias ágeis foram criadas a fim de auxiliar no processo de desenvolvimento de *software*, contudo, elas vem sendo expandidas para outros tipos de organizações e processos de gestão do conhecimento [Ciric et al. 2018]. Nesse sentido, elementos das metodologias ágeis podem ser considerados como blocos de construção para soluções de gerenciamento em geral. Esses elementos, combinados de uma nova forma e quando adotados, resultam em uma metodologia ágil específica para um determinado cenário de produção. Especificamente, para o desenvolvimento de projetos de PD&I, existem vários aspectos que devem ser considerados ao se criar uma metodologia ágil específica para uma equipe. É fundamental que, ao incorporar diferentes elementos das estratégias ágeis, preserve-se as práticas usuais da metodologia científica. Em sua concepção, projetos de PD&I consideram diferentes estratégias metodológicas, não necessariamente ágeis, para o desenvolvimento da pesquisa. As principais etapas que são recorrentemente consideradas nessas estratégias de desenvolvimento de pesquisa científica são:

- **Identificação do problema.** Para o desenvolvimento de uma pesquisa, o problema surge por intermédio de uma pergunta base. Tal pergunta pode ser elaborada levando em conta: i. A experiência (curiosidade) dos pesquisadores; ii. Os resultados prévios que não conseguiram ser explicados; ou iii. Um problema em aberto apre-

¹<https://www.pmi.org/>

sentado pelo setor produtivo ou sociedade. As atividades de pesquisa que envolvem projetos de PD&I na área de Computação, estão em torno do item iii. Aliado a identificação do problema, é comum termos hipóteses que precisam ser aceitas ou refutadas ao fim de uma pesquisa. Esse ponto, especificamente, pode tornar a pesquisa mais longa do que o aceitável pelo setor produtivo, que demandam soluções de curto e médio prazo. Logo, para o desenvolvimento de pesquisa junto ao setor produtivo é necessário considerar apenas as hipóteses que possam ser verificadas mais rapidamente.

- **Levantamento do estado da arte.** Espera-se que tal levantamento já tenha sido feito antes mesmo de se identificar o problema, pois, como apresentado, os problemas surgem por intermédio da experiência dos pesquisadores ou como continuação de uma pesquisa já existente. Portanto, o tema a ser estudado já é do domínio dos pesquisadores. No entanto, para o caso de um projeto de PD&I, os problemas que chegam até a equipe, não fazem parte da expertise dos pesquisadores. Portanto, o levantamento é feito após a identificação do problema. Nesse ponto, metodologias e processos que facilitem o rápido levantamento do estado da arte se fazem necessários.
- **Proposição de novas soluções.** Para uma pesquisa incremental, a proposição de novas soluções perpassa por soluções já desenvolvidas em outras aplicações, adaptando-as sempre de forma pautada na expertise e experiência dos pesquisadores. No entanto, para uma equipe que desenvolve um projeto de PD&I, note que não necessariamente há uma evolução linear do conhecimento adquirido, como no caso do pesquisador. Sendo assim, é fundamental a aplicação de métodos e soluções que acelerem a identificação do problema a ser resolvido. Geralmente, a solução adotada envolve a reprodução do estado da arte ou de aplicação direta de soluções que se assemelham ao problema que se está investigando.
- **Validação das soluções propostas.** Uma pesquisa, envolve diferentes métodos (qualitativos ou quantitativos) para a validação de uma solução, seja em um ambiente simulado ou real. Em todos os casos, o pesquisador tem liberdade para validar as soluções considerando diferentes ambientes e métodos que melhor se adéquem ao ferramental disponível. Além da validação anteriormente mencionada, em PD&I é necessário verificar a robustez da solução em ambiente de produção. Neste particular, a verificação de soluções de *software* envolve desempenho e escalabilidade. Assim, quanto mais rápido se tiver um versão verificável, mais rápido pode-se otimizar, adaptar (nem sempre) e transferir as soluções para os cenários de produção ou abandonar a estratégia adotada.

Recentemente, observou-se que a colaboração entre os institutos de pesquisa e o setor produtivo é, muitas vezes, evasiva [Atta-Owusu et al. 2021]. Uma razão para que isso ocorra pode ser dada pela existência de uma lacuna entre a produção científica e as necessidades do setor produtivo, seja pelo uso de diferentes metodologias de desenvolvimento ou pela dificuldade em integrar as soluções criadas. Desta forma, podemos assumir que projetos de PD&I envolvendo centros de pesquisa devem considerar, além da excelência na qualidade da pesquisa desenvolvida, estratégias que estreitem os laços com

o setor produtivo que carece de soluções de curto e médio prazo. Dentre as estratégias que permitem uma melhor colaboração entre ambos os setores, encontra-se a aplicação de métodos ágeis de gerenciamento de projetos e pessoas [Hidalgo 2019].

No Brasil, a evolução da universidade foi marcada por dois acontecimentos revolucionários inter-relacionados. Inicialmente, tivemos a “primeira revolução acadêmica”, que consistiu na inclusão da pesquisa na universidade, ao lado das atividades previamente definidas de docência/ensino. Em seguida, veio a “segunda revolução acadêmica” desencadeada pelo envolvimento da universidade com o setor produtivo, além do ensino e da pesquisa. Atualmente, mais de 95% da produção científica do Brasil nas bases internacionais deve-se à capacidade de pesquisa de suas universidades públicas². É fato indiscutível que a investigação científica é um importante motor da inovação social e econômica, na medida em que produz novos conhecimentos, competências e atitudes que são cruciais para o crescimento econômico e o desenvolvimento tecnológico do país. Dessa forma, a universidade, por consequência tornou-se esse motor.

Antes de falarmos sobre os desafios que as universidades, ou os institutos de pesquisa em geral, enfrentam para aplicar métodos ágeis em seus projetos de PD&I, é importante entendermos sobre o caminho existente entre a pesquisa científica e a inovação e como funciona a colaboração entre os institutos de pesquisa e o setor produtivo. São as restrições existentes nestes cenários que, majoritariamente, impulsionam a adoção de métodos de gerenciamento ágil no contexto da relação institutos de pesquisa e o setor produtivo.

Primeiramente, vale deixarmos claro o conceito amplo de inovação que estamos discutindo. Tratamos por projetos de inovação aqueles considerados como projetos em institutos de pesquisa com o objetivo de algum tipo de comercialização (que caracteriza a atividade de inovação, segundo o próprio manual de Frascati [OECD 2015]) ou projetos de pesquisa aplicada, já que esta trata-se de uma investigação original empreendida com o objetivo de adquirir novos conhecimentos, porém direcionada para um fim objetivo, prático e específico.

Os institutos de pesquisa são uma importante fonte de geração de conhecimento científico. Cada vez mais, o setor produtivo vem reconhecendo isso e buscando firmar alianças com institutos de pesquisa, não apenas para aprimorar sua base de conhecimento, mas também obter vantagem competitiva no setor produtivo. Para facilitar as colaborações entre os institutos de pesquisa/universidades e o setor produtivo (do inglês *University Industry Collaborations* - UICs), muitos países implementam políticas públicas que estimulam o apoio financeiro de empresas públicas e privadas para a alocação de recursos. No Brasil, uma dessas políticas públicas trata-se da Lei nº 10.973/2004 (Lei de Inovação) que estabelece medidas de incentivo à inovação e à pesquisa científica e tecnológica no ambiente produtivo, com vistas à capacitação e ao alcance da autonomia tecnológica e ao desenvolvimento industrial do País, nos termos dos arts. 218 e 219 da Constituição. Outra lei bastante popular, trata-se da Lei nº 11.196, de 21 de novembro de 2005, conhecida como *Lei do Bem*, que institui incentivos fiscais às empresas que promovam pesquisa e desenvolvimento de inovação tecnológica.

² <https://www.gov.br/capes/pt-br/centrais-de-conteudo/17012018-capes-incitesreport-final-pdf>

Podemos ver que a inovação se sustenta em um tripé que envolve institutos de pesquisa, setor produtivo e instituições governamentais, através de suas políticas públicas [Etzkowitz and Zhou 2017b]. Existem diversos modelos que implementam essa relação, a exemplo do modelo da Hélice Tríplice que oferece uma metodologia para buscar por pontos fortes e fracos locais e preencher lacunas nas relações entre institutos de pesquisa, setor produtivo e governos, objetivando desenvolver uma estratégia de inovação bem-sucedida [Etzkowitz and Zhou 2017a]. A Figura 2.1 ilustra o funcionamento do modelo da Hélice Tríplice, bem como os papéis de cada ator envolvido no processo: academia, setor produtivo e governo.



Figura 2.1. Tripé da inovação (ou triângulo da inovação) no modelo de Hélice Tríplice: Relação entre academia, setor produtivo e governo a fim de desenvolver estratégias de inovação bem-sucedidas. Fonte: Os próprios autores.

Contudo, por mais que os centros de pesquisa/universidades estejam no tripé da inovação, a gestão metodológica de atividades de inovação ainda não foi incorporada de forma eficiente nas mesmas. Por exemplo, a transferência de tecnologia desenvolvida dentro do ambiente universitário é frequentemente associada à transmissão formal de invenções baseadas na ciência através do licenciamento de tecnologia patenteada a uma empresa. Este é um processo altamente sobrecarregado na maioria dos países [Hayter et al. 2020]. Por mais que a pesquisa científica desenvolvida nas universidades seja fonte reconhecida de novos conhecimentos, e suas contribuições para a inovação se manifestem por meio da criação e transferência de novas tecnologias originadas da pesquisa acadêmica, essa transferência ainda acontece através de longos procedimentos burocráticos [Yordanova 2021].

Um trabalho muito interessante acerca dos fatores facilitadores das UICs foi apresentado por Tseng et al. [Tseng et al. 2020] que estudou sobre os fatores que afetam o desempenho de inovação universitária. Segundo os autores, para facilitar as colaborações entre os institutos de pesquisa e o setor produtivo, é necessário o apoio financeiro do governo e do setor produtivo para a alocação de recursos. Portanto, o estudo investigou os efeitos do financiamento das UICs no desempenho da inovação tecnológica das universidades em Taiwan. Um total de 163 questionários foram entregues a todas as universidades taiwanesas oficialmente reconhecidas e 145 respostas foram recebidas, gerando uma taxa

de resposta de 88,96%. Uma motivação para o desenvolvimento dessa pesquisa veio do fato de que o governo de Taiwan implementou uma variedade de políticas e programas para melhorar a capacidade de inovação em pesquisa das universidades e preencher a lacuna entre pesquisa e o setor industrial. Algumas hipóteses foram levantadas e validadas pelos autores desta pesquisa:

- H1** O financiamento de UICs influencia positivamente o desempenho de inovação tecnológica das universidades;
- H2** Mecanismos eficazes de gestão de UICs influenciam positivamente o financiamento da UIC;
- H3** O clima de inovação de UICs influencia positivamente o financiamento das UICs;
- H4** UICs influenciam positivamente o desempenho de inovação tecnológica das universidades;
- H5** O efeito direto do financiamento de UICs no desempenho da inovação tecnológica das universidades é positivamente moderado pelas recompensas que as UICs oferecem.

Com base nas relações entre essas hipóteses, um modelo de pesquisa foi apresentado (Figura 2.2). A regressão dos mínimos quadrados parciais (PLS) foi usada neste estudo para testar as hipóteses da pesquisa. O estudo confirmou que o financiamento das UICs influenciou significativamente a construção de ambientes de UICs bem desenvolvidos dentro das universidades e o aprimoramento do desempenho da inovação tecnológica (H1). Os resultados também mostraram que tanto o mecanismo formal de gestão das UICs (H2) quanto a implementação do clima de inovação das universidades (H3) afetam positivamente o financiamento das UICs. As recompensas financeiras das UICs são essenciais para afetar direta e positivamente o desempenho de inovação das universidades (H4). O estudo mostrou, por fim, que as recompensas financeiras das UICs não apenas mostraram um efeito direto no desempenho da inovação tecnológica das universidades, mas também moderaram a relação entre o financiamento da UIC e a produção acadêmica. Em outras palavras, o financiamento da UIC do governo e de parceiros industriais em combinação com programas de recompensa justa pôde facilitar um desempenho de inovação mais intenso nas universidades (H5).

No Brasil, diversas universidades, junto a seus centros/agências de inovação, desenvolveram meios de estimular o desenvolvimento de ações de inovação e empreendedorismo, a exemplo da AUSPIN - Agência USP de Inovação³ e da INOVA - Agência de Inovação da Unicamp⁴. Tais agências auxiliam pesquisadores em processos de definição de propriedade intelectual, transferência de tecnologia e afins.

Ainda em um contexto nacional, Cardoso et al. [Cardoso et al. 2018] analisou, de forma descritiva e qualitativa, os principais fatores facilitadores e restritivos no desenvolvimento de projetos de pesquisada da Universidade do Estado de Santa Catarina (UDESC) em colaboração com empresas. Sua análise refletiu o olhar de 10 líderes de

³<http://www.inovacao.usp.br/acoesausp/>

⁴<https://www.inova.unicamp.br/>

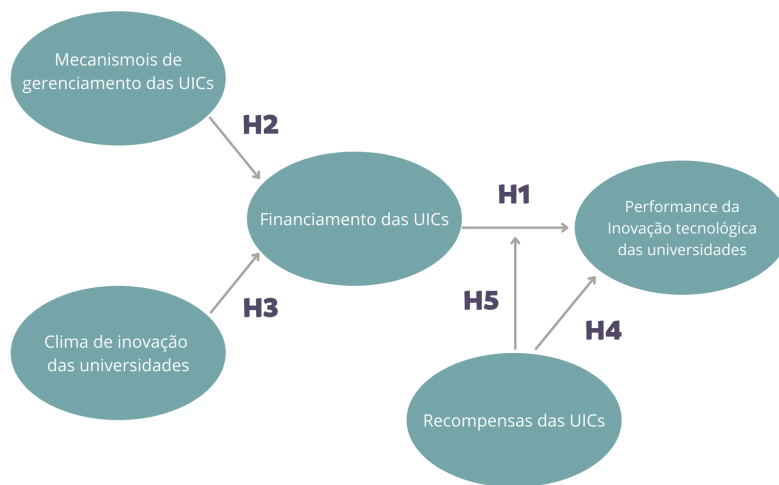


Figura 2.2. Modelo criado através das hipóteses colaborativas das UICs [Tseng et al. 2020]

grupos de pesquisa e de 14 gestores de empresas. O fator facilitador mais citado pelos líderes de grupos de pesquisa, entre os oito selecionados, foi “Recursos financeiros adicionais”, respondido por 70% (Figura 2.3). A Aquisição ou acesso a recursos materiais (ex: novos equipamentos) foi considerado por 60% dos pesquisados. O reforço da reputação e da imagem da universidade foi considerado por 40% dos pesquisados. Este reforço positivo também foi confirmado em pesquisas anteriores [Ankrah and AL-Tabbaa 2015], considerando UICs em outros países. As UICs promovem uma divulgação da imagem e o reforço da reputação da universidade na comunidade como um todo.

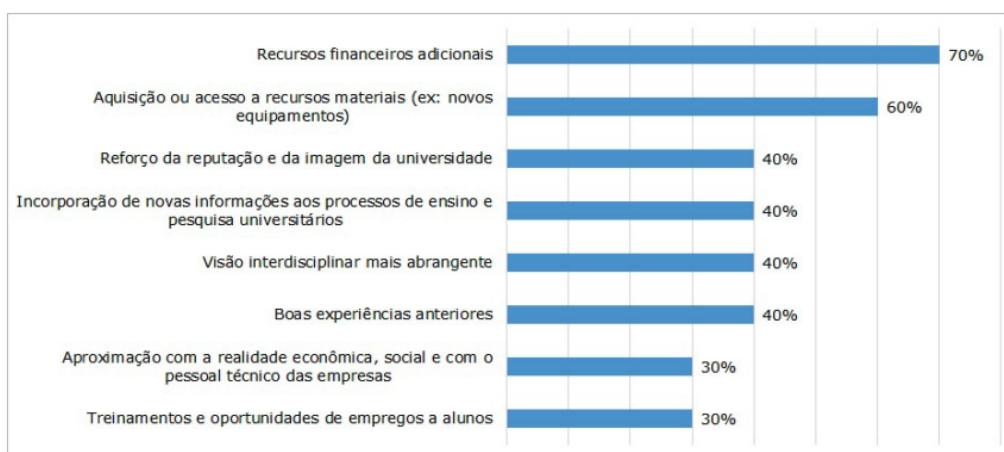


Figura 2.3. Fatores facilitadores das UICs na visão dos líderes de grupos de pesquisa [Cardoso et al. 2018].

Do ponto de vista das empresas, o fator facilitador mais citado foi o acesso a pesquisadores qualificados (mestres e doutores) na área de interesse (57%). A localização da universidade foi o segundo fator facilitador mais citado pelas empresas (50%). Outros fatores facilitadores identificados por 43% dos gestores de empresas foram, respectivamente: i. A redução dos custos e/ou riscos envolvidos nos projetos de PD&I; ii.

O aprendizado e atualização constante; iii. A possibilidade de resolução de problemas específicos da empresa; e iv. A possibilidade de transferência de tecnologia desenvolvida na universidade.



Figura 2.4. Fatores facilitadores das UICs na visão dos gestores de empresas [Cardoso et al. 2018].

Os fatores restritivos apontados pelos líderes de grupos e projetos de pesquisa envolvidas no estudo podem ser vistos na figura 2.5. A burocracia e morosidade jurídico-administrativa foi mencionado por 90% desses líderes. Segundo eles, o excesso de burocracia inibe a realização de parcerias com as empresas e, este fato, ligado aos aspectos legais e formais, é uma das principais barreiras no processo de formação de UICs. Também foi considerado que o processo burocrático também pode atrasar a transferência de tecnologias e o cumprimento de objetivos da empresa, podendo gerar conflitos e conduzir à baixa produtividade. O pouco conhecimento das empresas em relação ao potencial e capacidade da universidade e a complexidade de procedimentos envolvidos e a demora nas compras de materiais e equipamentos utilizados nas pesquisas foram citados por 50% dos líderes de grupos de pesquisa, respectivamente, como fatores restritivos nas UICs. A visão imediatista e de curto prazo por parte das empresas também foi considerada um fator restritivo para 40% dos líderes de grupos de pesquisa.

Esse estudo também apontou os fatores restritivos mencionado pelas empresas envolvidas nas UICs (Figura 2.6). Segundo Cardoso et al. [Cardoso et al. 2018], dos oito fatores restritivos citados, quatro deles também foram mencionados pelos líderes de grupos de pesquisa, a saber: i. Excesso de burocracia na universidade, citado por 50% dos gestores de empresas e por 90% líderes de grupos de pesquisa; ii. Pouco conhecimento do potencial e das capacidades instaladas na universidade, citado por 36% dos gestores de empresas e por 50% dos líderes de grupos de pesquisa; iii. Imediatismo por parte do setor produtivo na busca por resultados, citado por 43% dos gestores de empresas e por 40% dos líderes de grupos de pesquisa; e iv. Pouco conhecimento das leis que tratam sobre cooperação com as universidades foi outro fator restritivo identificado, citado por 36% dos gestores de empresas e por 20% dos líderes de grupos de pesquisa. Além do excesso de burocracia na universidade, citado por 50% dos gestores de empresas, o pouco conhecimento das linhas de pesquisa da universidade também foi outro fator restritivo citado por 50% dos gestores que participaram da pesquisa.



Figura 2.5. Fatores restritivos das UICs na visão dos líderes de grupos de pesquisa [Cardoso et al. 2018].



Figura 2.6. Fatores restritivos na visão dos gestores de empresas [Cardoso et al. 2018].

Algumas questões surgem quando analisamos os fatores levantados na pesquisa citada acima: i. Os excessos de procedimentos burocráticos salientam a priorização de negociações de contratos em detrimento à atividade de colaboração? ii. O pouco conhecimento do potencial e das capacidades instaladas na universidade nos evidenciam o abismo que existe entre nossos centros de pesquisa e o setor produtivo? iii. A visão do imediatismo por parte do setor produtivo poderia ser amenizada com o advento de estratégias de colaboração flexíveis e interativas, que respondessem bem à mudanças? e iv. A entrega frequente de soluções durante o processo de desenvolvimento do projeto engajaria os envolvidos e geraria mais satisfação?

O tradicional modelo em cascata é usado em institutos de pesquisa para o gerenciamento de projetos de PD&I. Existem algumas pesquisas que visam desenvolver iniciativas colaborativas de projetos de PD&I entre institutos de pesquisa e o setor produtivo, apresentando uma abordagem híbrida de gerenciamento de projetos, com um conjunto de práticas de gerenciamento de projetos distintas no contexto de tal colaboração.

Com base nas questões levantadas podemos assumir que o uso de estratégias ágeis no desenvolvimentos desses projetos de UICs, podem vir a sanar muitos desses problemas, hoje vistos como restrições colaborativas. Por fim, Yordanova et al. [Yordanova et al. 2019] apontou que uma das principais hipóteses do seu estudo é que projetos inovadores implementados em organizações científicas requerem uma metodologia específica para sua gestão bem sucedida e a criação de inovações sustentáveis. Metodologias flexíveis como as ágeis são ferramentas que têm sido implementadas com sucesso no setor produtivo, especificamente para desenvolver e gerir a inovação, e isso leva a supor que seriam ferramentas de sucesso para criar e gerenciar a inovação em organizações de pesquisa.

2.3. Estratégias ágeis para o desenvolvimento de software

O gerenciamento ágil, ou metodologia de desenvolvimento ágil, trata-se de uma abordagem iterativa para gerenciamento de projetos que ajuda as equipes a entregar valor aos seus contratantes mais rapidamente e com menos dores de cabeça. Para tanto, esta metodologia oferece suporte contínuo e incremental na organização e execução das tarefas prioritárias de trabalho, mesmo em face de possíveis mudanças [Cohen et al. 2004].

Os formatos de gerenciamento de projetos se encaixam em três categorias principais:

1. **Modelos preditivos:** Contemplam os modelos tradicionais em cascata (do inglês, *waterfall*) e definem, de maneira relativamente engessada, todo o escopo de criação, planejamento, implementação e testes de uma solução, executando consecutivamente e em sua completude cada uma dessas etapas.
2. **Modelos incrementais e iterativos:** Quebram a solução a ser entregue em escopos menores e implementam um ciclo preditivo em cada escopo, separadamente. Nesse caso, a solução desenvolvida cresce a cada iteração do processo.
3. **Modelos adaptativos:** Contemplam as metodologias ágeis e são incrementais e iterativos, com ciclos de tempo reduzidos e de duração fixa, chamados *Sprints*. Estas *Sprints* tratam-se de modelos preditivos flexíveis e orientadas a mudanças e não são engessadas.

Nosso foco está na aplicação de modelos adaptativos (ágeis) para o gerenciamento de projetos PD&I. Porém, antes de entrarmos em mais detalhes sobre o que são esses métodos ágeis, comecemos pelo “porquê”:

Métodos ágeis permitem que as organizações (do setor produtivo, da pesquisa científica, educacionais, etc) lidem com mudanças contínuas. Permitem florescer em um mundo cada vez mais volátil, incerto, complexo e ambíguo. A ascensão dos métodos ágeis é impulsionada tanto pela paixão daqueles que amam trabalhar dessa maneira quanto por organizações que estão fazendo uma descoberta transformadora: a única maneira de lidar de forma sustentável em ambientes cada vez mais inovadores é adotar esses métodos. As organizações devem se tornar tão ágeis quanto o contexto em rápida mudança em que se encontram.

A priori, as metodologias ágeis foram criadas a fim de auxiliar no processo de desenvolvimento de *software* no âmbito do setor produtivo. À medida que o gerenciamento de *software* se tornou um ponto central para o sucesso da maioria das empresas, métodos ágeis se tornaram a chave para o gerenciamento de tudo. As empresas deste setor abriram mão do método de desenvolvimento em cascata, que apostava na entrega de um produto através de um processo longo, engessado e de alto risco, e passaram a trabalhar com uma equipe ágil na entrega do produto desenvolvido com base em incrementos pequenos, mas consumíveis e com riscos menores associados.

O modelo de desenvolvimento em cascata traz uma abordagem linear que envolve etapas como: documentação de requisitos e análise do projeto (etapa de definição); implementação e testes (etapa de construção) e implantação e manutenção (etapa de lançamento) (Figura 2.7, parte superior). Neste modelo, cada uma destas etapas representa um estágio diferente no desenvolvimento de *software*, e estes estágios tendem a ser disjuntos na linha do tempo, ou seja, cada estágio geralmente termina antes do próximo iniciar. Por exemplo, os requisitos geralmente são revisados e aprovados pelo cliente antes que a etapa de construção do projeto seja iniciada, não havendo espaço para alteração dos mesmos em quaisquer etapas posteriores.

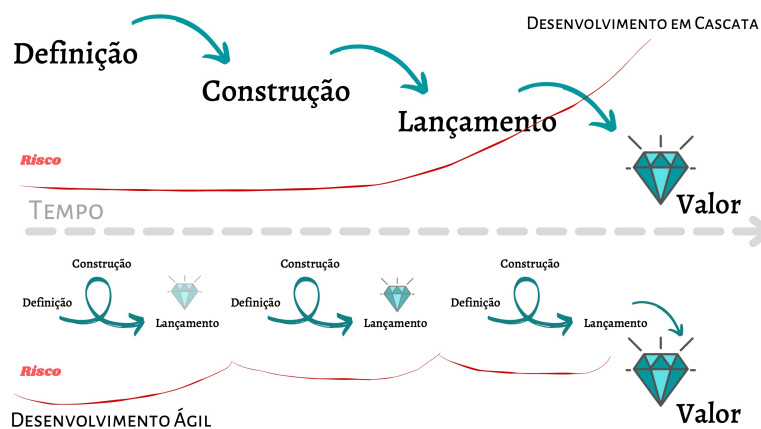


Figura 2.7. Modelo de desenvolvimento em cascata versus modelo de desenvolvimento ágil [Pogrebnoy and Yatskevich 2022].

A metodologia ágil (Figura 2.7, parte inferior), por outro lado, é bastante flexível e permite que sejam feitas mudanças nos requisitos de desenvolvimento do projeto, mesmo após o planejamento inicial ter sido concluído. Ademais, no modelo em cascata, todas as etapas de desenvolvimento do projeto são concluídas uma única única vez e quase nunca revisadas e/ou alteradas, enquanto que, como parte da metodologia ágil, elas seguem uma abordagem de desenvolvimento iterativo. Como resultado, estas etapas podem aparecer mais de uma vez durante todo o ciclo de vida do desenvolvimento.

Apesar de ter sua origem no desenvolvimento de *software*, as metodologias ágeis vem cada vez mais se expandindo para outros tipos de organizações e processos de gestão do conhecimento. Atualmente, existem mais de quarenta variações de métodos ágeis, faremos aqui uma breve explanação sobre alguns desses métodos partindo da compreensão de que todos são provenientes de uma mesma origem e buscam atender aos mesmos princípios e causas, resumidos no que conhecemos como manifesto ágil.

Anteriormente a criação do manifesto ágil, tais métodos eram popularmente conhecidos como métodos de “peso leve” (do inglês, *lightweight*), enquanto que modelos em cascata eram conhecidos como “peso pesado” (do inglês, *heavyweight*). Durante a criação do manifesto, o termo “peso leve”, que parecia mais uma reação ao modelo tradicional e não uma metodologia em si, foi substituído por “ágil”. É importante então percebermos que o gerenciamento ágil trata-se de uma metodologia eficiente e não uma anti-metodologia.

2.3.1. O manifesto ágil

Como já mencionado, os métodos ágeis, inicialmente, constituíram-se de uma reação às formas tradicionais de desenvolvimento de *software* e partiram do reconhecimento da necessidade de uma alternativa para execução de processos de desenvolvimento de *software* complexos e orientados a documentação [Beck et al. 2001]. A partir de meados da década de 90, alguns profissionais acharam o cumprimento dos passos do desenvolvimento em cascata frustrantes ou até mesmo impossíveis [Highsmith and Highsmith 2002]. A inovação tecnológica tem por natureza a constante ascensão e evolução cada vez mais rápida. Os clientes se tornaram cada vez mais incapazes de declarar definitivamente suas necessidades com antecedência. Contudo, ao mesmo tempo, passaram a esperar mais de seu *software*. Como resultado, vários consultores desenvolveram métodos e práticas de forma independente para responder à mudança inevitável que estavam vivenciando. Esses métodos ágeis são na verdade uma coleção de diferentes técnicas (ou práticas) que compartilham os mesmos valores e princípios básicos. Afim de fazer um *link* com o contexto deste trabalho, vale salientar que os requisitos em um projeto de PD&I mudam frequentemente e em algumas situações, drasticamente. Tais mudanças também exigem a adoção de metodologias de gerenciamento adaptativas e flexíveis.

Em fevereiro de 2001, houve uma reunião que envolveu 17 representantes de métodos como o *Extreme Programming* (XP), Scrum, *Dynamic Systems Development Method* (DSDM), *Adaptive Software Development*, *Crystal*, *Feature-Driven Development*, *Pragmatic Programming*, e outros métodos. Na pauta: discutir as semelhanças e diferenças entre os métodos que utilizavam/defendiam. Foi observado que os pontos em comum de seus métodos levavam a um consenso de suas práticas. A partir dessa grande interseção em seus modelos de gerenciamento do desenvolvimento de *software* em larga escala, surgiu o termo “ágil” e junto com ele, um manifesto com princípios e valores que até hoje servem como base para uma prática de gerenciamento de projetos que prioriza a eficiência, eficácia e flexibilidade. Segundo os autores do manifesto ágil [Beck et al. 2001]:

O movimento Ágil não é anti-metodologia, contrário a isso, muitos de nós querem restaurar a credibilidade da palavra metodologia. Queremos restaurar o equilíbrio. Abraçamos a modelagem, mas não para arquivar algum diagrama em um repositório corporativo empoeirado. Aceitamos a documentação, mas não centenas de páginas de tomos nunca mantidos e raramente usados. Planejamos, mas reconhecemos os limites do planejamento em um ambiente turbulento.

A não-flexibilidade do modelo em cascata, exigia uma grande quantidade de docu-

mentos, estendendo o processo de definição do projeto. Tudo precisava ser documentado seguindo rígidos protocolos de levantamento de requisitos e modelagem, fazendo com que as soluções em *software* mais complexos demorassem muito a serem desenvolvidas a ponto de já não terem mais função quando entregues. Dessa forma, um ponto em comum que justificou a reunião em 2001 foi a busca por métodos que minimizassem a documentação produzida no processo de desenvolvimento de *software*. Mas este não foi o único problemático a ser discutido, outros pontos motivadores importantes foram: valorização dos indivíduos envolvidos nas equipes de desenvolvimento de *software*; valorização do cliente e suas reais necessidades; valorização da colaboração entre equipes de desenvolvimento e clientes; valorização da entrega de Produtos Mínimos Viáveis (PMVs) durante todo o processo do desenvolvimento.

A partir de todos esses pontos, originou-se o Manifesto Ágil, com seus 4 valores e 12 princípios. A principal justificativa para a criação do manifesto, veio da seguinte premissa:

Estamos descobrindo maneiras melhores de desenvolver software fazendo-o nós mesmos e ajudando outros a fazerem. [Beck et al. 2001].

A partir desta premissa, de natureza colaborativa, foram construídos os 4 valores do “Ágil”, sustentáculos do manifesto:

1. **Indivíduos e interações mais que processos e ferramentas:** Um fator crucial de sucesso de um projeto reside nas relações profissionais entre as pessoas que unem forças na construção e implementação de um projeto. Dessa forma, observar como essas pessoas interagem e imprimir boas práticas colaborativas dentro de um time/equipe pode definir o sucesso do projeto. No gerenciamento ágil, essas interações podem vir de conversas rápidas diárias e de atividades de *brainstorming* para resolução de problemas e revisões de andamento das atividades do time. Processos e ferramentas continuam sendo importantes para o desenvolvimento de qualquer tipo de produto e devem ser definidos e utilizados durante todos os ciclos iterativos da execução de um projeto, ainda que de forma simplificada. Contudo, estes não podem substituir as interações humanas e o gerenciamento ágil prioriza estas interações. Estratégias de colaboração estão no cerne do ágil, que tende a prezar por estratégias que evitem desperdício de tempo e recursos. Dessa forma, afim de manter a agilidade, evita-se o uso “abusivo” de processos e ferramentas no lugar de uma breve conversa, por exemplo.
2. **Software em funcionamento mais que documentação abrangente:** No contexto de desenvolvimento, um *software* funcionando vale mais do que mil palavras documentadas. A partir dessa forma de pensamento, os representantes do manifesto nos trazem à luz a importância da simplificação da documentação de um projeto. Expandindo esse conceito para contextos diferentes, poderíamos substituir o termo. “*Software* funcionando” por “Soluções funcionando”. Nas próximas seções focaremos em “Soluções funcionando” no contexto específico do desenvolvimento de projetos PD&I, foco deste trabalho. Por hora, vamos nos ater a essa extrapolação do termo para contextos mais genéricos de gerenciamento de projetos. O conjunto

de soluções completas, entregues ao fim da execução de um projeto, tendem a ser produtos complexos, com regras de negócio e/ou protocolos de desenvolvimento embutidos. A medida que tais soluções forem entregues, paulatinamente, em formas de PMVs ao cliente, elas podem nortear os caminhos a serem tomados durante a execução do projeto, evitando riscos de precisarem ser alteradas em última instância, em que, seu nível de complexidade e dependência com outras soluções entregues em conjunto, torne a alteração inviável. Em síntese, a documentação do projeto deve ser parte integrante das entregas de PMVs, contudo, soluções em funcionamento entregue ao final dos ciclos iterativos durante o projeto é o que tem mais valor para o usuário final da solução desenvolvida.

3. **Colaboração com o cliente mais que negociação de contratos:** Segundo Cruz [Cruz 2015], este talvez seja o valor mais fácil de entender, contudo o mais difícil de implementar. O autor justifica sua afirmação sob a observação de que esta etapa envolve o cliente, que apesar de não ter controle sobre a equipe atuante no projeto, é parte integrante do mesmo. Numa relação indústria-ciência, por exemplo, para o desenvolvimento de projetos PD&I, o cliente, em geral, encontra-se do lado do setor produtivo e busca por soluções inovadoras e de curto e médio prazo, apesar de, geralmente, desconhecer as etapas do método científico implementadas na condução de um projeto dessa natureza. Colaborar com o cliente implica em trazê-lo o mais próximo o possível do projeto, deixando transparentes questões de sucesso e risco. Deixando explícito o que é possível realizar e o que não é, isso inclui não atender deliberadamente às vontades e imaginação do cliente. Quando existe colaboração entre o cliente e o time do projeto, há transparência e, por consequência, há mais espaço para colaboração. Ou seja, esse ciclo colaborativo é retroalimentado. Um projeto que mantém bons níveis colaborativos com o cliente, tende a engajá-lo mais no sentido de fazer parte do time do que de punir o time por possíveis contratemplos ou falhas de comunicação. Se o cliente, ou suas partes interessadas, não participa colaborativamente de um projeto, ele pode desenvolver o que Cruz [Cruz 2015] chama de miopia. Essa miopia implica em o cliente não enxergar as dificuldades, gargalos e/ou a capacidade real do time em realizar as tarefas vinculadas do projeto, o que acarreta em situações de cobranças indevidas, punições desnecessárias e problemas de entendimento.
4. **Responder a mudanças mais que seguir um plano:** Se nós formos pensar nas situações extremas de respostas à mudanças, nós podemos ter uma equipe inflexível, que entregará um trabalho engessado e incongruente com o cenário de constantes mudanças ou uma equipe que aceite qualquer “devaneio” que a faça mudar suas atividades drasticamente, a ponto de se perderem no processo e não atingirem seus objetivos. Ambas as situações não são adequadas ao andamento saudável do gerenciamento de projeto, de qualquer natureza. No contexto da metodologia ágil, a resposta à mudanças deve ser implementada com moderação, planejamento, foco, transparência e colaboração junto ao cliente. A mudança deve ser uma medida tomada conjuntamente, sob forte justificativa. Os impactos da mudança e os riscos associados a ela devem estar sempre claros. Esse conjunto de diretrizes em torno da mudança, são primordiais para entregar valor às soluções em funcionamento de um projeto.

Atender a uma mudança e responder a ela no tempo correto, com a velocidade adequada e na direção certa é um benefício para qualquer projeto e pode ser o divisor de águas entre o sucesso e o fracasso de um projeto. [Cruz 2015]

Projetos de PD&I, por exemplo, apesar de seguirem rígidos protocolos estabelecidos pela conduta científica, são desenvolvidos a partir de hipóteses, que a depender da condução do desenvolvimento do projeto podem ser validadas ou não. Isso implica que projetos dessa natureza, podem sofrer mudanças no decorrer de sua implementação e, que considerar as diretrizes de resposta à mudanças nesse tipo de projeto é fundamental para se obter uma solução inovadora ao final do mesmo.

Para os criadores do manifesto, os valores à esquerda do “mais que” são as premissas prioritariamente valoradas das metodologias de gerenciamento ágil. Os valores a direita, apesar de não serem excluídas do processo, não assumem protagonismos no gerenciamento. Para nossos fins específicos, vamos discutir sobre essas premissas para além do contexto do desenvolvimento de *software*, expandindo a discussão para o contexto mais generalista do gerenciamento de projetos.

Além dos valores acima discutidos, o manifesto apresenta também 12 princípios que equivalem a um conjunto de fundamentos que complementam esses valores e reforçam a importância da comunicação e da satisfação dos indivíduos envolvidos no processo. Além disso, também destaca-se a entrega contínua em um ritmo saudável e a funcionalidade produto entregue. Estes princípios [Beck et al. 2001] estão listados abaixo:

- **Geração de valor:** Nossa maior prioridade é satisfazer o cliente, através da entrega adiantada e contínua de *software* de valor.
- **Flexibilidade:** Aceitar mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adaptam às mudanças, para que o cliente possa tirar vantagens competitivas.
- **Frequência:** Entregar o *software* em funcionamento com frequência, seja na escala de semanas ou meses, dando preferência a períodos mais curtos.
- **União:** Tanto pessoas relacionadas a negócios como desenvolvedores devem trabalhar em conjunto, diariamente, durante todo o curso do projeto.
- **Motivação:** Construir projetos em torno de indivíduos motivados, dando a eles o ambiente e o suporte necessário e confiando neles para fazer o trabalho.
- **Comunicação:** O método mais eficiente e eficaz de transmitir informações para e entre uma equipe de desenvolvimento é por meio de conversa face a face.
- **Funcionalidade:** *Software* funcionando é a medida primária de progresso.
- **Sustentabilidade:** Os processos ágeis promovem desenvolvimento sustentável. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.

- **Revisão:** Contínua atenção à excelência técnica e bom *design*, aumenta a agilidade.
- **Simplicidade:** A arte de maximizar a quantidade de trabalho que não precisou ser feito.
- **Organização:** As melhores arquiteturas, requisitos e designs emergem de times auto-organizáveis.
- **Autoavaliação:** Em intervalos regulares, o time reflete em como ficar mais efetivo, então, se ajustam e otimizam seu comportamento de acordo.

Vale ratificar que o manifesto foi criado para ser efetivo em um contexto de desenvolvimento de *software*. Quando extrapola-se o contexto, é preciso verificar que princípios se adequam ao mesmo e a partir daí, fazer um uso eficiente da metodologia ágil. Como já foi apontado, atualmente, existem diversas variantes de métodos ágeis que se inspiram nos valores e princípios do manifesto ágil. Dentre os mais comumente utilizados estão os métodos Kanban [Kanban University 2021], XP [Beck and Andres 2004] e o Scrum [Sutherland 2014]. Abaixo detalhamos pontos interessantes desses métodos, com um foco maior no *framework* Scrum.

2.3.2. A metodologia Kanban

Extraímos do guia oficial do método Kanban [Kanban University 2021] os principais conceitos e definição relacionados a esse método. Kanban é um método para gerenciar todos os tipos de serviços profissionais, criado na década de 60 na empresa Toyota para auxiliar em sua linha de produção e hoje é utilizada nas mais diversas aplicações. Aos que são habituados a utilizar esse método, é internalizado que será aplicado uma maneira holística de pensar sobre seus serviços com foco em melhorá-los da perspectiva de seus clientes.

Com o método Kanban, você visualiza o trabalho invisível e como ele se move em um fluxo de um projeto. Isso ajuda a operar seus negócios com eficiência, incluindo a compreensão e o gerenciamento de riscos na entrega de seus serviços aos clientes. Com Kanban, você e sua empresa desenvolverão uma capacidade adaptativa ao longo do tempo para responder melhor e mais rapidamente às mudanças nas necessidades e expectativas de seus clientes ou em seu ambiente de negócios. O Kanban é amplamente conhecido pelo uso dentro das equipes, para aliviar a sobrecarga e ganhar o controle sobre o trabalho realizado pela equipe. Quando usado com foco no serviço, o Kanban é uma ferramenta eficaz de desenvolvimento organizacional.

De forma geral, como apresentado na figura 2.8, o método Kanban possui 3 grupos de princípios gerais: Gerenciamento de mudanças; Prestação de serviços; e Práticas gerais. Especificamente os princípios de gerenciamento de mudanças são comuns a todas as implementações Kanban e consiste nos seguintes: i. Comece com o que você faz agora; ii. Concorde em buscar melhorias por meio de evolução e mudança; e iii. Incentive atos de liderança em todos os níveis. Com base nesses princípios, o Kanban busca melhorar a maneira de se trabalhar usando diferentes formas de troca de informações e colaboração. A proposta é que se tenham mudanças evolutivas por meio de percepções obtidas pelas

peças que trabalham com o quadro Kanban e por atos de liderança visando a melhoria constante de sua maneira de trabalhar.

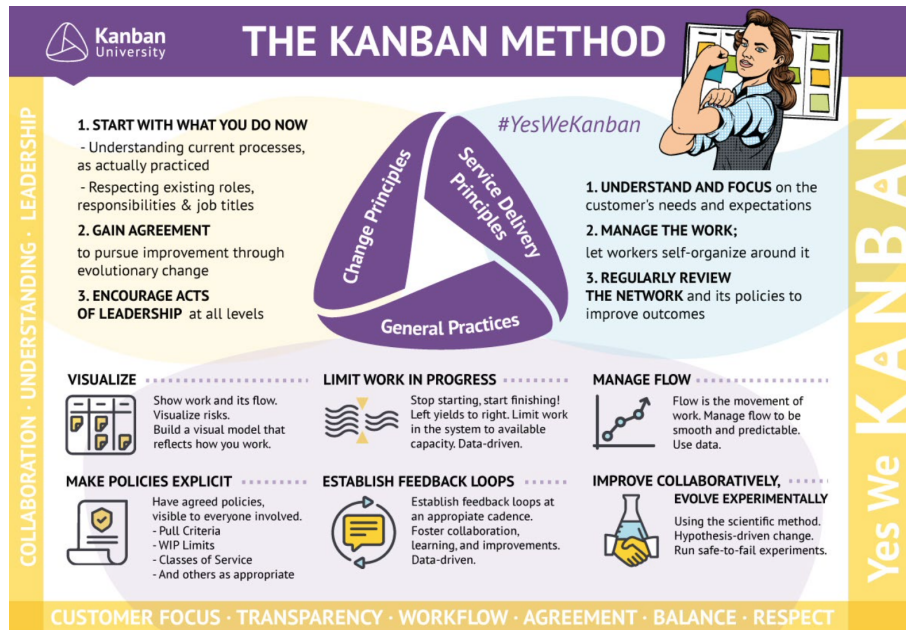


Figura 2.8. Método Kanban [Kanban University 2021].

O segundo grupo de princípios (Figura 2.8) são os de prestação de serviços. O Kanban incentiva você a adotar uma abordagem orientada a serviços para entender sua organização e como o trabalho flui por ela. Esse paradigma organizacional orientado a serviços baseia-se na ideia de que sua organização é uma entidade orgânica que consiste em uma rede de serviços, cada um deles vivendo, respirando e evoluindo. As solicitações dos clientes fluem por essa rede de serviços. Se quisermos melhorar a prestação de serviços, as melhorias devem ser guiadas pelo seguinte conjunto de princípios: i. Compreenda e foque nas necessidades do cliente e expectativas; ii. Gerencie o trabalho e deixe as pessoas se auto-organizarem em torno dele; e iii. Reveja regularmente a rede de serviços e suas políticas para melhorar os resultados. Esses princípios podem não ser utilizados desde o início pelas organizações, pois podem não ter uma mentalidade orientada a serviços como parte de sua cultura. Porém, a inclusão gradual desses princípios pode incrementar a qualidade dos serviços prestados.

Por fim, o terceiro grupo de serviços são os de práticas gerais. Para tal, o Kanban especifica 6 práticas, a saber:

- **Visualize:** Uma boa visualização é a chave para uma colaboração eficaz e para identificar oportunidades de melhoria. Visualizar o trabalho geral que necessita ser realizado e o fluxo desse trabalho melhora muito a transparência nas organizações. Além disso, a visualização auxilia na cooperação, pois todos os envolvidos literalmente têm a mesma imagem.
- **Limite o trabalho em andamento:** O trabalho em andamento (*work in progress* - WIP) indica o número de itens de trabalho em andamento em um determinado mo-

mento. Acredita-se que sistemas eficazes se concentram mais no fluxo de trabalho e menos na utilização do trabalhador. No Kanban, o WIP é limitado para equilibrar a utilização e ainda garantir o fluxo de trabalho.

- **Gerencie o fluxo:** O objetivo de gerenciar o fluxo de trabalho é concluir o trabalho da maneira mais suave e previsível possível, mantendo um ritmo sustentável. O monitoramento ou medição dos resultados do fluxo de trabalho são muito úteis para gerenciar expectativas com clientes e realizar previsões e melhorias.
- **Torne as políticas explícitas:** Durante o desenvolvimento dos projetos, inúmeras decisões são tomadas sobre a organização do trabalho, seja por indivíduos ou entre grupos de pessoas. Considere um novo funcionário na organização, logo com políticas explícitas, esse funcionário entenderá rapidamente como o trabalho é organizado. Dentre as possíveis políticas incluímos: i. Políticas de reposição do quadro (quando, quanto, por quem); ii. Definição de quando uma atividade de trabalho é concluída e quando o item de trabalho pode seguir em frente (“critérios de puxar”); iii. Limites de trabalho em andamento; iv. Políticas para lidar com itens de trabalho de diferentes classes de serviço; e v. Horários e conteúdo das reuniões. Além destes, outros princípios e acordos de colaboração são sempre empregados. Todas essas políticas devem ser acordadas em conjunto entre todas as partes envolvidas, incluindo clientes, partes interessadas e funcionários responsáveis pelo trabalho no quadro. As políticas devem ser colocadas em uma área claramente visível, de preferência ao lado do quadro.
- **Implementar ciclos de interações:** Esses ciclos são necessários para uma entrega coordenada e para melhorar a entrega do seu serviço. Um conjunto funcional de ciclos apropriados para o contexto dado fortalece as capacidades de aprendizagem da organização e sua evolução por meio de experimentos gerenciados. Alguns meios comumente usados são o quadro, as métricas e um conjunto de reuniões e revisões regulares que são chamadas de cadências.
- **Melhore colaborativamente, evolua experimentalmente:** No método Kanban nós “começamos com o que você faz agora” e “concordamos em buscar melhorias por meio de mudanças evolutivas”. Kanban é um método para mudança contínua, e fazemos essas mudanças de forma colaborativa usando experimentos projetados com base em modelos e no método científico. Projetamos experimentos tolerantes a falhas de tal forma que, se nossa hipótese estiver correta e o experimento der bons resultados, mantemos a mudança, mas se os resultados não forem positivos, podemos facilmente voltar ao estado anterior.

Todo esse processo acima descrito culmina na montagem dos quadros Kanban que, por sua vez, são o meio mais comum de visualizar um sistema Kanban. O seu funcionamento padrão envolve puxar os quadros num fluxo de trabalho da esquerda para a direita. Novos itens de trabalho sempre entram no quadro pela esquerda e quando eles saem à direita, o valor é entregue aos clientes. Os itens de trabalho podem ser de diferentes tipos e tamanhos, de tarefas a requisitos, tipos de artefatos, (grupos de) características de produtos e tópicos de projetos. Como exemplos temos: campanhas em agências, histórias

de usuários em equipes de desenvolvimento de *software*, cargos em RH ou produtos para um grupo de desenvolvimento de produtos.

Os itens de trabalho geralmente são exibidos em notas individuais (em papel), que geralmente são chamadas de cartões ou bilhetes. A série de atividades pelas quais esses itens de trabalho passam são chamadas de fluxo de trabalho. O Kanban é baseado na abordagem “comece onde você está agora”, portanto, o fluxo de trabalho real está sendo modelado no quadro Kanban. As etapas individuais no fluxo de trabalho são mostradas em colunas. As pistas são frequentemente usadas para diferentes tipos de trabalho, projetos, etc.

Considere o exemplo do trabalho de um provedor de serviços de treinamento interno em uma empresa [Kanban University 2021]. As ideias ou requisitos para novos cursos são coletados primeiro. Após um processo de seleção e refinamento, novos cursos são desenvolvidos, testados, finalizados e prontos para uso. A figura 2.9 mostra um possível *layout* simplificado do quadro Kanban para esse exemplo.

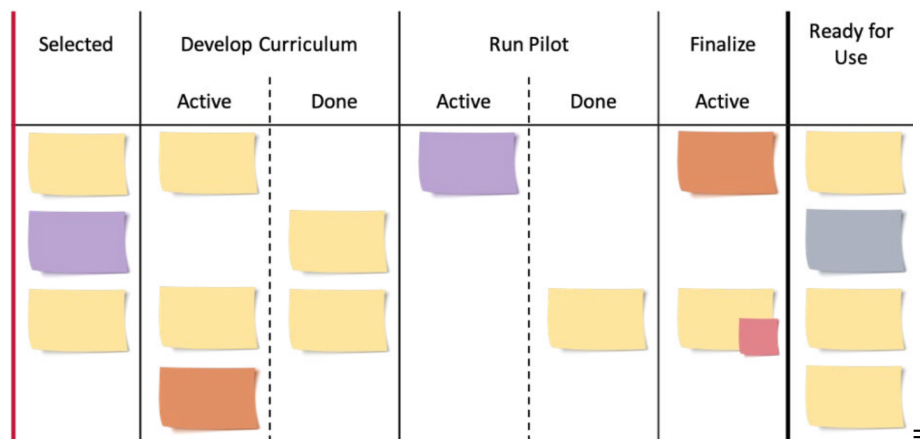


Figura 2.9. Exemplo de quadro do método Kanban [Kanban University 2021].

Um ponto importante a ser discutido sobre o Kanban, no cenário proposto nesse trabalho, é sua viabilidade em ser utilizado no contexto de projetos de PD&I. Embora seja possível se aplicar o Kanban no contexto de uma metodologia científica, a proposição de se ter o fluxo de trabalho real sendo modelado no quadro Kanban durante a execução do projeto fere um pouco a metodologia e princípios de uma pesquisa inovadora. Basicamente uma pesquisa engloba um mínimo planejamento prévio com base num levantamento sistemático e na definição de um conjunto de hipóteses. Com isso, a aplicabilidade do Kaban nesse cenário poderia ser incorporada durante o processo de confirmação ou rejeição das hipóteses, no caso nas sub-atividades de um processo de pesquisa.

2.3.3. A metodologia XP

Extraímos do livro *Extreme Programming Explained* de Kent Beck (criador do XP) e Cynthia Andres [Beck and Andres 2004] uma breve introdução, com os principais conceitos e definição relacionados à metodologia *Extreme Programming* (XP). Criado em 1990 e direcionado para o desenvolvimento do *software* o XP tem como proposta deixar de lado hábitos e padrões que foram interessantes no passado, mas que agora pode nos

impedir de fazer nosso trabalho melhor. XP é um estilo de desenvolvimento de *software* com foco na aplicação de técnicas de programação, comunicação clara e trabalho em equipe que nos permite realizar satisfatoriamente projetos complexos. XP considera uma filosofia de desenvolvimento de *software* baseada nos valores de comunicação, *feedback* e simplicidade. Engloba um conjunto de práticas comprovadamente úteis para melhorar o desenvolvimento de *software*. As práticas utilizadas se complementam, amplificando seus efeitos no resultado final do projeto. Ademais, o XP considera um conjunto de princípios complementares e técnicas para traduzir os valores em prática.

XP distingue-se de outras metodologias por considerar práticas simplificadas no decorrer de suas atividades. Dentre as diferentes práticas podemos destacar: ciclos de desenvolvimento curtos, resultando em *feedback* precoce, concreto e contínuo; planejamento incremental, que deve evoluir ao longo da vida do projeto; implementação imediata de funcionalidades que atendem às necessidades prioritárias de negócios e que estão em constante mudanças; dependência de testes automatizados escritos por programadores, clientes e testadores, permitindo o monitoramento do progresso do desenvolvimento e que o sistema evolua e detecte defeitos precocemente; dependência de um processo de design evolutivo que dura tanto quanto o sistema durar; e a dependência de práticas de curto prazo que atendam os interesses de longo prazo do projeto.

A definição base do XP é que “ele é uma metodologia leve para equipes pequenas e médias que desenvolvem *software* diante de requisitos vagos ou que mudam rapidamente”. No entanto, devido a sua difusão entre diferentes projetos e times, podemos ampliar essa definição considerando que: o XP é leve, nele você só faz o que precisa para criar valor para o cliente; XP é baseada no tratamento de restrições no desenvolvimento de *software*; XP pode ser aplicada à equipes de qualquer tamanho; e XP se adapta tanto a requisitos vagos que mudam rapidamente como também a requisitos que não são voláteis. Por fim, podemos resumir o uso do XP considerando as seguintes práticas:

- **Programação em pares:** É quando duas pessoas trabalham em uma única máquina. Enquanto um colaborador codifica, o outro faz sugestões pertinentes e em algum momento eles trocam de funções. Essa troca garante uma comunicação mais fácil, levando à maior produtividade e um projeto de alta qualidade.
- **Projeto simples:** Essa prática garante que a criação em excesso de generalizações dentro do código fonte seja evitada. Também prepara o sistema para possíveis mudanças e testes constantes.
- **Teste:** Cada funcionalidade de um projeto possui um teste antes de ser codificado, isso garante que não exista retrabalho entre os colaboradores e mantém a qualidade do projeto.
- **Refatoração:** É a mudança e melhoria do código, sem que seu comportamento e sua funcionalidade sejam alterados.
- **Propriedade coletiva:** Qualquer colaborador pode modificar códigos em todas as partes do projeto. Isso garante que, caso alguém saia da equipe, o projeto não seja prejudicado.

- **Interação contínua:** Os códigos são integrados e testados constantemente. Assim, caso algum problema for detectado será corrigido imediatamente.
- **Cliente presente:** O cliente deve estar presente nas etapas de aprovação, definição de prioridade e auxiliar no desenvolvimento do projeto. Semanalmente, o cliente deve ser atualizado e visualizar as novidades palpáveis. Espera-se que o projeto avance de uma entrega para a outra.
- **Semana de 40 horas:** Evita-se trabalhar mais de 40 horas semanais, visando o bem-estar da equipe para manter a qualidade do projeto.
- **Padrões de código:** O fato de todos os colaboradores participarem de todos os processos requer que um padrão na codificação seja mantido.
- **Metáfora:** Uma linguagem comum garante que o entendimento de ideias seja simplificado, para que ideias complexas sejam simplificadas.
- **Reunião diária:** Nela se discute o que foi feito anteriormente, o que está sendo feito no dia atual e o que ainda tem que ser feito.

Um ponto importante a ser discutido sobre o XP, no cenário proposto nesse trabalho, é sua viabilidade em ser utilizado no contexto de projetos de PD&I. Assim como para o Kanban, é possível se aplicar o XP no contexto de uma metodologia científica, especificamente se for uma atividade que envolva um time pequeno. A proposta do XP considera um baixo planejamento prévio e é voltada diretamente para implementação, logo sua utilização fere um pouco a metodologia e os princípios da metodologia científica tradicional. Como já discutido, uma pesquisa engloba um mínimo planejamento prévio e incorpora um conjunto de hipóteses associadas a um problema central. Com isso, a aplicabilidade do XP para projetos de PD&I poderia ser incorporada apenas no momento da implementação ou execução prática do projeto, no caso para uma pequena sub-atividade dentro do processo de pesquisa.

2.3.4. O *framework* Scrum

O Scrum, apesar de uma metodologia ainda bastante utilizada, foi criado em formalmente 1993 e constitui um conjunto específico de princípios e práticas ágeis para equipes multifuncionais e auto-organizadas em projetos de desenvolvimento de *software* [Sutherland 2014], podendo ser visto como uma ferramenta para desenvolver e manter produtos onde as pessoas conseguem resolver problemas complexos e adaptativos, ao mesmo tempo que entregam produtos com valor maximizado de forma criativa e produtiva. Apesar de ter surgido do ambiente de desenvolvimento de *software*, o Scrum vem sendo expandido para outros tipos de organizações e processos de gestão do conhecimento. Estudos recentes avaliam como a aplicação dos princípios do Scrum podem contribuir para a gestão colaborativa e coordenação de tarefas em processos de pesquisa científica [Hidalgo 2019].

As discussões em torno do Scrum datam da década de 80. Um artigo para a *Harvard Business Review* intitulado de “*The new new product development game*” exemplificava o *rugby* (esporte popular na Europa) como comparação para mostrar os benefícios da auto-organização de equipes no desenvolvimento de produtos inovadores [Takeuchi and

Nonaka 1986]. A partir daí, Jeff Sutherland, Ken Schwaber e Mike Beedle adaptaram as ideias do artigo para o desenvolvimento de *softwares* e criaram um novo método chamado “Scrum” [Sutherland 2014].

De acordo com o *Scrum Guide* [Schwaber and Sutherland 2020], esse *framework* é baseado em empirismo e *lean thinking*. O empirismo afirma que o conhecimento vem da experiência e da tomada de decisões com base no que é observado. O *lean thinking* - método focado na gestão com técnicas que valorizam o produto que chega ao cliente - reduz o desperdício e se concentra no essencial.

Tal qual qualquer metodologia ágil, o Scrum emprega uma abordagem iterativa e incremental para otimizar a previsibilidade e controlar o risco. Este *framework* envolve grupos de pessoas que, em conjunto, possuem todas as habilidades e conhecimentos necessários para realizar um certo trabalho e compartilhar ou adquirir essas habilidades a medida que seja necessário. O Scrum é baseado em um conjunto de papéis, eventos, artefatos e regras cada qual com um propósito. Quatro eventos formais são combinados para inspeção e adaptação, contidos dentro de um evento, conhecido por *Sprint*. Esses eventos funcionam porque implementam os pilares empíricos: i. (Transparência) Todo o processo deve estar visível a todos os envolvidos. Esta transparência deve ser refletida, no ambiente, nas pessoas e nos processos; ii. (Inspeção) O processo em si deve ser inspecionado regularmente na busca por anomalias e/ou oportunidades de melhorias; e iii. (Adaptação) Caso a inspeção detecte algum processo que precise ser ajustado ou melhorado, as adaptações deverão ser feitas o mais rápido possível. O quanto antes as mudanças sejam feitas, antes o novo processo proposto é testado e validado.

Além de seus pilares empíricos, o Scrum apresenta um conjunto de 5 valores (Figura 2.10⁵), são eles: Compromisso, foco, abertura, respeito e coragem. Esses valores orientam o time Scrum em relação ao seu trabalho, ações e comportamento. As decisões que são tomadas pelo time, os passos dados e a maneira como o *framework* é usado devem reforçar esses valores, não diminuí-los ou miná-los. A principal ideia em relação a estes valores é que os membros do time os aprendem e exploram à medida que trabalham com os eventos e artefatos do Scrum.

Como já mencionamos, no Scrum os ciclos iterativos do projeto são chamados de *Sprint*, onde todos os eventos acontecem. Cada evento é uma oportunidade formal para inspecionar e adaptar os artefatos associados ao projeto. Esses eventos são projetados especificamente para permitir a transparência necessária ao processo para que este seja flexível, eficaz e responda bem a quaisquer mudanças. A falha em operar quaisquer eventos conforme prescrito resulta em oportunidades perdidas de inspeção e adaptação. Os eventos são usados no Scrum para criar regularidade e minimizar a necessidade de reuniões imprevistas. O ideal é que todos os eventos sejam realizados no mesmo horário e local para induzir rotina e reduzir complexidades.

As *Sprints* são o núcleo do Scrum, são elas que implementam a metodologia adaptativa do gerenciamento que caracterizam o projeto dentro do modelo ágil. São nelas que ideias são transformadas em valor. Para o Scrum se caracterizar como uma metodologia adaptativa, as *Sprints* devem ter duração fixa de, no máximo, um mês, afim de imprimir

⁵<https://www.scrum.org/>



Figura 2.10. Valores do Scrum.

ritmo e gerar consistência no projeto. Uma nova *Sprint* deve iniciar imediatamente após o fechamento da anterior e durante sua execução não é feita nenhuma mudança que coloque em risco a sua meta. O *product backlog* (artefato que incorpora a meta do produto/solução a ser desenvolvida no projeto, detalhado mais adiante) é refinado conforme necessário; e o escopo pode ser esclarecido e renegociado com o *product owner* (responsável por maximizar o valor da solução resultante do trabalho do time e pelo gerenciamento eficaz do *product backlog*, detalhado mais adiante).

Os objetivos das *Sprints* devem ser bem definidos visando criar previsibilidade e garantir a inspeção e adaptação do progresso em direção a uma meta da solução com regularidade. As *Sprints* devem ser curtas, pois se muito longas, se tornarão inválidas, aumentarão a complexidade dos artefatos e, conseqüentemente, os riscos. *Sprints* mais curtas geram mais ciclos de aprendizagem e limitam os riscos de custo e esforço. Cada *Sprint* pode ser vista como um sub projeto de curta duração que entregará uma pequena solução em funcionamento.

Dentro das *Sprints* existem vários recursos de acompanhamento do trabalho e previsão de progresso, como *burn-downs*, *burn-ups* ou *cumulative flows*. Embora comprovadamente úteis, eles não substituem a importância do empirismo. Em ambientes complexos, o que acontecerá é desconhecido. Somente o que já aconteceu pode ser usado para a tomada de decisão voltada para o futuro.

Uma *Sprint* pode ser cancelada caso sua meta torne-se obsoleta. Contudo, só o *product owner* tem autonomia para cancelar uma *Sprint*. Os eventos relativos à *Sprint*, são: Planejamento, *daily meeting*, revisão e retrospectiva. Especificamente, o planejamento é o evento que inicia uma *Sprint* definindo o trabalho a ser realizado. Isto deve ser construído de forma colaborativa por todo o time. O *product owner* garante que os participantes estejam preparados para discutir os itens mais importantes do *product backlog* e como eles serão mapeados para a meta da solução. O time também pode convidar

outras pessoas para participar do planejamento para requerer conselhos. Nesse evento, são abordados tópicos como: i. Por que esta *Sprint* é valiosa? ii. O que pode ser feito nesta *Sprint*? e iii. Como o trabalho escolhido será realizado?

Para cada item do *product backlog* selecionado, os desenvolvedores planejam o trabalho necessário para criar um *increment* (é uma solução em funcionamento, um trampolim concreto em direção a meta do produto, detalhado mais adiante) que atenda à definição de “feito”. Isso geralmente é realizado decompondo itens do *product backlog* em itens de trabalho menores de um dia ou menos.

O outro evento importante numa *Sprint* é o *daily Scrum* (ou *daily meeting*). Seu propósito é acompanhar o progresso em direção a meta da *Sprint* e adaptar o *Sprint backlog* (meta do produto/solução a ser desenvolvida na *Sprint*, detalhado mais a seguir) conforme necessário, ajustando o próximo trabalho planejado.

A *daily Scrum* é um evento de 15 minutos para os desenvolvedores do time. Para reduzir a complexidade, é realizado no mesmo horário e local, todos os dias úteis da *Sprint*. Caso o *product owner* ou o *Scrum master* (responsável por estabelecer o ritmo ágil no time, detalhado mais adiante) estiverem ativamente envolvidos nos trabalhos dos itens do *Sprint backlog*, eles também participam desse evento. Os desenvolvedores conduzem a *daily Scrum* como acharem conveniente, desde que seu objetivo se concentre no progresso em direção a meta da *Sprint* e produza um plano de ação para o próximo dia de trabalho. Isso cria foco e melhora o auto gerenciamento. As *dailies Scrum* melhoram as comunicações, identificam os impedimentos, promovem a rápida tomada de decisões e consequentemente, eliminam a necessidade de outras reuniões.

O próximo evento importante numa *Sprint* é a revisão. Seu propósito é acompanhar o resultado da *Sprint* e estabelecer as adaptações futuras. O time apresenta os resultados de seu trabalho para o cliente e se discute o progresso em direção a meta da solução geral. Durante este evento, os participantes revisam o que foi realizado na *Sprint* e o que mudou em seu ambiente. Com base nessas informações, estes colaboram sobre o que fazer a seguir. O *product backlog* também pode ser ajustado para atender a novas oportunidades. A revisão da *Sprint* é uma sessão de trabalho e o time deve evitar limitá-la a simples apresentação.

Por fim, o último evento importante numa *Sprint* a ser considerado é a retrospectiva. Seu objetivo é planejar maneiras de aumentar a qualidade e a eficácia do trabalho. O time inspeciona como foi a última *Sprint* em relação a indivíduos, interações, processos, ferramentas e sua definição de “feito”. Os elementos inspecionados geralmente variam com o domínio de trabalho. O time discute o que deu certo durante a *Sprint*, quais impedimentos foram identificados e como esses impedimentos foram (ou não) resolvidos. O time identifica as mudanças para melhorar sua eficácia. As melhorias mais urgentes são realizadas o mais rápido possível. Essas podem até ser adicionadas ao *Sprint backlog* da próxima *Sprint*. Desta reunião, clientes e outros *stakeholders* não participam.

Além dos eventos considerados no Scrum, também existem os artefatos. Estes representam as atividade de trabalho ou valor entregue. Eles são projetados para maximizar a transparência das principais informações. Cada artefato contém um compromisso para garantir que ele forneça informações que aumentem a transparência e o foco pelo

qual o progresso do trabalho pode ser mensurado. São considerados artefatos do Scrum o *product backlog* que é a meta do produto/solução a ser desenvolvida no projeto; o *Sprint backlog* que é a meta do produto/solução a ser desenvolvida na *Sprint*; e o *increment* que trata-se da definição de “feito/pronto”. Esses compromissos existem para reforçar o empirismo e os valores Scrum para o time, clientes e outros *stakeholders*.

Especificamente o *product backlog* é uma lista ordenada e emergente do que é necessário para melhorar o produto/solução entregue ao final da execução do projeto. É a única fonte para o trabalho imprimido pelo time. Os itens do *product backlog* que podem ser realizados pelo time em uma *Sprint* são selecionados, desmembrados e refinados no evento de planejamento da *Sprint*. Esse refinamento trata-se do ato de quebrar e incluir definição adicional aos itens do *product backlog* para ter itens menores e mais precisos, que possam ser executados em um espaço de tempo mais curto. Esta é uma atividade contínua para adicionar detalhes, como descrição, ordem e tamanho. Os atributos geralmente variam de acordo com o domínio de trabalho. Os desenvolvedores que farão o trabalho são responsáveis pelo dimensionamento. O *product owner* pode influenciar os desenvolvedores, ajudando-os a entender e selecionar possíveis trocas de itens. O compromisso do *product backlog* é a meta do produto/solução, que descreve um estado futuro do produto que pode servir como um alvo para o time planejar. A meta do produto está no *product backlog* e é o objetivo de longo prazo para o time.

O outro artefato considerado é o *Sprint backlog* que é composto pela meta da *Sprint*, o conjunto de itens do *product backlog* selecionados e refinados para a *Sprint*, bem como um plano de ação para entregar o *increment*. Ele é um plano feito por e para os desenvolvedores. Este deve estar visível, em tempo real e refletir o trabalho que os desenvolvedores planejam realizar durante a *Sprint* para atingir a sua meta. O *Sprint backlog* é, portanto, atualizado ao longo da *Sprint* conforme a meta se aproxima. Essa lista de itens, deve ter detalhes suficientes para que eles possam inspecionar seu progresso na *daily Scrum*. O compromisso do *Sprint backlog* é a meta da *Sprint*. Embora esta seja compromisso dos desenvolvedores, ela fornece flexibilidade em termos do trabalho exato necessário para alcançá-la. A meta da *Sprint* também cria coerência e foco, encorajando o time a trabalhar junto ao invés de iniciativas separadas.

Um *increment* é uma solução em funcionamento, um trampolim concreto em direção a meta do produto. Cada *increment* é adicionado a todos os anteriores e completamente verificado, garantindo que todos funcionem juntos. A fim de fornecer valor, ele deve ser utilizável. Vários *increments* podem ser criados em uma *Sprint* e a soma deles é apresentada na revisão da *Sprint*, apoiando assim o empirismo. O trabalho não pode ser considerado parte de um *increment* a menos que atenda a definição de “feito/pronto”. O compromisso do *increment* é o “feito/pronto”

Por fim, a pedra fundamental do Scrum é o time de pessoas (*Scrum team*). Esse time é composto por papéis bem definidos e incluem: *Scrum master*, *product owner* e desenvolvedores. No *Scrum team*, não deve haver sub-times ou hierarquias. O time é uma unidade auto-gerenciável e coesa de pessoas focadas em um objetivo de cada vez: a solução em desenvolvimento. Os *Scrum teams* são multifuncionais e/ou multidisciplinares e são responsável por todas as atividades relacionadas à solução a ser desenvolvida, desde a colaboração com *stakeholder*, verificação, manutenção, operação, experimentação, pes-

quisa e desenvolvimento, e qualquer outra coisa que possa ser necessária. Eles são estruturados e empoderados pela organização (empresa, universidade, setores públicos, ou qualquer outro órgão que esteja adotando o Scrum como método ágil de gerenciamento de trabalho) para gerenciar seu próprio trabalho.

Os desenvolvedores são pessoas do time que estão comprometidas em criar qualquer aspecto de um *increment* utilizável a cada *Sprint*. Suas habilidades podem variar, mas incluem: i. Criar um plano para a *Sprint*, o *Sprint backlog*; ii. Introduzir gradualmente qualidade aderindo a uma definição de “feito” sobre o *Sprint backlog*; iii. Adaptar seu plano a cada dia em direção à meta da *Sprint*; e iv. Responsabilizar-se mutuamente como profissionais.

O *product owner* é responsável por maximizar o valor da solução resultante do trabalho do time e pelo gerenciamento eficaz do *product backlog*, que inclui: i. Desenvolver e comunicar explicitamente a meta do produto; ii. Criar e comunicar claramente os itens do *product backlog*; iii. Ordenar os itens do *product backlog*; e iv. Garantir que o *product backlog* seja transparente, visível e compreensível.

O *Scrum master* é responsável por estabelecer o ritmo ágil no time e serve ao time da seguinte maneira: i. Treina os membros do time em auto-gerenciamento e funcionalidade cruzada; ii. Ajuda o time a se concentrar na criação de *increments* de alto valor que atendem à definição de “Feito”; iii. Provoca a remoção de impedimentos ao progresso do time; e iv. Garante que todos os eventos Scrum ocorram e sejam positivos, produtivos e mantidos dentro dos ciclos incrementais do projeto. O *Scrum master* serve ao *product owner* da seguinte maneira: i. Ajudar a encontrar técnicas para a definição eficaz de meta do produto e gerenciamento do *product backlog*; ii. Ajudar o time a entender a necessidade de itens do *product backlog* claros e concisos; iii. Ajudar a estabelecer o planejamento empírico do produto para um ambiente complexo; e iv. Facilitar a colaboração dos clientes e *stakeholder*, conforme solicitado ou necessário. Por fim, o *Scrum master* serve à organização ao qual está vinculado da seguinte forma: i. Liderar, treinar e orientar a organização na adoção do Scrum; ii. Planejar e aconselhar implementações de Scrum dentro da organização; iii. Ajudar os funcionários e os *stakeholders* a compreender e aplicar uma abordagem empírica para trabalhos complexos; e iv. Remover barreiras entre *stakeholders* e *Scrum teams*.

Ainda sobre os papéis dentro do time, o *product owner* é uma pessoa, não um comitê. Ele pode representar as necessidades dos clientes e/ou *stakeholders* no *product backlog*. Aqueles que desejam alterar o *product backlog* podem fazê-lo tentando convencer o *product owner*. O *Scrum team* deve ser pequeno o suficiente para permanecer ágil e grande o suficiente para ser eficiente dentro de uma *Sprint*, normalmente o time é composto de 10 ou menos pessoas. Se os times se tornarem muito grandes, eles devem considerar a reorganização em vários times coesos, cada um focado no mesmo produto. Em suma, a Figura 2.11⁶ ilustra o andamento de uma *Sprint* com todos os seus papéis, eventos e artefatos envolvidos.

Considerando o cenário proposto nesse trabalho, diferente das demais abordagens o Scrum é uma metodologia de fácil adaptação às metodologias científica, logo torna-se

⁶<https://www.scrum.org/>

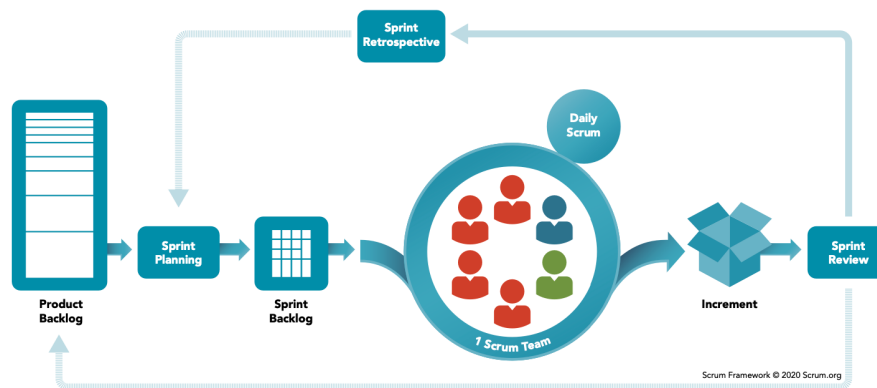


Figura 2.11. Framework Scrum.

um opção atrativa para projetos de PD&I. Sua proposta em desenvolvimento baseado em *Sprints*, considerando sua revisão e planejamento em ciclos curtos, favorecem a produtividade do time e permite que as abordagens para o desenvolvimento da pesquisa sejam verificadas e alteradas em tempo hábil sem maiores prejuízos. Com isso, a aplicabilidade do Scrum para projetos de PD&I pode ser considerando todo o processo de pesquisa, onde os insumos podem variar de um relatório com o catálogo dos trabalhos relacionados ao problema até avaliação em grande escala de experimentos previamente realizados. Com isso, o Scrum irá prevalecer como principal metodologia ágil aqui abordado e será a base para nossa explanação sobre o melhor uso dos métodos ágeis no contexto de projetos de PD&I.

2.4. Estratégias ágeis aplicadas à pesquisa científica

Segundo o Manual de Oslo [OECD and Eurostat 2018], principal fonte internacional de diretrizes para coleta e uso de dados sobre atividades inovadoras do setor produtivo, existem quatro categorias principais de fatores que têm relação primária com a inovação: i. As empresas comerciais; ii. As instituições dedicadas a ciência e tecnologia; iii. As questões de transferência e absorção de tecnologia, conhecimentos e habilidades; e iv. Conjunto de fatores definido pelo o ambiente que cerca as instituições, sistemas jurídicos, o contexto macroeconômico e outras condições que independem de quaisquer considerações sobre inovação. O mapa exibido na Figura 2.12 rotula esses quatro domínios gerais.

A partir desse mapa, iremos, portanto, chamar atenção do leitor para três questões específicas:

1. **A base de ciência e engenharia:** Corresponde ao conhecimento acumulado e as instituições de ciência e tecnologia que sustentam a inovação comercial, fornecendo treinamento tecnológico e conhecimento científico;
2. **Os fatores de transferência:** Influenciam fortemente a eficácia dos elos de fluxo de informações e competências e absorção de aprendizado, essenciais para a inovação comercial; e

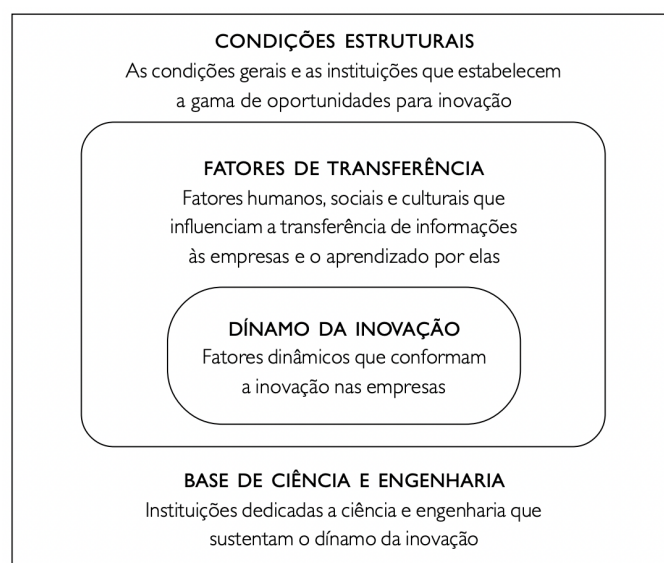


Figura 2.12. O Campo das Políticas de Inovação — Um Mapa das Questões [OECD and Eurostat 2018].

3. **O dínamo da inovação:** Corresponde ao domínio mais central da inovação comercial — ele cobre fatores dinâmicos dentro das empresas ou em sua vizinhança imediata que têm um impacto muito direto em sua capacidade inovadora.

Entre os elementos da base nacional de ciência e engenharia estão: o sistema de universidades; o sistema de apoio à pesquisa básica; boas atividades públicas e estratégicas de de PD&I. Ademais, dentre os fatores de transferência, encontra-se a facilidade que o setor produtivo tenha de acesso à PD&I. A partir dessas premissas já consolidadas, podemos levantar a seguinte questão:

O que impulsiona a colaboração entre institutos de pesquisa e o setor produtivo, excelência em pesquisa ou estratégias de colaboração?

Esta pergunta foi o tema de um artigo recente que abordou importantes aspectos na relação entre a universidade e o setor produtivo (UIC) [Atta-Owusu et al. 2021]. Neste trabalho, foram analisados dados da *Community Innovation Survey* (CIS) para empresas norueguesas e dados da base *Scopus* sobre a produção de pesquisa das universidades norueguesas em várias disciplinas. De acordo com os autores, a política de projetos de PD&I visa aumentar a produção de pesquisa e a UIC, a fim de permitir que as empresas tenham acesso ao conhecimento científico de ponta desenvolvido nos centros de pesquisa das universidades. Os autores partem da premissa de que, como parte de sua missão, espera-se que as universidades de muitos países contribuam localmente para a inovação em suas regiões.

No entanto, ainda segundo Atta-Owusu et al. [Atta-Owusu et al. 2021], a relação entre a produção de pesquisa científica e a UIC ainda não está suficientemente clara: as universidades podem produzir pesquisas que expandem a fronteira do conhecimento e estas tornam-se atraentes para as empresas, porém, há uma grande lacuna entre a pesquisa

produzida e as necessidades das empresas locais. Isso pode prejudicar a capacidade das empresas locais de cooperar totalmente com as universidades ou forçá-las a procurar além da sua região outras universidades adequadas para interagir.

Os resultados da análise mostram, de maneira curiosa, que a qualidade da pesquisa das universidades locais se relacionam negativamente com a propensão das empresas em colaborar com as universidades. Isso indica que a pesquisa, orientada à excelência promovida pelas universidades pode ter menos impacto na transferência imediata de conhecimento para a sociedade do que a ênfase das políticas colaborativas. Em outras palavras, buscar a excelência em pesquisa, em vez de maximizar os benefícios da pesquisa para os sistemas socioeconômicos locais no curto prazo, pode custar a colaboração entre institutos de pesquisa e o setor produtivo. Por outro lado, a tendência das empresas de colaborar com outros parceiros externos, como fornecedores, clientes ou consultores, é um dos principais impulsionadores da UIC.

As universidades são muitas vezes consideradas, especialmente em muitas estratégias regionais de inovação e desenvolvimento, como um farol para a inovação nas empresas. Parcerias universidades e o setor produtivo estão sendo recomendadas como forma melhorar a qualidade do processo de inovação, a produtividade e a competitividade das empresas [Atta-Owusu et al. 2021]. No entanto, observou-se que esta colaboração é muitas vezes evasiva. Uma razão para isso pode ser que, embora a busca de excelência em pesquisa seja desejável para a sociedade como um todo, pode existir ainda uma lacuna entre a produção científica e as necessidades das empresas. Desta forma, podemos considerar que projetos de PD&I envolvendo centros de pesquisa devem considerar, além da excelência na qualidade da pesquisa desenvolvida, estratégias que estreitem os laços com o setor produtivo que exige soluções de curto e médio prazo. Dentre as estratégias que permitem uma melhor colaboração entre ambos os setores, encontra-se a aplicação de métodos ágeis de gerenciamento de projetos e pessoas, que vem sendo estudados e implementado com cada vez mais frequência [Hidalgo 2019].

2.4.1. Métodos ágeis e o gerenciamento de projetos de PD&I

Alessandro di Fiori et al. [di Fiori et al. 2019], fundador e CEO do *European Centre for Strategig Innovation*, nos deu uma estrutura de como mais organizações podem aderir ao movimento da ciência ágil:

- **Antecipação do ceticismo:** As percepções são importantes, especialmente em um campo infundido com um senso natural de ceticismo, como é o campo da pesquisa científica. Os cientistas são treinados para confiar apenas em experimentos controlados. Para alterar qualquer visão negativa inicial sobre o uso de metodologias ágeis, se faz necessária uma abordagem eficiente para divulgar e executar os conceitos.
- **Ênfase nos “porquê”:** Concentrar-se mais no “*porquê*” da mudança para o ágil do que no “*o que*” e “*como*” incorporar a metodologia. O consenso em torno das metas deve impulsionar decisões sobre qual modelo organizacional e práticas devem ser aplicadas. Há várias razões para uma transformação ágil: reduzir custos, aumentar a produtividade, tornar-se mais centrado no cliente, adaptando-se rapi-

damente aos *feedbacks*, acelerando a tomada de decisões, aumentando o envolvimento dos funcionários e assim por diante. Quando esse “*porquê*” é identificado e comunicado, as pessoas são muito mais propensas a aceitar e até mesmo acolher a mudança.

- **Implementação flexível:** Métodos ágeis não requerem uma abordagem de “*tudo ou nada*”. Algumas rotinas e ferramentas ágeis podem ser muito valiosas e aplicáveis a contextos específicos, outras menos. Tudo depende da especificidade e das necessidades de negócios. A rigidez do processo é a antítese da agilidade. As equipes ágeis de PD&I podem não — quase certamente não — se parecer com equipes de *software* ágeis, e isso é perfeitamente aceitável.
- **Organização em torno do tipo certo de equipes:** As equipes são a unidade básica de entrega para uma organização que implementa modelos ágeis. Mas o tipo certo de organização baseada em equipe é difícil de encontrar em negócios orientados pela ciência. As pessoas são designadas para projetos, mas continuam a se identificar com funções e departamentos ferozmente independentes que usam seu domínio de conhecimento científico como forma de assumir o controle de “decisões relacionadas”. Além dos silos organizacionais típicos, os domínios de conhecimento criam feudos, removidos de qualquer questionamento por pessoas de fora.

Ainda segundo Fiori et al. [di Fiori et al. 2019], o uso do ágil na ciência está apenas começando. Comunidades científicas entrincheiradas resistirão à mudança de seus hábitos de trabalho. Mas, à medida que organizações pioneiras relatam seus resultados e experiências positivas, a demanda por ciência ágil provavelmente aumentará. Aprender e adaptar está, afinal, no centro do que se trata a ciência. A inovação é um processo de aprendizagem, assim como o método científico.

2.4.2. Metodologia de pesquisa ágil Ny

Essa metodologia é resultado de uma longa experiência dos autores deste trabalho (10 anos aproximadamente), atuando na coordenação de projetos de pesquisa, tanto acadêmica como em conjunto com empresas. A *Ny-methodology* considera a integração entre a metodologia de pesquisa tradicional (Figura 2.13) e o *framework* Scrum como base para o desenvolvimento da pesquisa (Figura 2.14).

É utilizado a dinâmica do Scrum para o desenvolvimento e avanço das atividades de pesquisa tradicional. Nesse sentido, temos que as seguintes etapas de pesquisa a serem seguidas, com o seu respectivo enquadramento no Scrum:

- **Identificação do problema:** Esta etapa consiste de um processo de mapeamento e extração do máximo de informações do contratante referentes a necessidade do produto de inovação tecnológica. Essa etapa possui como insumo um relatório com a descrição detalhada do problema e sua execução pode durar de 1 a 3 *Sprints*, considerando estas em janelas de 15 dias.
- **O que existe em termos de soluções:** Busca das principais soluções existentes para o problema. Consulta ao estado da arte baseado nos princípios de revisão sistemática envolvendo artigos, patentes e soluções de mercado. Essa etapa possui como

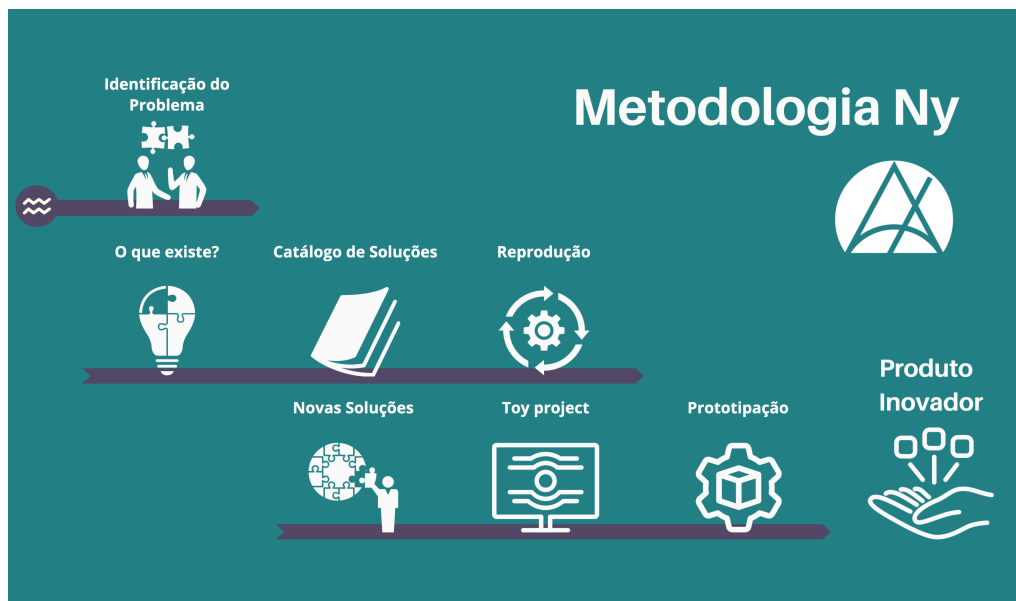


Figura 2.13. Metodologia Ny. Fonte: Os próprios autores.

insumo um relatório incremental (todos os relatórios são incrementais e seguem um modelo de apresentação) com o resultado da revisão sistemática considerando os artigos encontrados. Esta etapa pode ser executada entre 4 e 6 *Sprints*. No caso do problema ter vários desdobramentos (subproblemas), ou seja um problema bem estabelecido na literatura que demande a leitura de muitos artigos, recomendamos, respectivamente, que se quebre a revisão do problema central em sub-revisões, no caso uma para cada subproblemas; ou que se divida os artigos entre os membros mais experientes do time, por exemplo, uma divisão em diferentes janelas de tempo.

- **Catálogo de Soluções:** Listagem e documentação das abordagens encontradas, ou seja, definição do *benchmark*, recomendamos separar para a próxima etapa até três soluções por revisão realizada. Essa etapa possui como insumo a lista com o detalhamento das soluções mais próximas ao problema identificado e sua execução pode durar de 1 a 2 *Sprints*.
- **Reprodução das melhores estratégias:** Aprofundamento das soluções catalogadas e que são mais aderentes ao problema e eleição da solução a ser reproduzida. Recomendamos a reprodução de uma ou no máximo duas soluções de todas catalogadas. Essa etapa possui como insumo um conjunto de resultados de experimentos que atestam a fiel reprodução dos métodos listados. Por sua complexidade, costuma ser a etapa mais custosa de toda a metodologia e sua execução pode durar de 4 a 8 *Sprints*.
- **Novas soluções:** Avançar no estado da arte para tornar os produtos dos clientes ainda mais competitivos. Neste etapa, podemos desenvolver uma nova técnica propriamente dita, a partir da experiência já adquirida com o problema, ou mesmo criar um mecanismo agrupando soluções já existentes para problemas similares. Essa etapa possui como insumo uma API das soluções implementadas, a fim de se

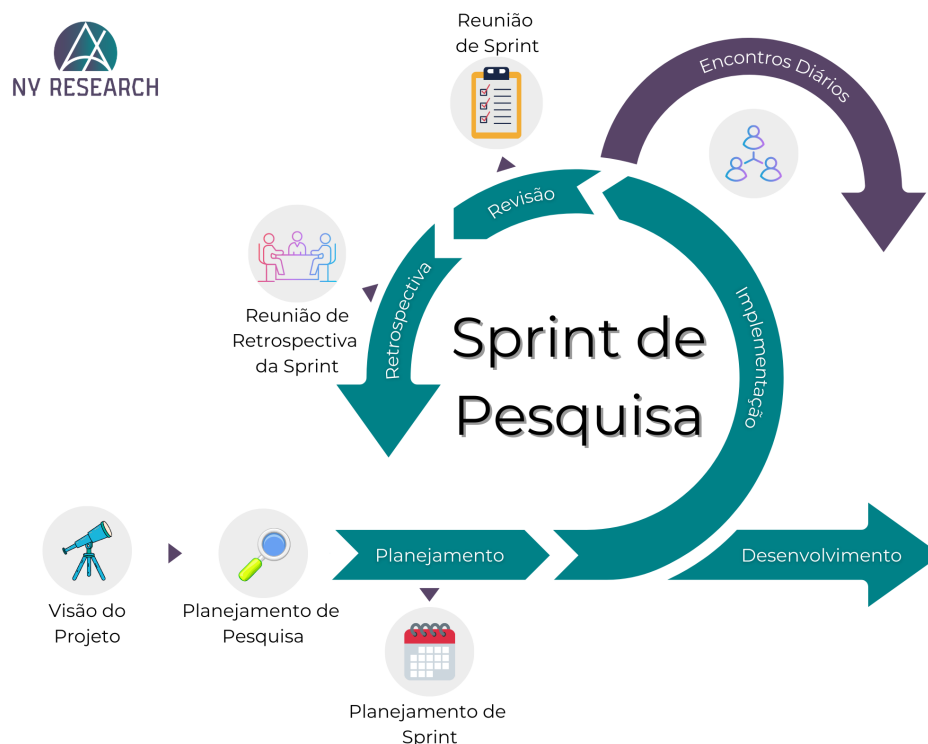


Figura 2.14. Scrum adaptado para a PD&I na metodologia Ny. Fonte: Os próprios autores.

permitir a execução dos testes na próxima etapa, e sua execução pode durar de 4 a 6 *Sprints*.

- **Toy project, teste e validação:** Execução da solução encontrada em um ambiente similar ao de produção, executando testes para validação qualitativa e quantitativa da nova solução apresentada. Essa etapa possui como insumo um relatório de execução, teste e validação tanto das soluções reproduzidas como da nova solução implementada e tem como foco a identificação de gargalos e outros possíveis problemas que possam inviabilizar o uso da solução desenvolvida por parte do setor produtivo e solucionar esses percalços. Essa etapa pode ter sua execução durando de 3 a 6 *Sprints*.
- **Prototipação e consolidação:** Disponibilização de uma API para ser incorporada ao produto do cliente e acompanhamento dos testes em produção, permitindo ajustes finos no ambiente de produção ou retroalimentando os próximos ciclos de execução. Essa etapa possui como insumo a API validada, absolutamente funcional, o relatório de sucesso e a documentação da API desenvolvida. Finalizamos o desenvolvimento com essa etapa que pode durar de 2 a 3 *Sprints*.

Como destacado, as etapas acima são desenvolvidas em *Sprints* de 15 dias, logo podem durar de 15 a 60 dias a depender da complexidade do projeto e disponibilidade de uma equipe do setor produtivo para acompanhar o desenvolvimento da solução. Em particular as etapas “novas soluções”, “toy project” e “prototipação”, são executadas aninhadas em vários ciclos até atingir a maturidade desejada pelo pesquisador, que assume papel de

Scrum Master na execução do projeto. Um ciclo trata-se de uma janela de 3 *Sprints*, separadas ou intercaladas, a depender do projeto, ou seja temos a sequência “novas soluções – *toy project* – prototipação”, que se repete até atingirmos a maturidade desejada. Assim, essa estratégia permite que a solução entre rapidamente em produção e que ajustes sejam feitos de forma imediata.

Nossa adaptação do Scrum para uma metodologia de pesquisa considera o desenvolvimento em *Sprints* de pesquisa que se retroalimentam e possuem as seguintes etapas: i. Planejamento e composição do *backlog* da *Sprint*; ii. Implementação que pode ser desde a atividade de codificação da solução, geração de um relatório de revisão bibliográfica ou realização de cálculos para a proposição de um modelo matemático; iii. Revisão do que foi implementado, no meio da *Sprint* para que se avalie o progresso das atividades em andamento; e iv. Retrospectiva do que foi desenvolvido e que se discute com toda a equipe os progressos realizados e percalços encontrados durante a *Sprint* de pesquisa. A nossa metodologia considera as seguintes atividades durante uma *Sprint* (Figura 2.14):

- **Visão do projeto:** Consiste na definição dos papéis, identificação do problema, montagem de infraestrutura, nívelmaneto da equipe, dentre outras atividades inerentes à pesquisa a ser realizada. A chamamos de *Sprint zero*.
- **Planejamento da pesquisa:** Consiste na consolidação do que foi definido na *Sprint zero* combinada com a composição do *backlog* do produto (ou da pesquisa). É nesse ponto que todos os artefatos e ferramentas de gestão de projetos são adaptados e configurados (falaremos um pouco mais deles adiante).
- **Planejamento das Sprints:** Durante a execução do projeto teremos várias *Sprints* de pesquisa, nesse planejamento avaliamos quais as atividades do *backlog* do produto irá para o *backlog* da *Sprint*.
- **Encontros diários:** Atividade inerente a boa parte das metodologias ágeis. Nesses encontros avaliamos o que foi realizado no dia anterior e o que será realizado no dia corrente. Com esses encontros é possível nivelar e avançar mais assertivamente na pesquisa, uma vez que espera-se que a equipe seja heterogênea (graduados, mestres e doutores) a ponto que orientações pontuais possam ser dadas. Essa integração permite uma maior sinergia entre o time.
- **Revisão da Sprint de pesquisa:** Essa atividade ocorre ao final de cada *Sprint* e permite a identificação dos progressos realizados, percalços encontrados e é feita com os membros internos do projeto e o cliente, permitindo que seja dado um retorno do que foi feito e da identificação das atividades que podem entrar na próxima *Sprint*.
- **Retrospectiva da Sprint de pesquisa:** Essa atividades é a continuação da revisão, porém apenas com os membros internos do projeto. Tais atividades pode ser continuação das atividades da *Sprint* anterior, caso a mesma apresente algum percalço significativo; atividades já presentes no *backlog* do produto que venha a agregar valor significativo ao projeto; ou novas atividades que foram identificadas durante a revisão e retrospectiva e que antes não havia sido considerada.

Para facilitar a utilização da Metodologia Ny, disponibilizamos um conjunto de artefatos que podem ser utilizados e adaptados para projetos específicos. Basicamente, utilizamos uma apresentação em slides (Figura 2.15) para compor o relatório das atividades realizadas e facilitar a comunicação entre o time e o cliente; e o Trello⁷ como ferramenta básica de organização da execução do projeto (Figuras 2.16–2.18).



Figura 2.15. Organização dos slides que são utilizados nas reuniões de retrospecto.

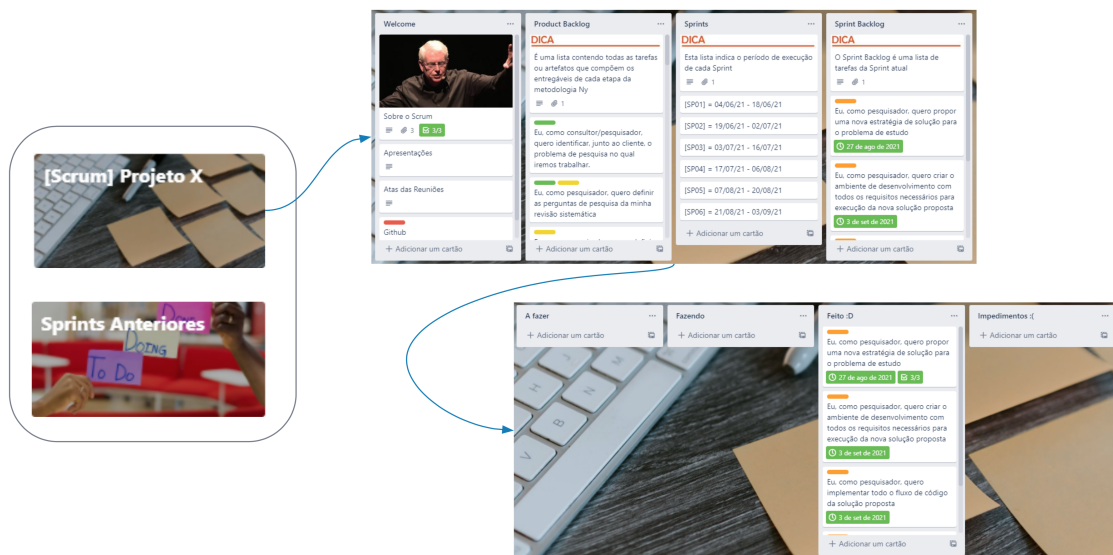


Figura 2.16. Cartões para identificação das estórias e seus respectivos responsáveis.

Dentro da Metodologia Ny consideramos os papéis tradicionais: Desenvolvedores que são os estudantes de graduação e pós-graduação, profissionais contratados especificamente para atuar no projeto ou profissionais disponibilizados pelo contratante para participar do desenvolvimento; *Scrum master* que é um pesquisador mais experiente, doutores ou pós doutorandos que possuem disponibilidade integral para atuar na gestão do projeto e garantir que a *Sprint* avance com sucesso, permitindo mais celeridade na geração dos incrementos; e *product owner* que é o representante técnico coordenador do projeto. Para além destes, incluímos o papel do *pesquisador consultor* que possui experiência na temática do problema a ser desenvolvido e atua de forma transversal auxiliando nas atividades de pesquisa. Esse pesquisador atua como garantidor da execução do objeto de pesquisa investigado.

⁷<https://www.trello.com/>



Figura 2.17. Detalhe de um cartão.

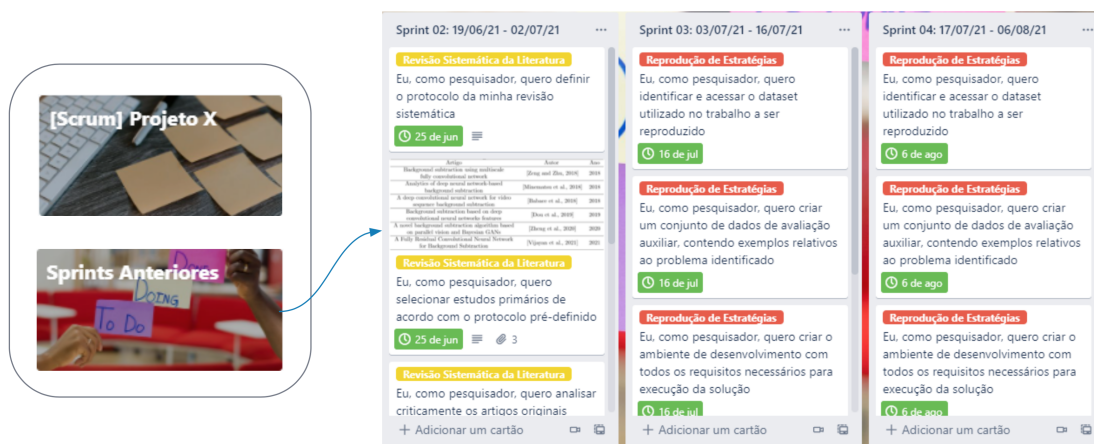


Figura 2.18. Cartões que registram o histórico das Sprints.

2.5. Estudos de casos

Velocidade e capacidade de resposta são essenciais para o sucesso no desenvolvimento de *software*, pois os pré-requisitos estão mudando continuamente. Sob essas circunstâncias, a compreensão profunda do desenvolvimento do *software* desejado requer estar em meio às situações que estão ocorrendo. Fazer pesquisa científica no contexto de *software* e dentro das empresas requer, portanto, uma estreita colaboração entre os pesquisadores e os profissionais envolvidos, tendo então conhecimento do problema que se deseja suprimir ou processo a ser aprimorado. Portanto, é importante encontrar maneiras práticas de facilitar a colaboração que será fundamental para que a solução desenvolvida atenda às necessidades do setor produtivo e tenha aderência ao ambiente de produção. Entretanto, mesmo se compreendendo a necessidade dessa colaboração, ainda é algo desafiador estabelecê-la de forma consistente.

Tanto o setor produtivo quanto a academia começam com um problema ou desa-

rio que não é bem compreendido, trabalham em um ambiente em constante mudança e os objetivos são alcançados por meio de etapas iterativas. Todas essas semelhanças na execução dos processos têm se tornado cada vez mais uma motivação justa para utilização de metodologias ágeis como terreno comum para uma colaboração bem-sucedida em pesquisa entre o setor produtivo e academia.

Nesta seção iremos apresentar dois relatos da utilização bem sucedida de estratégias ágeis para a execução de projetos de PD&I em parceria com o setor produtivo. Um deles é o sobre a adaptação do Scrum para gestão na ciência realizado pelo *Centro para a Avaliação da Complexidade Através do Nexus* (CECAN⁸) e o outro é sobre a gestão de projetos de forma ágil realizado pelo *Laboratório de Computação Científica e Análise Numérica* (LaCCAN⁹) e que culminou na Metodologia Ny apresentada anteriormente.

2.5.1. Estudo de caso 1: Adaptação do Scrum para gestão na ciência – CECAN

Como primeiro estudo de caso, iremos relatar a experiência de Enric Senabre Hidalgo [Hidalgo 2019] que explorou a adoção de métodos ágeis para o gerenciamento de projetos em iniciativas de pesquisa colaborativa. O relato teve como cenário o *Centro para a Avaliação da Complexidade Através do Nexus* (CECAN¹⁰), sediado no Reino Unido. O estudo mostrou em que medida os princípios e ferramentas-chave do Scrum podem contribuir para a gestão colaborativa e coordenação de tarefas em processos associados à pesquisa científica. As respostas de entrevistas com 17 pesquisadores, bem como o acompanhamento e análise de atividades *online*, foram examinadas e apresentadas como um estudo de caso sobre a adoção de práticas de Scrum no CECAN, centro de pesquisa distribuída dedicado à avaliação de políticas públicas. O autor motiva sua pesquisa diante dos seguintes fatores: i. A literatura sobre práticas de pesquisa indica que o trabalho em equipe e a colaboração dominam a produção de conhecimento em organizações acadêmicas e são predominantes em redes internacionais de pesquisa de grande escala [Cooke and Hilton 2015]; ii. Os benefícios da colaboração em pesquisa vão desde um aumento nas citações como resultado da coautoria de artigos até um melhor uso dos recursos disponíveis para a pesquisa [Ynalvez and Shrum 2011]; e iii. A capacidade de gerar um impacto social mais amplo por meio de projetos de pesquisa em larga escala [Bammer 2008].

O foco esteve na apropriação do Scrum como referencial metodológico e seu uso experimental na gestão de iniciativas de pesquisa distribuída e interdisciplinar. Procurou-se, portanto, identificar as experiências e percepções de pesquisadores na adoção de princípios ágeis, mostrando assim seus potenciais benefícios e limitações no cenário da pesquisa colaborativa. Nesse estudo, o autor buscou responder às seguintes questões de pesquisa: i. Quais condições favorecem a apropriação do gerenciamento de projetos ágeis para colaboração em pesquisa? ii. Até que ponto os princípios e ferramentas específicas do Scrum podem ser adotados em contextos interdisciplinares? e iii. Quais são as limitações e vantagens de adaptar métodos ágeis em uma organização de pesquisa distribuída?

O estudo utilizou três abordagens metodológicas: observação participante, análise da atividade *online* e entrevistas semiestruturadas. Essa combinação de abordagens cons-

⁸<https://www.cecan.ac.uk/>

⁹<https://www.laccan.ufal.br/>

¹⁰<https://www.cecan.ac.uk/>

tituiu a base para a análise da adoção de princípios e práticas ágeis e do *framework* Scrum no CECAN. Um período de seis meses de observação participante de várias atividades realizadas pelo CECAN resultou na geração de um banco de dados de notas de observação para análise.

Os dados da pesquisa foram codificados para temas emergentes e, posteriormente agrupados em três áreas de investigação relacionadas às questões de pesquisa: i. Condições para adoção de métodos ágeis em pesquisa; ii. Adoção de práticas e ferramentas de Scrum; e iii. Limitações e vantagens da adoção de métodos ágeis em um ambiente distribuído.

Um ponto importante a observarmos foi a adaptação do Scrum pra esse contexto específico: O papel do *Scrum master* foi amplamente adotado. No CECAN, a função foi concebida como “coordenação de estudos de caso”. Os *Scrum masters* do CECAN viam seu papel como o elo entre tarefas e objetivos específicos e outros pesquisadores colaboradores, bem como a ligação com formuladores de políticas e representantes de agências governamentais. Esse papel fundamental foi desempenhado por pesquisadores do CECAN em vez de *Scrum masters* profissionais, e foi focado na coordenação, facilitando conexões e fornecendo diretrizes para estudos de caso específicos. Nesse sentido, o papel do *Scrum master* pode ser considerado uma apropriação e reinterpretação. Foi uma das principais práticas ágeis adotadas no CECAN e, embora percebida por alguns participantes como não totalmente implementada, o *Scrum master* pareceu desempenhar um papel crítico de facilitação e contribuiu para expandir as práticas de gestão ágil para as diversas iniciativas e projetos de pesquisa.

O quadro Kanban foi uma das práticas de gestão ágil adotada. Os quadros do CECAN eram digitais e criados no Trello ¹¹ e foram um dos principais canais de documentação. Eles eram gerenciados principalmente pelos *Scrum masters*, e eram acessados pelos demais pesquisadores do CECAN e ocasionalmente por colaboradores externos ou outros *stakeholders*.

Segundo o estudo, o conceito de *Sprints* foi menos explícito ou utilizado entre os participantes. Por exemplo, a prática de estabelecer reuniões regulares ou reuniões retrospectivas no final de cada período de *Sprint* não era seguida rotineiramente. Em vez disso, os pesquisadores geralmente estabelecem acordos coletivos sobre a duração e as responsabilidades relacionadas a tarefas específicas, dependendo do projeto.

O conceito de desenvolvimento incremental por lançamentos de “produtos mínimos viáveis”, quando adotado inicialmente a partir do desenvolvimento de *software*, foi percebido como outra abordagem complexa a ser abordada no contexto da pesquisa acadêmica. Contudo, vale observarmos que o tipo de pesquisa conduzida nesse estudo de caso não estava direcionada a implementação de protótipos tecnológicos nem soluções computacionais, mas se referiu a projetos dentro da área de avaliação de políticas públicas.

¹¹<https://www.trello.com/>

2.5.2. Estudo de caso 2: Gestão de projetos de forma ágil – LaCCAN

O Laboratório de Computação Científica e Análise Numérica (LaCCAN¹²) da Universidade Federal de Alagoas (UFAL) tem como base a computação científica, de forma interdisciplinar e transversal atuando principalmente na área de Ciência dos Dados aplicada aos problemas de Cidades inteligentes. Propomos diferentes soluções para problemas de análise e processamento de grandes volumes de dados (texto e imagem), sistemas de transporte inteligente, monitoramento ambiental, segurança de informação em redes de computadores, análise de anomalias em fluxo de documentos, identificação de padrões em imagens de satélite entre outros. Nos últimos cinco anos, fomos responsáveis pela captação de um grande volume de recursos em projetos de PD&I. Tais projetos financiados por diversas agências de fomento à pesquisa, órgão públicos e empresas.

Dada a necessidade de um gerenciamento prático e assertivo dos projetos e que levasse à resultados rápidos, ao longo dos anos fomos adaptando a utilização de métodos ágeis para o desenvolvimento dos diferentes projetos executados pelo laboratório. Para tal consideremos os seguintes cenários de aplicação do método ágil para o desenvolvimento da pesquisas no LaCCAN: pesquisas individuais envolvendo projetos com estudantes de graduação, mestrado e doutorado; pesquisa como atividade fim, financiadas a fundo perdido pelas agências de fomento; e pesquisa como atividade meio, financiadas por empresas públicas ou privadas destinadas à resolução de problemas do setor público e/ou produtivo.

No contexto do LaCCAN a necessidade de se utilizar estratégias ágeis surge com as pesquisas individuais, vivenciada pelos pesquisadores do laboratório que incorporam a utilização de um método em espiral dentro da pesquisa científica. Para uma pesquisa individual o problema e as hipóteses são bem definidas a priori, logo podemos simplificar a pesquisa individual em quatro atividades básicas: i. Levantamento bibliográfico; ii. Reprodução das soluções; iii. Proposição de novas soluções; e iv. Teste e análise. Nesse sentido a utilização do espiral se dá entre as atividades iii e iv, ou seja, desenvolve-se a solução mínima viável e vai incrementando a solução de acordo com o problema definido. Para essa estratégia consideramos cada ciclo do espiral como sendo uma contribuição científica fechada, ou seja, um artigo. Para a pesquisa individual, os atores são os orientadores, o estudante orientado e eventualmente outros pesquisadores que auxiliam o desenvolvimento trabalho. Não existem prazos bem definidos, porém o conceito de *Sprints* (com duração de 2 semanas) sempre foi adotado. Realizamos reuniões semanais, uma semana envolvendo a revisão e planejamento da Sprint e na semana seguinte o acompanhamento das atividades. O único prazo é o para a entrega do relatório final (trabalho de graduação), dissertação ou tese, que nesse caso varia entre um e quatro anos. O gerenciamento e acompanhamento é feito com base no Trello e em apresentações de slides como sugerido pela Metodologia Ny.

Para além das pesquisas individuais, temos as pesquisa como atividade fim, ou seja, o desenvolvimento de projetos de pesquisa financiados por agências de fomento. Nesse caso, incorporamos uma gestão ágil combinada com uma execução em modo espiral. Esse tipo de pesquisa, muitas vezes distribuída, envolve pesquisadores e grupos distintos (nacionais e internacionais), nesse caso o objetivo do projeto são os resultados

¹²<https://www.laccan.ufal.br/>

finais considerando uma execução independentes, porém com uma gestão centralizada. Para a gestão ágil desses projetos consideramos ciclos de 2 a 6 meses, tendo como principal insumo a geração de relatórios de acompanhamento com resultados fechados. A gestão das atividades de execução fica sob a responsabilidade de cada grupo de pesquisa envolvido no projeto. No entanto, internamente no LaCCAN, as atividades globais são subdivididas em atividades individuais e realizamos a gestão de cada sub-atividade como uma pesquisa individual já discutida anteriormente.

Por fim, os projetos de pesquisa como atividade meio exigiram um nível de sistematicidade que culminou na incorporação completa das metodologias ágeis e na proposição da Metodologia Ny. Esses projetos ditos de PD&I têm como premissa o desenvolvimento pesquisa que gerem protótipos, MVPs ou produtos a serem incorporados em empresas públicas ou privadas. Para tal, a gestão e execução é interna ao LaCCAN, conta com uma média de 10 a 15 membros e com um tempo de finalização que varia entre 1 e 2 anos.

Para esses projetos utilizamos a Metodologia Ny considerando *Sprints* lineares fixas (de 1 a 3 ciclos) para as etapas “Identificação do problema”, “O que existe?”, “Catálogo de soluções” e “Reprodução das melhores estratégias”. Já as etapas “Novas soluções”, “*Toy project*” e “Prototipação” são executadas encadeadas e considerando vários ciclos de um conjunto de Sprints. Por exemplo, executamos em 2 *Sprints* a “Novas soluções” seguida de 1 *Sprint* para “*Toy project*” e 1 *Sprint* para “Prototipação”, ao fim dessa última *Sprint* voltamos para o refinamento das “Novas soluções” e um novo encadeamento é executado. Assim, conseguimos ter uma primeira versão do protótipo em execução rapidamente e com incrementos durante o desenvolvimento do projeto.

Um exemplo satisfatório da aplicação da Metodologia Ny, no contexto do Programa LaCCAN de Residência em Ciência dos Dados em parceria com a Secretaria de Ciência, Tecnologia e Inovação do estado de Alagoas¹³, foi o desenvolvimento do sistema para o gerenciamento do Programa Criança Alagoana (CRIA)¹⁴ que contempla mais de 125.000 famílias carentes no estado de Alagoas, o sistema desenvolvido contém informações de acompanhamento de gestantes e crianças na primeira infância (desde a gestação até os primeiros 06 anos de vida). Possui uma assistência virtual e monitoramento remoto das usuárias do cartão CRIA. Os dados são inseridos de forma inicial a partir de um banco de dados `.xlsx` ou `.csv` da base do CAD Único, que em suma, a maioria dos dados estão contidos. Para além dos referidos dados, algumas informações novas são inseridas a partir dos Centro de Referência da Assistência Social (CRAS) em cada um dos municípios, além de gerenciadas e monitoradas pela central existente no escopo da Secretaria Estadual de Assistência e Desenvolvimento Social (SEADES).

O desenvolvimento desse sistema teve início em junho de 2021 e sua versão atual (maio de 2022) contempla todas as funcionalidades básicas exigidas. A estrutura interna do sistema faz parte de um grande projeto de pesquisa que envolve a construção de um *data lake* lógico [Gorelik 2019]. A tabela 2.1 apresenta os principais insumos referentes à aplicação da Metodologia Ny para a construção desse sistema. Note que com aproximadamente 3 meses do início das atividades o sistema já estava no ar e susceptível à melhoramentos e incrementos já em produção.

¹³<http://www.cienciaetecnologia.al.gov.br/>

¹⁴<https://cadastro.cria.al.gov.br/>

Tabela 2.1. Aplicação da Metodologia Ny para o sistema CRIA.

Indicador	Característica
<i>Product owner</i>	1 pesquisador doutor
<i>Scrum master</i>	1 pesquisador mestre
Desenvolvedores	9 residentes (graduandos e graduados)
Pesquisador consultor	–
Duração total	12 meses
Tempo das <i>Sprints</i>	2 semanas
Quantidade de <i>Sprints</i>	20
Primeira entrega	<i>Sprint</i> 6 com 30% das funcionalidades

Para o desenvolvimento dessa solução empregamos a Metodologia Ny com as seguintes etapas de pesquisa:

- **Identificação do problema:** O *product owner* (pesquisador doutor), ouviu as necessidades do cliente (gestores da Secretaria Estadual de Assistência Social (SE-ADS) e da Secretaria de Ciência e Tecnologia do Estado de Alagoas (SECTI)) e confeccionou um relatório com a descrição detalhada do problema.
- **O que existe em termos de soluções:** Para essa etapa foi tomado como base o sistema anterior em produção e que seria substituído. Foi feita uma engenharia reversa permitindo a identificação das principais soluções empregadas. Uma das funcionalidades do sistema era a sua integração com o *data lake* lógico que estava sendo desenvolvido em paralelo no escopo do Programa LaCCAN de Residência em Ciência dos Dados. O *Scrum master* (pesquisador mestre), ativou o time de desenvolvedores para realizar a consulta às soluções do setor produtivo similares ao que se ia desenvolver e às soluções do estado da arte (para o *data lake*) baseado nos princípios de revisão sistemática.
- **Catálogo de Soluções:** Não foi necessária essa etapa.
- **Reprodução das melhores estratégias:** Não foi necessária essa etapa.
- **Novas soluções:** Foram consideradas as boas práticas e melhores estratégias de desenvolvimento para o *front-* e *back-end* do sistema. Até sua primeira versão essa etapa foi executada em série, após uma versão já em produção todas as *Sprints* seguintes foram executadas em conjunto com as próximas duas etapas. Os desenvolvedores, uma vez familiarizados com o sistema utilizado anteriormente, junto da orientação do *Scrum master*, implementaram uma solução mais robusta para o problema.
- **Toy project, teste e validação:** A validação foi feita de forma sistemática para todos os incrementos entregues e disponibilizados para testes. Considerávamos sempre duas versões do sistema: uma de validação e outra de produção. Sempre ao fim de uma *Sprint*, uma versão de teste era disponibilizada para que os clientes realizassem

os testes necessários (1 dia para essa atividade) e após a homologação a nova versão era atualizada em produção.

- **Prototipação e consolidação:** Ao fim de cada ciclo, após a homologação uma nova versão era atualizada em produção. A partir da geração da primeira versão em produção essa etapa era executada ao fim de cada *Sprint* com todos os incrementos obtidos na *Sprint* corrente.

De forma geral, para o desenvolvimento do projeto seguimos todas as atividades previstas para uma *Sprint*: Definição da visão do projeto; planejamento da pesquisa; planejamento das *Sprints* iterativas; encontros diários durante todo o desenvolvimento do projeto; revisão das *Sprints* permitindo a interação do cliente e refinamento do que se estava sendo desenvolvido; e retrospectiva da *Sprint* permitindo a cada ciclo uma reflexão sobre o sucesso e as falhas da *Sprint* afim de empregar melhorias futuras.

Para além desse exemplo, no âmbito do LaCCAN já desenvolvemos com o auxílio de Metodologia Ny uma série de projetos com diferentes funcionalidades e objetivos. Desde sistemas com baixa complexidade como o caso do OxeTech¹⁵ que é um sistema para o cadastro e gerenciamento do ecossistema de inovação do estado de Alagoas; até um sistema de alta complexidade para identificação de erros de preenchimento de notas fiscais de operações de trânsito de mercadorias interestaduais. Esse sistema, chamado de CALT¹⁶, envolveu mais de 30 participantes e durou 4 anos.

Nesse contexto, com o intuito de levar para sociedade as soluções e a metodologia de pesquisa para resolução de problemas de forma mais célere, em 2021 foi criada a Ny Research¹⁷. Empresa de consultoria em PD&I, *Spinoff* do LaCCAN, que leva ao setor produtivo soluções de pesquisa inovadoras. Para tal, como diferencial da Ny Research é utilização da Metodologia Ny no desenvolvimento de seus projetos. Com isso, a Ny Research integra processos de pesquisa científica com métodos ágeis para a resolução de problemas complexos do setor produtivo.

Diferente dos projetos desenvolvidos por intermédio do LaCCAN a Ny Research tem como premissa a geração rápida de produtos mínimos viáveis em até 6 meses. Um dos projetos desenvolvidos, visava o emprego de técnicas do estado da arte para a remoção de fundo de imagens. A tabela 2.2 apresenta os principais insumos referentes à aplicação da Metodologia Ny para a construção desse sistema. Uma peculiaridade desse sistema é que desenvolvemos apenas o motor de remoção de fundo. A solução entregue conseguia resultados melhores do que o estado da arte. Toda a parte de integração e inclusão da solução na ferramenta foi realizada por uma equipe interna do cliente.

Para o desenvolvimento dessa solução empregamos a Metodologia Ny com as seguintes etapas de pesquisa:

- **Identificação do problema:** Nesta etapa, o *product owner* (pesquisador doutor), ouviu as necessidades do cliente afim de verificar a viabilidade de transformação destas em hipóteses possíveis de validação dentro do prazo de execução do projeto.

¹⁵<https://oxetech.al.gov.br/>

¹⁶<http://shorturl.at/duQUV/>

¹⁷<http://nyresearch.com.br/>

Tabela 2.2. Aplicação da Metodologia Ny para o sistema de remoção de fundo.

Indicador	Característica
<i>Product owner</i>	1 pesquisador doutor
<i>Scrum master</i>	1 pesquisador mestre
Desenvolvedores	1 profissional
Pesquisador consultor	–
Duração total	6 meses
Tempo das <i>Sprints</i>	2 semanas
Quantidade de <i>Sprints</i>	11 <i>Sprints</i>
Primeira entrega	Ao fim do projeto MVP completo

- **O que existe em termos de soluções:** Uma vez definido o problema, este foi adicionado à lista de *product backlog* e categorizado como item a ser considerado na etapa de catalogação de soluções da Metodologia Ny. O *Scrum master* (também pesquisador doutor), ativou a equipe de desenvolvimento para realizar a consulta ao estado da arte baseado nos princípios de revisão sistemática envolvendo artigos, patentes e soluções do setor produtivo.
- **Catálogo de Soluções:** Nesta etapa, o desenvolvedor, sob orientação do *Scrum master*, teve completa autonomia para selecionar três soluções do estado da arte que atendeu aos requisitos vindos da necessidade do cliente e explicitados pelo *product owner*.
- **Reprodução das melhores estratégias:** O desenvolvedor realizou a reprodução das soluções contidas no catálogo, a saber, implementação de modelos de *Deep Learnig* que automatizam a remoção de *background* em imagens. Impedimentos encontrados no processo, foram acompanhados e elucidados pelo *Scrum master*. Demandas técnicas incompreendidas eram esclarecidas rapidamente nas reuniões da *Sprint* que incluem o *product owner* e o cliente. Dessa forma, as *Sprints* entregaram incrementos previsíveis com os riscos de soluções inapropriadas minimizados.
- **Novas soluções:** Nesta etapa, o desenvolvedor, uma vez familiarizado com a hipótese do projeto e as soluções reproduzidas no catálogo, junto a orientação do *Scrum master*, implementou uma solução mais robusta para o problema, a partir da combinação dos métodos catalogados.
- **Toy project, teste e validação:** Nesta etapa o time validou o modelo produzido como nova solução do problema. Para a criação de um modelo com boa capacidade de generalização, foi imprescindível a participação do cliente, no fornecimento de dados que representassem bem, os desafios do ambiente de produção.
- **Prototipação e consolidação:** A API entregue com protótipo incorporou o modelo já validado dentro do ambiente de produção do cliente. Foram realizados ajustes finos para deixar a solução completamente funcional. Ademais, o relatório de sucesso e a documentação da API foi entregue.

Tal qual no estudo de caso anterior, para o desenvolvimento do sistema acima seguimos todas as atividades previstas para uma *Sprint*: Definição da visão do projeto; planejamento da pesquisa; planejamento das *Sprints* executadas de forma cíclica até o fim do projeto; encontros diários durante todo o desenvolvimento do projeto; revisão das *Sprints* com a interação direta do cliente e refinamento do que se estava sendo desenvolvido; e retrospectiva da *Sprint* permitindo a cada ciclo uma reflexão e melhoria das práticas de desenvolvimento no âmbito do projeto.

Identificamos que por intermédio da Metodologia Ny, foi possível nos aproximar mais do setor produtivo, uma vez que conseguimos dialogar de forma ágil e gerar resultados rápidos, atendendo assim as expectativas. Em relação aos pesquisadores, apesar de rápidas entregas e um baixo planejamento, o emprego da metodologia de pesquisa com todos os seus passos dá a segurança necessária para o sucesso da pesquisa e a consequente entrega do produto pretendido. Por fim, identificamos empiricamente que os resultados da pesquisa e a satisfação dos clientes são positivos frente ao uso de uma metodologia de pesquisa ágil. Tal evidência, nos impulsiona a continuar investigando e testando boas práticas ágeis afim de melhorar ainda mais a proposta da Metodologia Ny.

2.6. Considerações finais

Projetos de pesquisa com colaboração entre organizações de pesquisa e o setor produtivo, visivelmente requerem novas técnicas de gerenciamento de projetos que acompanhem a evolução das atividades de pesquisa e o quanto estas atividades estão atendendo às necessidades de inovação do setor produtivo que, por sua vez, já incorpora estratégias ágeis em seus processos de desenvolvimento, logo não alinha bem com a realidade do desenvolvimento de pesquisa tradicional. Neste sentido, apresentamos os principais desafios, conceitos, relatos e estudos de caso da adoção de estratégias ágeis, em particular o *framework* Scrum, em pesquisa colaborativa e multidisciplinar voltada para produção de inovação tecnológica.

Este trabalho teve como motivação o compartilhamento de experiências adquiridas pelos seus autores ao longo de 10 anos atuando na coordenação de projetos de pesquisa, tanto acadêmica como em conjunto com empresas públicas e privadas. Levantamos a discussão e algumas curiosidades em torno do uso de estratégias ágeis dentro de projetos de pesquisa, mais especificamente em projetos de PD&I desenvolvidos entre grupos de pesquisa e o setor produtivo.

De forma geral, apresentamos as principais etapas de desenvolvimento de pesquisa científica e eventuais entraves em torná-las ágeis. Por outro lado, identificamos que existe uma constante necessidade de colaboração entre institutos de pesquisa e o setor produtivo, porém esta não consegue se concretizar satisfatoriamente. Uma razão para que isso ocorra é pela existência de uma lacuna entre a produção científica e as necessidades das empresas, seja pelo uso de diferentes metodologias de desenvolvimento ou pela dificuldade em integrar as soluções criadas

Apresentamos também, um visão geral do conceito de estratégias ágeis e detalhamos os principais métodos ágeis considerados no setor produtivo: Kanban, XP e Scrum. Em linhas gerais, sobre a viabilidade de integração entre esses métodos e a pesquisa científica identificamos que: O Kanban apesar de ser passível de se aplicar no contexto um

problema de pesquisa, a sua proposição de se ter o fluxo de trabalho real sendo modelado no quadro Kanban durante a execução do projeto fere um pouco a metodologia e princípios de uma pesquisa inovadora. O XP também é passível de se aplicar no contexto de uma metodologia científica, especificamente se for uma atividade que envolva um time pequeno. Porém, sua proposta considerando um baixo planejamento prévio e a direta implementação das soluções, fere um pouco a metodologia e os princípios de uma pesquisa inovadora. Com isso, a aplicabilidade do XP para projetos de PD&I poderia ser incorporada apenas no momento e para a demanda de implementação, no caso uma pequena sub-atividade dentro do processo de pesquisa. O Scrum, por sua vez, é uma metodologia de fácil adaptação às metodologias científica, logo torna-se um opção atrativa para projetos de PD&I. Sua proposta em desenvolvimento baseado em *Sprints*, considerando sua revisão e planejamento em ciclos curtos, favorecem a produtividade do time e permite que as abordagens para o desenvolvimento da pesquisa sejam verificadas e alteradas em tempo hábil sem maiores prejuízos.

Em seguida, discutimos sobre a aplicação das estratégias ágeis para o desenvolvimento de projetos de base científica. Nesse sentido, abordamos as principais questões que motivam o desenvolvimento de projetos de PD&I em conjunto com as instituições de pesquisa. De forma complementar apresentamos como as estratégias ágeis podem ser úteis para diminuir as distâncias entre o setor produtivo e as instituições de pesquisa. Por fim, como exemplo, apresentamos a Metodologia Ny que é resultado de uma longa experiência dos autores deste trabalho atuando na coordenação de projetos de pesquisa, tanto acadêmica como em conjunto com empresas.

Finalizamos esse trabalho apresentando dois relatos sobre o uso de métodos ágeis em conjunto com instituições de pesquisa, em particular os casos sobre o CECAN, LaC-CAN e Ny Research. De forma geral, identificamos que por intermédio do uso de metodologias ágeis é possível reduzir as distâncias entre o setor produtivo e as instituições de pesquisa.

Agradecimentos

O presente trabalho foi realizado com apoio da Fundação de Amparo à Pesquisa do Estado de Alagoas (FAPEAL) e da empresa Ny Research.

Referências

- [Ankrah and AL-Tabbaa 2015] Ankrah, S. and AL-Tabbaa, O. (2015). Universities–industry collaboration: A systematic review. *Scandinavian Journal of Management*, 31(3):387–408.
- [Atta-Owusu et al. 2021] Atta-Owusu, K., Fitjar, R. D., and Rodríguez-Pose, A. (2021). What drives university–industry collaboration? research excellence or firm collaboration strategy? *Technological Forecasting and Social Change*, 173:121084.
- [Bammer 2008] Bammer, G. (2008). Enhancing research collaborations: Three key management challenges. *Research Policy*, 37(5):875–887.

- [Beck and Andres 2004] Beck, K. and Andres, C. (2004). *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional, Boston, 2 edition.
- [Beck et al. 2001] Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001). Manifesto for agile software development.
- [Cardoso et al. 2018] Cardoso, M. G., Amboni, N., Lagemann, G. V., and de Andrade, R. O. B. (2018). Fatores facilitadores e restritivos à cooperação universidade e empresa: o caso udesc. *Desenvolvimento em Questão*, 16(45):273–291.
- [Ciric et al. 2018] Ciric, D., Lalic, B., Gracanin, D., Palcic, I., and Zivlak, N. (2018). Agile project management in new product development and innovation processes: Challenges and benefits beyond software domain. In *2018 IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)*.
- [Cohen et al. 2004] Cohen, D., Lindvall, M., and Costa, P. (2004). An introduction to agile methods. In *Advances in Computers*, volume 62, pages 1–66. Elsevier.
- [Cooke and Hilton 2015] Cooke, N. J. and Hilton, M. L., editors (2015). *Enhancing the Effectiveness of Team Science*. The National Academies Press, Washington, DC.
- [Cruz 2015] Cruz, F. (2015). *Scrum e Agile em Projetos - Guia Completo: Conquiste sua certificação e aprenda a usar métodos ágeis no seu dia a dia*. BRASPORT.
- [di Fiori et al. 2019] di Fiori, A., West, K., and Segnalini, A. (2019). Why science-driven companies should use agile. *Harvard Business Review*, page 34.
- [Etzkowitz and Zhou 2017a] Etzkowitz, H. and Zhou, C. (2017a). Hélice tríplice: inovação e empreendedorismo universidade-indústria-governo. *Estudos avançados*, 31(90):23–48.
- [Etzkowitz and Zhou 2017b] Etzkowitz, H. and Zhou, C. (2017b). *The triple helix: University–industry–government innovation and entrepreneurship*. Routledge.
- [Gorelik 2019] Gorelik, A. (2019). *The Enterprise Big Data Lake: Delivering the Promise of Big Data and Data Science*. O’Reilly Media, 1 edition.
- [Grimpe and Kaiser 2010] Grimpe, C. and Kaiser, U. (2010). Balancing internal and external knowledge acquisition: The gains and pains from r&d outsourcing. *Journal of Management Studies*, 47(8):1483–1509.
- [Hayter et al. 2020] Hayter, C. S., Rasmussen, E., and Rooksby, J. H. (2020). Beyond formal university technology transfer: innovative pathways for knowledge exchange. *The Journal of Technology Transfer*, 45(1):1–8.
- [Hidalgo 2019] Hidalgo, E. S. (2019). Adapting the Scrum framework for agile project management in science: Case study of a distributed research initiative. *Heliyon*, 5(3):e01447.

- [Highsmith and Highsmith 2002] Highsmith, J. A. and Highsmith, J. (2002). *Agile Software Development Ecosystems*. Addison-Wesley Longman Publishing Co., Inc., USA.
- [Kanban University 2021] Kanban University (2021). *The official guide to the Kanban method*. Mauvius Group Inc.
- [Lee et al. 2014] Lee, D. D., Kirkpatrick-Husk, K., and Madhavan, R. (2014). Diversity in alliance portfolios and performance outcomes: A meta-analysis. *Journal of Management*, 43(5):1472–1497.
- [Mendes (org.) 2022] Mendes (org.), M. (2022). *Para não esquecer: políticas públicas que empobrecem o Brasil*. Autografia, 1 edition.
- [OECD 2015] OECD (2015). *Frascati Manual 2015*. OECD publishing.
- [OECD and Eurostat 2018] OECD and Eurostat (2018). *Oslo Manual 2018*. OECD publishing.
- [Pogrebnoy and Yatskevich 2022] Pogrebnoy, K. and Yatskevich, O. (2022). Agile or waterfall: Choose the right approach to your software project management. Accessed: May 2022.
- [Raharjo and Purwandari 2020] Raharjo, T. and Purwandari, B. (2020). Agile project management challenges and mapping solutions: A systematic literature review. In *Proceedings of the 3rd International Conference on Software Engineering and Information Management, ICSIM '20*, pages 123–129, New York, NY, USA. Association for Computing Machinery.
- [Schwaber and Sutherland 2020] Schwaber, K. and Sutherland, J. (2020). *The Scrum Guide – The Definitive Guide to Scrum: The Rules of the Game*. Share-Alike license of Creative Commons.
- [Sutherland 2014] Sutherland, J. (2014). *Scrum: The Art of Doing Twice the Work in Half the Time*. Random House Publishing Group.
- [Takeuchi and Nonaka 1986] Takeuchi, H. and Nonaka, I. (1986). The new new product development game. *Harvard Business Review*.
- [Trentim 2017] Trentim, M. H. (2017). *Gerenciamento de Projetos: Guia para Certificações CAPM e PMP*. Atlas, 2 edition.
- [Tseng et al. 2020] Tseng, F.-C., Huang, M.-H., and Chen, D.-Z. (2020). Factors of university–industry collaboration affecting university innovation performance. *The Journal of Technology Transfer*, 45(2):560–577.
- [Wagner et al. 2017] Wagner, C. S., Whetsell, T. A., and Leydesdorff, L. (2017). Growth of international cooperation in science: Revisiting six case studies. *Scientometrics*, 110:1633–1652.

- [Weber and Heidenreich 2018] Weber, B. and Heidenreich, S. (2018). When and with whom to cooperate? investigating effects of cooperation stage and type on innovation capabilities and success. *Long Range Planning*, 51(2):334–350.
- [Ynalvez and Shrum 2011] Ynalvez, M. A. and Shrum, W. M. (2011). Professional networks, scientific collaboration, and publication productivity in resource-constrained research institutions in a developing country. *Research Policy*, 40(2):204–216.
- [Yordanova 2021] Yordanova, Z. (2021). *Agile Application for Innovation Projects in Science Organizations - Knowledge Gap and State of Art*, volume 1330, pages 108–117. Springer, Cham.
- [Yordanova et al. 2019] Yordanova, Z., Stoimenov, N., Boyanova, O., and Ivanchev, I. (2019). The long way from science to innovation – a research approach for creating an innovation project methodology. In *Business Information Systems*.

Capítulo

3

Big Data e Tecnologias Digitais Aplicadas à Economia Circular: Oportunidades para Cadeias Produtivas mais Sustentáveis

Rosângela de Fátima Pereira Marquesone, Francisco Pereira Junior, Tereza Cristina Melo de Brito Carvalho

Abstract

Several environmental, economic and social challenges are currently being identified, requiring sustainable approaches that enable the preservation of the ecosystem and promote current and future well-being of society. Given this scenario, the circular economy proposal has been considered a potential resource to achieve such goals, generating changes in the production and consumption of products, from business models based on principles of ecodesign, reduction, recovery, reuse and recycling of products. However, it is identified that such business models must be supported by data-driven solutions and digital technologies, to be scalable, facilitate collaboration, and promote greater awareness. Therefore, this course has as main objective to show how big data and digital technologies, such as the internet of things, cloud computing and blockchain, play a key role in the transition to the circular economy, pointing out characteristics of these solutions and how they meet the needs of business models in the context of the circular economy, thus being able to contribute to the current sustainability challenges.

Resumo

Diversos desafios de cunho ambiental, econômico e social têm sido identificados atualmente, necessitando de abordagens sustentáveis que possibilitem a preservação do ecossistema e promovam o bem estar atual e futuro da sociedade. Diante desse cenário, a proposta de economia circular tem sido considerada um recurso potencial para se alcançar tais objetivos, gerando mudanças na produção e no consumo de produtos, a partir de modelos de negócios baseados em princípios de ecodesign, redução, recuperação, reuso e reciclagem de produtos. Entretanto, identifica-se que tais modelos de negócios devem ser apoiados por soluções orientadas a dados e tecnologias digitais para serem escaláveis,

facilitarem a colaboração e promoverem maior conscientização. Dessa forma, esse curso tem como principal objetivo mostrar como big data e as tecnologias digitais, tais como internet das coisas, computação em nuvem e blockchain, desempenham um papel chave na transição para a economia circular; identificando-se características dessas soluções e como essas atendem às necessidades de modelos de negócio no contexto de economia circular, podendo assim contribuir para os desafios atuais de sustentabilidade.

3.1. Introdução

Nos últimos anos têm sido cada vez mais perceptível os efeitos decorrentes da mudança climática. Um dado que mostra o quanto essa situação é alarmante é apresentado pela Convenção-Quadro das Nações Unidas sobre a Mudança do Clima (*United Nations Framework Convention on Climate Change - UNFCCC*), relatando que, em 2018, o uso dos recursos naturais ocorreu 1,7 vezes mais rápido do que a capacidade do ecossistema se regenerar [UNFCCC 2018]. Nessa circunstância, existe uma preocupação global de que, caso o padrão persista, em 2030 será necessário o equivalente a aproximadamente três planetas para prover os recursos naturais, considerando as estimativas de crescimento populacional [ONU 2021a]. Em adição, o sexto relatório de avaliação (AR6) do Painel Intergovernamental sobre Mudanças Climáticas (*Intergovernmental Panel on Climate Change Sixth Assessment Report - IPCC*) apresentou informações alarmantes sobre os efeitos da mudança climática no planeta, afirmando que o aumento dos gases de efeito estufa atmosféricos são inequivocamente causados por atividades humanas [IPCC 2021]. Diante de tais cenários, as iniciativas de produção e consumo sustentáveis têm sido consideradas urgentes e desafiadoras, para preservar os recursos naturais e reduzir as consequências para a humanidade.

Uma alternativa que tem sido considerada promissora para tornar a produção e o consumo mais sustentáveis é a economia circular [EMF 2015, Kirchherr et al. 2017, Gupta et al. 2019]. Diferente da economia linear, que se caracteriza pela contínua extração de matérias-primas e a geração de resíduos quando o produto gerado chega ao seu fim de vida, a economia circular tem como objetivo um sistema de produção e consumo que considere não somente o crescimento econômico, mas também a preservação ambiental e o bem-estar social [Kirchherr et al. 2017]. Dessa forma, empresas que buscam ser mais sustentáveis estão adotando como estratégia a transição para a economia circular.

Nesse contexto, identifica-se, também, que nos últimos anos têm ocorrido um crescente avanço em termos de soluções orientadas a dados e da adoção de tecnologias digitais, ocasionando mudanças significativas nas estratégias de negócios, na maneira com que nos socializamos e na cultura organizacional [Lee 2017, Martínez-Caro et al. 2020]. A partir de tecnologias de big data, bem como outras tecnologias digitais como blockchain e computação em nuvem, empresas e setores industriais identificam a necessidade de se adaptarem a esse novo cenário, utilizando os dados e as inovações tecnológicas como instrumentos chave para o aperfeiçoamento de suas estratégias. Entretanto, embora big data e tecnologias digitais já sejam adotadas por empresas para atender determinados objetivos como estratégias de *marketing*, otimização de processos e melhoria da experiência do cliente, ainda não é claro na literatura como essas podem contribuir na transição para a economia circular [Gupta et al. 2019, Khan et al. 2022]. Dada essas circunstâncias, nesse trabalho é apresentado um estudo sobre como big data e tecnologias digitais

podem ser utilizadas como elementos chave para a obtenção de cadeias produtivas mais sustentáveis, a partir de estratégias de economia circular. Espera-se, a partir desse estudo, realizar uma aproximação entre os temas big data, tecnologias digitais e economia circular, identificando como profissionais dessas áreas podem contribuir colaborativamente na busca por soluções sustentáveis.

Para atender a esse objetivo, na Seção 3.2 é apresentada uma contextualização sobre sustentabilidade e economia circular, descrevendo-se seus conceitos e suas características. Na sequência, a Seção 3.3 contém uma investigação sobre a definição e as características de big data, além de uma descrição das tecnologias de big data existentes e as técnicas de big data analytics. Em complemento, na Seção 3.4, são relatados os fundamentos referentes a tecnologias digitais, com ênfase nas características de algumas dessas tecnologias. Após a contextualização desses termos, na Seção 3.5 é apresentada uma investigação sobre a aplicação de big data e tecnologias digitais na transição para a economia circular, identificando-se estratégias que podem contribuir para o sucesso nessa transição. A partir desse contexto, na Seção 3.6 é apresentado como essa adoção se aplica a uma indústria específica, a indústria têxtil, uma vez que essa apresenta diversos desafios de sustentabilidade ao longo de sua cadeia produtiva. Por fim, a Seção 3.7 contém as considerações finais. A Figura 3.1 apresenta uma ilustração do relacionamento entre os conteúdos abordados.

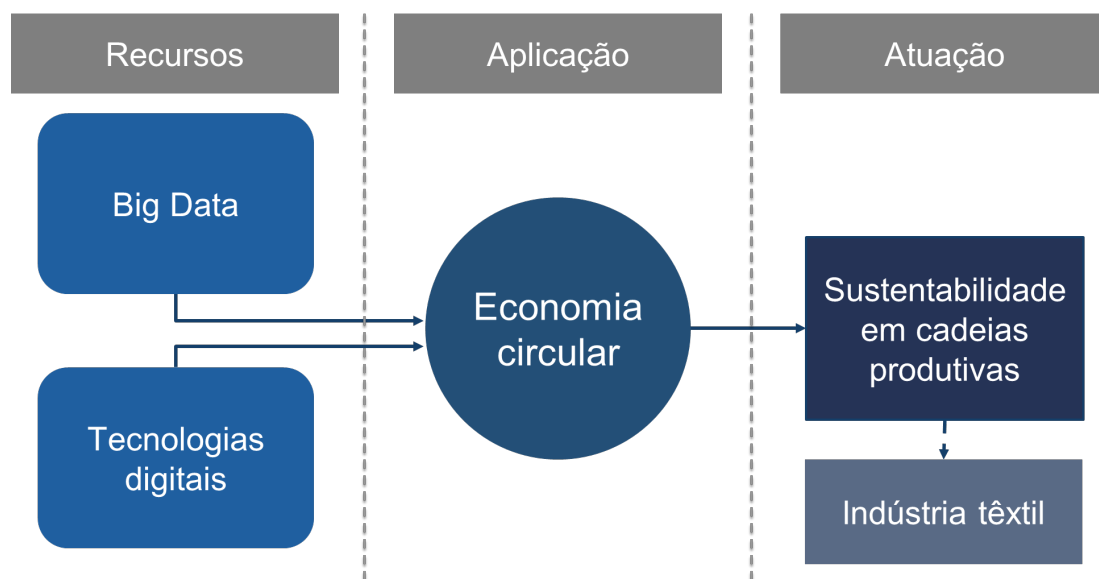


Figura 3.1. Estrutura do conteúdo

3.2. Sustentabilidade e Economia Circular

Essa seção aborda o conceito de sustentabilidade, descrevendo, também, os pilares que o regem. Em complemento, é relatado como as Organizações das Nações Unidas (ONU) propõem uma agenda global para objetivos de desenvolvimento sustentável. Por fim, essa seção contém informações referentes à economia circular, uma vez que ela tem sido considerada um recurso chave para se alcançar a sustentabilidade nos modos de produção e consumo.

3.2.1. Definição e Pilares de Sustentabilidade

Dentre os marcos históricos significativos referentes ao tema sustentabilidade, pode-se citar a introdução do termo “desenvolvimento sustentável”, apresentado em 1972, na Conferência das Nações Unidas sobre o Homem e o Meio Ambiente. Esse conceito foi descrito posteriormente no relatório da Comissão Brundtland, gerando o livro intitulado “Nosso Futuro Comum” [Brundtland 1987]. Nesse relatório, desenvolvimento sustentável foi descrito como aquele que “atende às necessidades do presente sem comprometer a capacidade das gerações futuras de atenderem suas próprias necessidades”. Essa definição enfatizou que sustentabilidade refere-se a um desafio global, tornando-se necessário avaliar e reconhecer os limites do crescimento e avaliando-se seu impacto ambiental e social, de forma a considerar não somente os efeitos atuais, como também para as próximas gerações [Rogers and Hudson 2011].

Desde a publicação do relatório de Brundtland, novas discussões e propostas de definição sobre o tema sustentabilidade foram sendo apresentadas na literatura. Nesse contexto, surgiu, também, a proposta de um *framework* contendo os pilares de sustentabilidade: econômico, ambiental e social, conhecido como Tripé da Sustentabilidade ou *Triple Bottom Line* (TBL) [Elkington 1997]. Nesse *framework*, o desenvolvimento sustentável é visto como aquele que avança de forma simultânea na prosperidade econômica, na preservação do meio ambiente e na equidade social.

Nos últimos anos, além do TBL, novos *frameworks* e estratégias foram sendo propostos, com foco em sustentabilidade. O Global Reporting Initiative (GRI), por exemplo, contribui, desde 1997, com a proposta de diretrizes e indicadores para que empresas construam e divulguem, de forma voluntária, relatórios de sustentabilidade, apresentando um relato dos impactos ambientais e sociais causados em suas práticas [GRI 2021]. Atualmente, essas diretrizes têm sido utilizadas como modelo pelas empresas para a geração de relatórios voluntários de Responsabilidade Social Corporativa (*Corporate Social Responsibility* - CSR) [Fuente et al. 2017].

Recentemente, uma outra iniciativa proposta em relação à sustentabilidade é o conjunto de práticas e padrões *Environment, Social and corporate Governance* (ESG), referente a questões ambientais, sociais e de governança corporativa. Alguns pesquisadores argumentam que as métricas e as práticas presentes no ESG são mais relevantes que o TBL, pois justificam que TBL tem como principal foco o desempenho financeiro [Ferrell 2021].

3.2.2. Objetivos de Desenvolvimento Sustentável (ODS)

Além dos pilares da sustentabilidade apresentados, diversos movimentos têm ocorrido em esfera global, com o objetivo de discutir e fornecer soluções estratégicas para reverter situações alarmantes. Um movimento significativo nesse contexto são os Objetivos de Desenvolvimento Sustentável (ODS), propostos pela ONU em setembro de 2015, como uma agenda global para a promoção da prosperidade humana, respeitando o planeta [ONU 2021b]. O plano de ação, intitulado "Agenda 2030 para o Desenvolvimento Sustentável", é composto por 17 ODS, delineando 169 metas específicas, abrangendo aspectos da sustentabilidade em níveis distintos, tais como erradicar a pobreza, reduzir desigualdades e melhorar a educação, conforme ilustrado na Figura 3.2.



Figura 3.2. Objetivos de Desenvolvimento Sustentável (ODS) [ONU 2021b]

Em [Kostoska and Kocarev 2019], os autores propõem uma organização dos ODS em relação aos pilares de sustentabilidade, identificando como esses estão interligados. Um exemplo dessa organização pode ser visualizado na Figura 3.3. Identifica-se, assim, que o tema sustentabilidade é discutido em diferentes perspectivas. Por exemplo, um ODS específico é o “ODS 12 - Consumo e produção responsáveis”, focado, em resumo, na promoção do crescimento econômico sem degradação ambiental, aumentando a eficiência dos recursos e proporcionando estilos de vida sustentáveis [ONU 2021a]. Dessa forma, o objetivo de se obter cadeias produtivas mais sustentáveis está diretamente relacionado a esse ODS. Entretanto, os resultados de soluções focadas nesse objetivo também podem se relacionar a outros, direta ou indiretamente, como por exemplo, ao ODS 13, referente à ação contra a mudança global do clima.



Figura 3.3. Objetivos de Desenvolvimento Sustentável (ODS). Adaptado de [Kostoska and Kocarev 2019]

3.2.3. O Conceito de Economia Circular

Embora a busca pelo desenvolvimento sustentável esteja atualmente em evidência, ainda é considerado um desafio para empresas e indústrias identificarem como reconfigurar suas estratégias e proposta de valor para alcançar esse objetivo. Com a proposta de ser uma alternativa ao modelo linear, atualmente predominante no processo industrial, a economia circular tem sido identificada como uma solução promissora para o desenvolvimento sustentável [Geissdoerfer et al. 2018].

Economia circular é considerado um conceito “guarda-chuva”, por incorporar em sua base diferentes escolas de pensamento, tais como ecologia industrial [Graedel 1996], *cradle-to-cradle* [McDonough and Braungart 2010] e biomimética [Benyus 1997]. Embora não seja um conceito recente, foi somente na última década que o termo obteve maior visibilidade, sendo creditada sua popularização à Fundação Ellen MacArthur, que ampliou as discussões sobre o tema. Em um relatório de 2012, desenvolvido pela fundação, economia circular foi definido como “um sistema industrial restaurador ou regenerativo por intenção ou *design*. Substitui o conceito de fim de vida por restauração, faz uso de energia renovável, elimina o uso de produtos químicos tóxicos que prejudicam a reutilização e visa a eliminação de resíduos por meio do *design* superior de materiais, produtos, sistemas, e dentro disso, modelos de negócios” [EMF 2012]. Observa-se, nessa definição, que a proposta de economia circular é abrangente, visando um novo paradigma de produção e consumo, a partir de um conjunto de iniciativas.

Embora a definição da Fundação Ellen MacArthur tenha sido a que trouxe maior popularidade ao termo, outras definições também são encontradas na literatura, em relação à compreensão de economia circular. A partir de uma investigação na literatura realizada por Kirchherr, Reike e Hekkert, em 2017, foram identificadas 114 diferentes definições sobre o termo, sendo a maioria derivadas da definição feita pela fundação [Kirchherr et al. 2017]. A partir desse estudo, os autores identificaram que economia circular faz referência a um sistema econômico baseado em modelos de negócios que substituem o conceito de fim de vida por redução e alternativamente por reutilização, reciclagem e recuperação de materiais em processos de produção/distribuição e consumo. Também acrescentam nessa definição o fato da economia circular ser aplicada em diferentes níveis operacionais, mencionando que ela opera nos níveis micro (e.g., empresas), intermediário (e.g., parques industriais) e no nível macro (e.g., cidades), com o objetivo de promover o desenvolvimento sustentável, criando ao mesmo tempo qualidade ambiental, prosperidade econômica e equidade social, para benefício das gerações atual e futuras [Kirchherr et al. 2017]. A Figura 3.4 contém um resumo dos principais fatores apresentados nas definições apresentadas.

3.2.4. Características da Economia Circular

É possível identificar, a partir dos conceitos apresentados, que a economia circular não tem como proposta lidar somente com o final de vida de produtos e materiais, por meio de estratégias como reciclagem, por exemplo. Sua proposta vai além, visando uma mudança de paradigma que abrange todo o ciclo de vida do produto, iniciando pelo *design*, uma vez que é a partir dessa etapa que as decisões tomadas poderão impactar as etapas seguintes, incluindo o uso e o pós-uso. Nesse contexto, as seguintes características podem

ser observadas, em relação à economia circular:

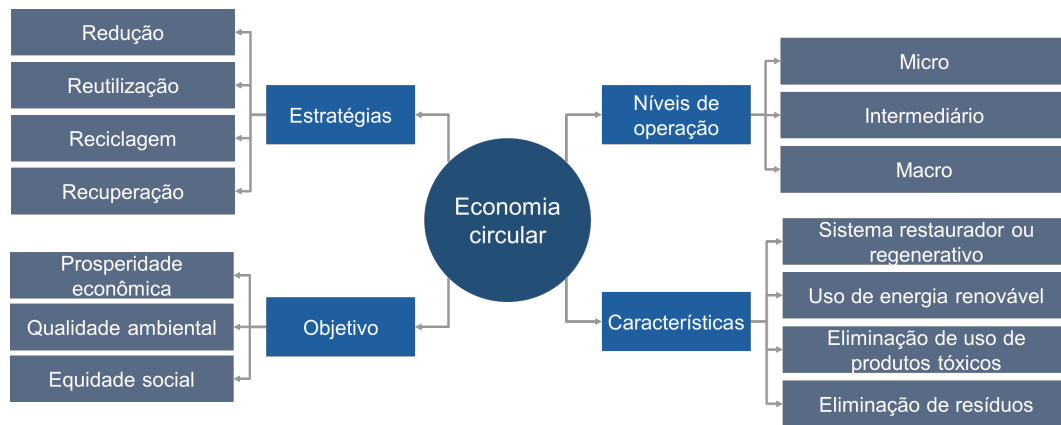


Figura 3.4. Fatores referentes à definição de economia circular

- **Visão sistêmica e holística.** A proposta de economia circular identifica a necessidade de que as decisões referentes ao processo da cadeia produtiva, incluindo todo o ciclo de vida de um produto, sejam tomadas a partir de uma visão sistêmica e holística. Ou seja, em cada etapa, decisões como a escolha dos materiais e do processo envolvido devem ser feitas a partir de uma abordagem que avalie o contexto em sua totalidade, compreendendo a importância e a interligação entre cada parte do sistema. Essa abordagem permite compreender, por exemplo, como uma decisão realizada na etapa de produção, pode trazer impactos ambientais negativos quando o produto estiver em seu fim de vida [EMF 2015]. Nesse contexto, a visão holística também compreende a necessidade de se avaliar o sistema a partir das três dimensões, econômica, ambiental e social.
- **Prolongamento do uso de um produto.** A economia circular busca estender o valor em cada estágio do ciclo de vida do produto, incluindo a fase final, aumentando assim a durabilidade desse, bem como possibilitando maior intensividade de seu uso [Liu et al. 2021]. Como consequência, essa extensão também visa a redução de material e de energia, por maximizar a utilização do produto.
- **Redução do impacto ambiental.** Na economia circular existe a preocupação com o impacto ambiental, por isso o sistema é projetado com foco em minimizar a poluição, o consumo de energia e de água, e não gerar resíduos [Modgil et al. 2021]. Busca-se a maximização do uso de energias renováveis, e o *redesign* de produtos, com foco na redução de materiais e no uso de materiais biodegradáveis.
- **Sistema de circuito fechado.** Ao invés de considerar os recursos incessantes, a economia circular busca atuar em sistemas circulares ou em “circuito fechado” em seu processo produtivo, permitindo a redução do uso de recursos naturais e evitando a geração de resíduos [Brydges 2021]. Nessa abordagem holística, há uma tentativa de integração durante as atividades produtivas, onde o produto resultante de um nível é utilizado como insumo para o nível seguinte, mimetizando os ciclos da natureza [Gupta et al. 2019].

- Harmonia entre os pilares de sustentabilidade. Identifica-se, também, o fato de a economia circular incorporar os três pilares de desenvolvimento sustentável (econômico, ambiental e social). Há estudos que relatam que algumas estratégias tendem a focar somente na dimensão ambiental, não considerando os impactos sociais dessa aplicação [Geissdoerfer et al. 2017, Kirchherr et al. 2017]. Nesse caso, não se cumpriria com os objetivos da economia circular, caso não haja uma harmonia entre as dimensões.

3.2.5. Estratégias de Circularidade

Conforme apresentado na definição de Kirchherr, Reike e Hekkert, a economia circular abrange estratégias voltadas à redução, reutilização, reciclagem e recuperação de materiais e produtos [Kirchherr et al. 2017]. Entretanto, na literatura identifica-se, também, a proposta dos 10 Rs, visando novas estratégias de circularidade e modelos de negócio que permitam atender aos objetivos da economia circular.

É possível visualizar na Tabela 3.1 que essas estratégias são divididas em três grupos. O primeiro refere-se ao uso e à fabricação inteligente do produto, por meio de estratégias para recusar, repensar e reduzir o uso de materiais em sua produção. O segundo grupo refere-se ao prolongamento da vida útil do produto e de suas peças, por meio de estratégias como a reutilização, reparação, reforma, remanufatura e reaproveitamento dos produtos. Por fim, o terceiro grupo refere-se a aplicações úteis de materiais, envolvendo estratégias de reciclagem e de recuperação do material, por meio da geração de energia, por exemplo.

Tabela 3.1. Estratégias da economia circular.

Grupo	Categoria	Significado
Uso e fabricação inteligentes do produto	R0 - Recusar	Reutilização de um produto, dando a ele uma nova função, ou provendo a mesma função, entretanto com um produto radicalmente diferente.
	R1 - Repensar	Intensificação do uso do produto (e.g., compartilhando produtos ou colocando produtos multifuncionais no mercado).
	R2 - Reduzir	Aumento da eficiência na fabricação ou no uso do produto, consumindo assim menos recursos naturais.
Prolongar a vida útil do produto e suas peças	R3 - Reutilizar	Reutilização do produto descartado ainda em boas condições e que cumpre sua função original.
	R4 - Reparar	Reparação e manutenção de produto com defeito para ser usado com sua função original.
	R5 - Reformar	Restauração e atualização de um produto antigo.
	R6 - Remanufaturar	Uso de partes do produto descartado em um novo produto com a mesma função.
	R7 - Reaproveitar	Uso de produtos descartados ou suas partes em um novo produto com uma função diferente.
Aplicações úteis de materiais	R8 - Reciclar	Processamento de materiais para obter a mesma qualidade (maior ou menor grau).
	R9 - Recuperar	Incineração de material com recuperação de energia.

Fonte: Adaptado de [Potting et al. 2017].

As estratégias são apresentadas e sugeridas para que sejam aplicadas nessa ordem, sendo consideradas as primeiras mais efetivas e as últimas as menos efetivas, uma vez

que o processo de reciclagem e de recuperação, por exemplo, exigem atividades mais complexas, que podem gerar mais consumo de recursos naturais [Potting et al. 2017]. Observa-se, também, que tais estratégias podem ser incorporadas tanto por empresas, quanto pelos consumidores, contribuindo para que a produção e o consumo ocorram de forma consciente e colaborativa. Há, portanto, o compartilhamento de responsabilidades para que as estratégias sejam implementadas de forma efetiva. Ressalta-se também que, para a implantação efetiva da economia circular, há a necessidade de colaboração de todos os *stakeholders*, tais como produtores, fabricantes, fornecedores, varejistas, consumidores e governantes, uma vez que é necessária uma mudança do *status quo* dos hábitos de produção e de consumo. Somente com a soma das partes é que a sustentabilidade pode ser alcançada. Por exemplo, uma empresa que adote uma estratégia de circularidade necessita da colaboração de seus consumidores, para que esses tenham hábitos de consumo também sustentáveis. Além disso, governantes e formuladores de política desempenham um papel essencial nessa transição, criando políticas e regulamentos que estimulem e favoreçam a adoção de economia circular.

Além dos 10 Rs apresentados, há na literatura a proposta do *framework* ReSOLVE, desenvolvido pela Fundação Ellen MacArthur, com foco em também contribuir com estratégias no contexto de economia circular [EMF 2015]. Conforme apresentado na Tabela 3.2, nesse *framework*, as estratégias são divididas nas categorias regenerar, compartilhar, otimizar, criar ciclos, virtualizar e trocar. Percebe-se, nessa proposta, uma consonância com os 10 Rs anteriores, identificando que a reciclagem, novamente, não é considerada a única alternativa para prover a circularidade e atender aos objetivos de sustentabilidade.

Tabela 3.2. Framework ReSOLVE.

Estratégia	Descrição
Regenerar (<i>Regenerate</i>)	Refere-se a estratégias que visam a adoção de energia e materiais renováveis. Também incorpora meios que possibilitam a retenção e a regeneração da saúde da biosfera. Tem-se, como exemplo, o processo de compostagem, realizado a partir de ciclos biológicos.
Compartilhar (<i>Share</i>)	Modelos de negócios voltados à economia compartilhada e ao prolongamento do uso dos produtos. Tem-se, como exemplo, o serviço de compartilhamento de carros, bibliotecas e serviços de aluguel de roupas.
Otimizar (<i>Optimize</i>)	Adoção de tecnologias que promovam maior desempenho das organizações nas estratégias circulares, possibilitando, por exemplo, aumentar a eficiência de produtos e reduzir a geração de resíduos.
Criar ciclos (<i>Loop</i>)	Estratégias para eliminar a geração de resíduos e convertê-los em recursos, por meio de ciclos técnicos e biológicos. Exemplos incluem a reciclagem e a remanufatura de produtos.
Virtualizar (<i>Virtualize</i>)	Estratégias voltadas à troca de produtos físicos por produtos virtuais e pela prestação de serviços, evitando assim a extração de recursos e a geração de resíduos. Como exemplo, tem-se o serviço de <i>streaming</i> de músicas, evitando a produção de mídias físicas.
Trocar (<i>Exchange</i>)	Estratégias que possibilitam a troca de itens antigos e não renováveis por novos itens, baseados em tecnologias avançadas e materiais renováveis. Exemplos incluem a fabricação de peças a partir da impressão 3D.

Fonte: Adaptado de [EMF 2015]

3.3. Big Data

Os avanços tecnológicos ocorridos nas últimas décadas trouxeram mudanças significativas em toda a sociedade. Fatores como a ascensão das redes sociais, a evolução das tecnologias móveis e o crescente número de serviços disponíveis na Internet possibilitaram novos meios de comunicação, novas formas de trabalho e novos modelos de negócios. Conseqüentemente, tais avanços tecnológicos resultaram em um crescente volume de dados gerados por pessoas e por máquinas, a partir de fontes e formatos variados [Gandomi and Haider 2015]. Nesse contexto, empresas passaram a adotar os dados em sua estratégia, identificando meios para lidar com o conceito de big data, aprimorando suas percepções e aperfeiçoando o processo de tomada de decisão [Lee 2017]. Mais recentemente, as tecnologias de big data também estão sendo adotadas para a geração de soluções orientadas a dados para o contexto de sustentabilidade. Dessa forma, essa seção apresenta detalhes sobre a definição, as características, as tecnologias e as técnicas existentes em big data.

3.3.1. Definição de Big Data

A base para a definição do termo big data é creditada à Doug Laney, em 2001, a partir da publicação de um *white paper*, onde o autor apresenta os desafios de gerenciamento de dados, considerando as dimensões de volume, variedade e velocidade de dados [Laney et al. 2001]. Desde então, esses desafios ficaram conhecidos como os “Vs” de big data, sendo adicionadas outras dimensões com o passar do tempo, tais como o valor e a veracidade dos dados.

Seguindo essa abordagem, uma das definições adotadas para big data é aquela dada pela empresa de consultoria Gartner, na qual descreve big data como sendo “ativos de informações de grande volume, variedade e velocidade que demandam formas inovadoras e rentáveis de processamento da informação, para melhor percepção e tomada de decisão” [Gandomi and Haider 2015, Gartner 2022]. Ou seja, nessa definição é possível identificar que a adoção de big data oferece a empresas públicas e privadas tanto oportunidades e desafios a partir de uma abordagem orientada a dados, que, embora necessite de mudanças, como a adoção de uma infraestrutura que permita lidar com o volume, variedade e velocidade dos dados, pode gerar valor por meio de novas percepções e uma tomada de decisão otimizada.

3.3.2. Os Vs de Big Data

Para compreender melhor os desafios e as oportunidades referentes a big data, apresenta-se, a seguir, uma descrição dos principais Vs que circundam esse conceito.

O **volume** refere-se à quantidade sem precedentes de dados gerados nas últimas duas décadas. Estima-se que, a cada 2 ou 3 anos, a humanidade torna-se capaz de armazenar mais dados do que a quantidade armazenada desde o início da civilização [Hilbert 2022]. Quantidades massivas de dados, na ordem de terabytes e petabytes, passam a ser geradas a partir de diferentes fontes, sendo essas utilizadas pelas empresas, com foco no desenvolvimento de soluções e modelos de negócios orientados a dados e no aperfeiçoamento do processo de tomada de decisão [Marquesone 2016]. Embora esse aumento de volume trouxe benefícios para as empresas, que puderam compreender me-

lhor seus clientes, gerar modelos preditivos mais assertivos e compreender melhor um processo, tornou-se necessário um investimento em infraestrutura para possibilitar o armazenamento, o processamento e a análise de dados em larga escala.

Ao considerar o crescente volume de dados gerado e o crescente interesse das empresas em utilizar dados em suas estratégias de negócios, em [Mazzei and Noble 2019], os autores classificam o indivíduo (pessoa e dispositivo) como potenciais geradores de dados, e as empresas como processadoras de informação. Nesse contexto, por um lado, pessoas e dispositivos estão constantemente gerando dados a partir de tecnologias digitais e, por outro, as empresas estão investindo em formas inovadoras de transferir os dados para a criação de valor e desenvolver vantagem competitiva. Esse contexto gera o que se chama de “paradoxo da escolha”, que afirma que muita informação é tão problemática quanto pouca informação, no processo de tomada de decisão [Dong and Yang 2018].

A **variedade**, por sua vez, faz referência à ampla diversidade de dados obtidos atualmente, de fontes distintas, gerados por humanos e por máquinas, em formato estruturado e não estruturado [Marquesone 2016]. Por exemplo, dados gerados por humanos, contendo informações referentes ao sentimento e à propriedade intelectual de uma pessoa, são criados em diferentes contextos, tais como em redes sociais, na avaliação de um produto, bem como em serviços troca de mensagens. Dados gerados por máquinas são, também, originados de fontes variadas, tais como em sensores no contexto de IoT, em registros de *log* em servidores da Web, em agentes conversacionais e em sistemas corporativos. Além do formato texto, tais dados podem, também, ser obtidos no formato de imagens, áudios e vídeos, necessitando assim de novas estratégias de armazenamento, processamento e análise de dados.

A **velocidade** pode ser compreendida a partir de duas vertentes. A primeira, refere-se à velocidade com que os dados estão sendo gerados, contribuindo também para o aumento do volume. A segunda vertente, e, sendo essa a mais desafiadora, refere-se à velocidade com que esses dados estão sendo capturados e utilizados no processo de tomada de decisão [Marquesone 2016]. Por exemplo, para lidar com os bilhões de dispositivos conectados que geram dados constantemente, novas soluções de *streaming* de dados vêm evoluindo, para estar em conformidade com a velocidade necessária para processar e analisar tais dados. Nesse contexto, surge a necessidade do investimento em tecnologias que possibilitem o processamento em tempo real ou próximo ao tempo real, para que empresas tenham a capacidade de atuar sobre os dados à medida que eles são gerados.

Na Figura 3.5 é apresentado um resumo dos Vs existentes em big data. Além dos 3 Vs iniciais, o **valor** dos dados é, também, uma dimensão amplamente considerada na literatura para a definição e compreensão de big data. Essa dimensão refere-se à identificação de quais dados devem ser considerados em um projeto ou em uma empresa, para assim produzir informações valiosas [Lee 2017]. Denota a necessidade de avaliar previamente os objetivos de um projeto orientado a dados, para assim identificar quais bases deverão ser utilizadas para a análise.

Em complemento, como big data tem um potencial cada vez maior nos negócios, a **veracidade** tornou-se um atributo importante a ser considerado. Esse atributo está relacionado à qualidade dos dados, revelando a importância de, mesmo com dados massivos,

definir políticas que permitam às organizações verificar a confiabilidade em termos de precisão, consistência, pontualidade e integridade dos dados [Lee 2017]. Nesse contexto, insere-se a expressão “*garbage in, garbage out*”, expressando que a entrada de dados incorretos/imprecisos resultará na saída de dados incorretos/imprecisos. Dessa forma, com a premissa da tomada de decisão baseada em dados, avaliar a veracidade dos dados é fundamental para evitar ações baseadas em dados imprecisos.

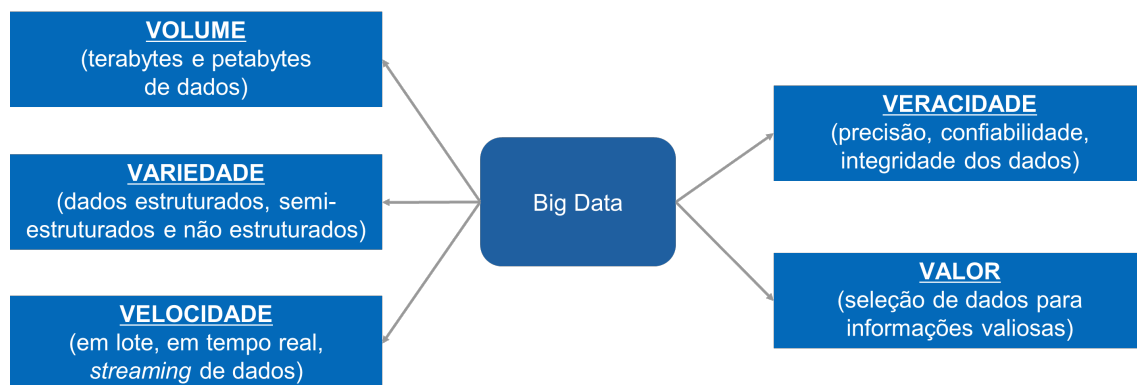


Figura 3.5. Os Vs de big data

Além dos Vs mencionados, outras dimensões foram sendo acrescentadas à definição de big data. Exemplos incluem a variabilidade, a validade e a versatilidade [Ranjan 2019, Khan et al. 2019]. Entretanto, tais dimensões ainda não foram amplamente discutidas no contexto de big data, havendo, assim, incertezas sobre suas compreensão.

3.3.3. Tecnologias de Big Data

Para atender aos requisitos que o volume, a variedade e a velocidade trouxeram aos projetos orientados a dados, novas técnicas e tecnologias foram sendo desenvolvidas conforme aumentava-se o interesse em soluções orientadas a dados. Nessa seção são apresentados exemplos dessas tecnologias, utilizadas nas diferentes etapas de um projeto de big data.

Em relação às tecnologias de big data para atender aos requisitos de armazenamento dos dados, uma tecnologia adotada no contexto de big data é a tecnologia NoSQL. O termo refere-se à “*Not only SQL*”, ou seja, descreve o surgimento de novas abordagens que vão além de somente o uso de bancos de dados relacionais, que utilizam *Structured Query Language* (SQL). Dessa forma, a tecnologia NoSQL faz menção a um conjunto de soluções de armazenamento voltado ao provimento de escalabilidade e disponibilidade dos dados, para atender às necessidades de dados em larga escala [Ramzan et al. 2019].

Em [Cattell 2011], o autor descreve seis propriedades em comum de soluções NoSQL:

1. A habilidade para escalar horizontalmente em ambientes de *cluster*.
2. A habilidade para prover a replicação dos dados nos servidores.
3. O uso de uma interface simples de acesso aos dados (em contraste ao uso de SQL de banco de dados relacionais);.

4. Um modelo de concorrência de dados menos rígido que a propriedade ACID dos bancos de dados relacionais.
5. O uso eficiente de índices para armazenamento dos dados.
6. A habilidade para adicionar dinamicamente novos atributos aos registros de dados.

Uma outra forma de categorizar as soluções existentes no contexto de tecnologias NoSQL é por meio do modelo de dados utilizado. Atualmente, as soluções existentes são divididas nas seguintes categorias: orientadas à chave-valor (e.g., Redis, Memcached); orientadas a documentos (e.g., MongoDB, CouchDB); orientadas a colunas (e.g., HBase, Cassandra) e; orientadas a grafos (e.g., Neo4J, AllegroGraph). As três primeiras compartilham uma estratégia similar, provendo o armazenamento dos dados a partir de um conjunto de chaves e valores específicos para cada chave. A última refere-se a uma estratégia de armazenamento de dados em que esses são organizados por meio de vértices e arestas, em um grafo.

Além do NoSQL, uma das soluções pioneiras em termos de tecnologias de big data é o Apache Hadoop, um *framework* de código aberto mantido pela Apache Software Foundation (ASF), com foco em prover mecanismos para o armazenamento e processamento distribuído de dados em larga escala. Seu lançamento oficial, em 2006, apresentava uma solução chamada Hadoop Distributed File System (HDFS), voltada ao armazenamento distribuído de grande volume de dados, oferecendo tolerância a falhas e abstração na distribuição de dados em um ambiente de *cluster*. Para o processamento dos dados, o Hadoop apresentava o MapReduce, possibilitando o processamento distribuído dos dados armazenados no HDFS [Goldman et al. 2012].

As funcionalidades e as capacidades oferecidas pelo Apache Hadoop passaram a ser amplamente adotadas por empresas, universidades e instituições de pesquisa, que necessitavam de um *framework* que atendesse aos requisitos decorrentes dos Vs de big data. Diante dessa aceitação, o código-fonte do Apache Hadoop passou a ser continuamente otimizado, surgindo novas contribuições, que resultaram em subprojetos, voltados à atender necessidades específicas em projetos de big data. Assim, atualmente há um ecossistema de soluções Hadoop, contendo um grande número de subprojetos, especializados, que atendem à demanda de uma empresa.

Na Figura 3.6 é apresentada uma evolução das soluções do ecossistema Hadoop geradas ao longo do tempo. Como exemplo, pode-se citar o subprojeto Apache Hive – voltado à provisão de uma solução de *data warehouse* para dados de larga escala; o subprojeto Apache Sqoop – com foco em possibilitar a importação e exportação de dados entre o Hadoop e uma solução de armazenamento de dados relacionais; o subprojeto Apache Ni-Fi – com foco em prover mecanismos para ingestão e fluxo de dados para o ambiente Hadoop. Além desses, cabe também citar o Apache Spark [Zaharia et al. 2016] – considerado uma evolução do MapReduce, provê um conjunto de mecanismos para o processamento de dados em larga escala, possibilitando o uso de diferentes modelos computacionais, além do processamento em lote, com o MapReduce. Oferece, também, mecanismos para o processamento de *streaming* de dados, com o Spark Streaming, o processamento de algoritmos de aprendizado de máquina, com o Spark MLlib, e, o processamento de grafos, com o Spark GraphX.

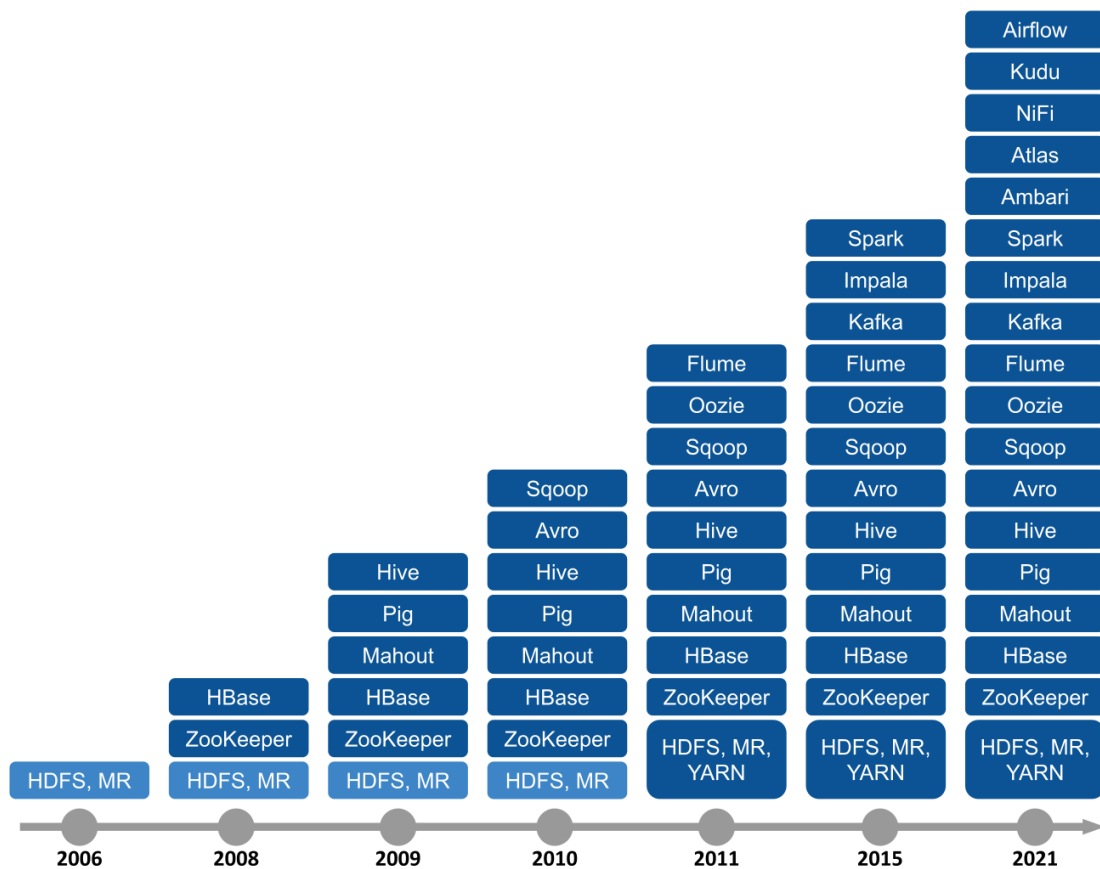


Figura 3.6. Soluções do ecossistema Hadoop. Adaptado de [Marquesone 2016]

Além das tecnologias apresentadas, destacam-se, também, aquelas referentes ao processamento de dados em tempo real, possibilitando extrair, processar e analisar dados conforme são gerados em um determinado evento. Nesse contexto, diversas soluções estão sendo utilizadas para esse fim. Um exemplo é o Apache Kafka – considerado atualmente uma plataforma de código aberto para *streaming* de dados, oferecendo mecanismos para a construção de *pipelines* distribuídos, captura e integração de dados e análise em tempo real dos dados. Além do Apache Kafka, exemplos de outras soluções para esse fim são: Apache Storm, Apache Flink e Amazon Kinesis.

É possível perceber, a partir de tais tecnologias, que a sua adoção trouxe tanto mudanças técnicas, quanto também mudanças organizacionais e culturais. A capacidade de extrair valor dos dados obtidos de diferentes fontes e formatos trouxe a necessidade de remoção de silos de dados, para assim serem analisados em uma visão integrada, aumentando a percepção de um determinado processo ou evento. Nesse contexto, surgiu também a proposta de um método de gerenciamento de dados denominado *data lake* [Sawadogo and Darmont 2021]. Além da proposta de remoção de silos de dados, integrando-os em um repositório comum à organização, a abordagem de *data lake* propõe o armazenamento dos dados em sua forma original, possibilitando que sejam armazenados sem a definição prévia de um *schema*. Nesse contexto, utiliza-se a abordagem

“*schema on read*”, no qual a estruturação dos dados é realizada de acordo com a necessidade de quem o utiliza, no momento de sua leitura.

3.3.4. Big Data Analytics

Além das tecnologias apresentadas, no contexto de big data insere-se, também, o termo big data analytics, uma área de estudo que contempla estratégias de análise de dados que considere o grande volume, variedade e velocidade de dados, visando a extração de valor. Conforme apresentado em [Mikalef et al. 2019], big data analytics pode ser compreendido como “uma nova geração de tecnologias e arquiteturas, projetadas para extrair valor econômico de volumes muito grandes de uma ampla variedade de dados, permitindo a captura, a descoberta e/ou a análise de alta velocidade”.

Ressalta-se, também, que big data analytics é considerado uma extensão dos seguintes termos: big data, analytics e *business intelligence* (BI). A partir desses três tópicos, big data analytics é composto pelo uso de tecnologias para gerenciar efetivamente grandes volumes de dados, melhorando as percepções de negócios a partir da análise de dados em larga escala [Côrte-Real et al. 2019]. Essa fusão resulta no uso de diversas técnicas, como estatística, mineração de dados, processamento de dados e gerenciamento de banco de dados.

Na literatura, as estratégias de big data analytics dividem-se em três categorias: a análise descritiva, a análise preditiva e a análise prescritiva. A seguir é apresentado um resumo sobre essas categorias.

- **Análise descritiva.** Tem como foco principal responder à pergunta: “o que aconteceu ou o que está acontecendo?”, provendo técnicas que permitem compreender um determinado processo, por meio da geração de métricas e indicadores. Nesse contexto, é comum a utilização de soluções de *business intelligence*, painéis de visualização de dados (*dashboards*) e a geração de relatório, possibilitando extrair métricas que possam auxiliar na compreensão e no monitoramento do desempenho de um determinado processo ou produto.
- **Análise preditiva.** Visa responder a questão: “o que irá acontecer?”, provendo mecanismos para que seja possível a realização de previsões sobre possíveis comportamentos futuros. Nesse contexto, é comum a utilização de técnicas estatísticas, de mineração de dados e de aprendizado de máquina, possibilitando gerar modelos preditivos a partir de dados históricos e atuais.
- **Análise prescritiva.** Considerada a mais complexa, visa responder a pergunta: “o que fazer para acontecer?” provendo técnicas voltadas à automação do processo de tomada de decisão, ou seja, voltado a tomada de decisão orientada a dados, sem a necessidade da intervenção humana, por meio de soluções em que o próprio sistema computacional seja capaz de prover um curso de ação para se alcançar um resultado desejado. Nesse contexto, é comum a adoção de técnicas de simulação, algoritmos de aprendizado de máquina e de *deep learning*.

A Figura 3.7 apresenta um resumo das categorias apresentadas, contendo exemplos de técnicas utilizadas e possíveis aplicações de negócios. Embora a análise descritiva

seja considerada a mais simples, entre as três apresentadas, sua aplicação é necessária para se obter os dados indispensáveis para as outras análises, a preditiva e a prescritiva. Além disso, mesmo nessa primeira categoria, percepções significativas podem ser obtidas, por meio do levantamento de métricas e de indicadores de um determinado processo.

Denota-se, nesse contexto, que a inovação tecnológica é um elemento-chave para o uso de big data analytics. Portanto, empresas que desejam automatizar o processo de tomada de decisão devem reformular sua infraestrutura, incluindo tecnologias de big data e ferramentas de analytics, capazes de suportar o volume massivo, a variedade de dados e a velocidade de análise de dados em tempo real. Essa mudança de infraestrutura é atualmente considerada um requisito, dado que ferramentas tradicionais de gerenciamento de dados não fornecem recursos em termos de alta escalabilidade, flexibilidade, armazenamento e processamento distribuído. Adicionalmente, um dos desafios existentes na extração de *insights* valiosos dos dados é a falta de talento humano com a habilidade necessária para lidar com dados estruturados e não estruturados, nos processos de extração, processamento e análise de dados [Härting et al. 2018]. Conforme afirmado por [Härting et al. 2018], sem uma equipe de dados adequada e com as habilidades certas, é impossível realizar uma estratégia de big data analytics.

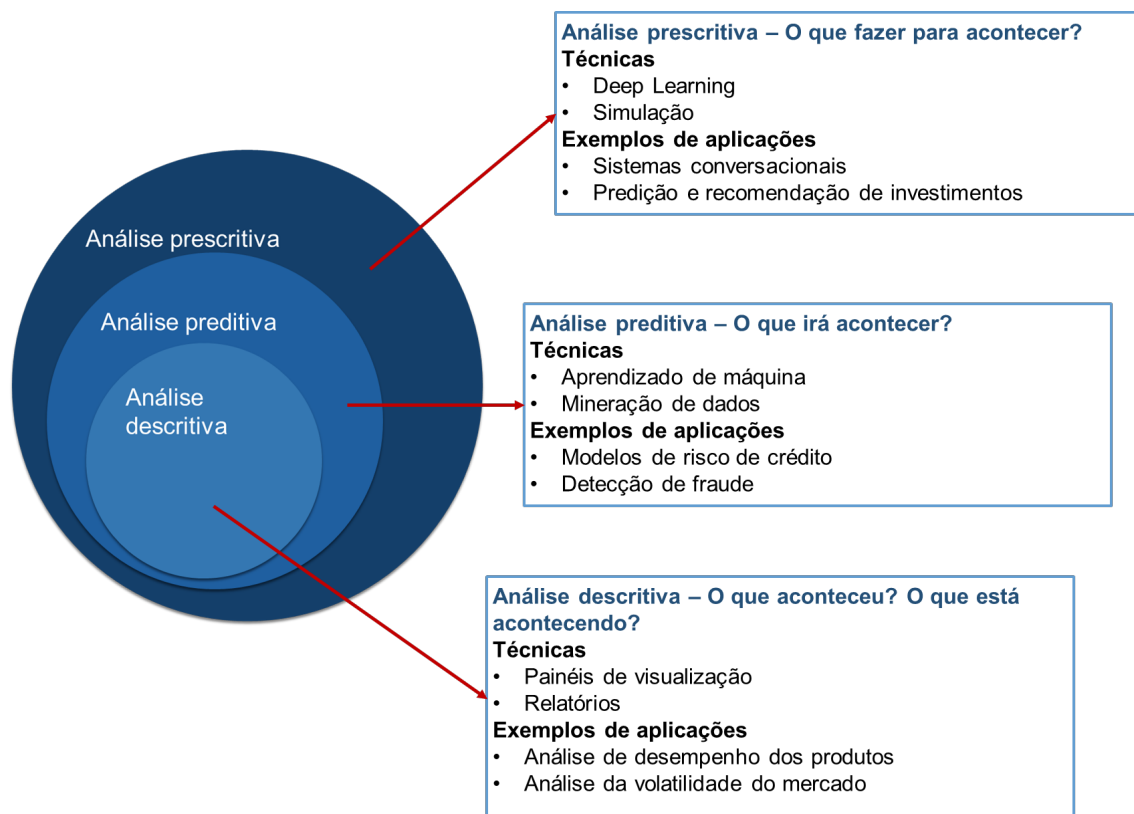


Figura 3.7. Categorias de big data analytics.

Por fim, observa-se que big data analytics oferece inúmeras oportunidades de inovação nas empresas. No entanto, é necessário, também, um claro entendimento dos executivos tanto sobre big data quanto sobre o objetivo para essa adoção. Somente com essa mentalidade é possível escolher as técnicas e as tecnologias adequadas e alinhadas às

estratégias de negócios [Lehrer et al. 2018].

3.4. Tecnologias Digitais

Além das tecnologias de big data existentes para lidar com o grande volume, variedade e velocidade de dados gerados atualmente, os últimos anos também foram marcados pela introdução de novas tecnologias digitais, gerando mudanças significativas na sociedade [Hilbert 2022]. A evolução dessas tecnologias culminaram na caracterização de um novo estágio da revolução industrial, denominado indústria 4.0. Diante desse cenário, as empresas estão identificando que tecnologias como computação em nuvem, Internet das Coisas (*Internet of Things* - IoT), Blockchain e Inteligência Artificial (IA) possibilitam reformular modelos de negócios, oferecendo vantagens estratégicas [Martínez-Caro et al. 2020].

Nos últimos anos tem-se discutido, também, a importância de tecnologias digitais para o avanço na transição para a economia circular [Khan et al. 2022]. Portanto, essa seção apresenta uma contextualização sobre tecnologias digitais, descrevendo suas características, aplicações e benefícios.

3.4.1. Computação em Nuvem

Ao abordar o tema "tecnologias digitais", uma das tecnologias amplamente adotadas nos últimos anos é a computação em nuvem. Esse termo foi formalmente definido em 2011 pelo órgão americano responsável pela padronização e métricas na área tecnológica chamado National Institute of Standards and Technology (NIST). Computação em nuvem é, então, compreendido como "um modelo que permite o acesso a um conjunto de recursos compartilhados e configuráveis como por exemplo, servidores, armazenamento, aplicativos e serviços. Podem ser provisionados e liberados com o mínimo de esforço de gerenciamento e interação do provedor de serviços" [Liu et al. 2011].

Identifica-se, nessa definição, que a computação em nuvem proporciona recursos de processamento e armazenamento a partir de características como: a flexibilidade, no qual usuários da nuvem podem obter os recursos conforme a sua necessidade; o acesso via rede, possibilitando o acesso remoto aos recursos e serviços disponibilizados; o compartilhamento de recursos, possibilitando meios para compartilhar recursos computacionais com diferentes aplicações e usuários; e a elasticidade, possibilitando que a quantidade de recursos possa ser ampliada ou reduzida de forma automatizada. Tais características possibilitam que empresas reduzam seus custos operacionais e com a aquisição de hardware para seus *data centers* locais, mas obtenham acesso rápido e escalável aos recursos computacionais da nuvem, de acordo com a sua demanda.

Uma outra característica de computação em nuvem refere-se à divisão de responsabilidades entre o provedor e o usuário de nuvem. Uma das maneiras de identificar como ocorre essa divisão é por meio das três categorias de modelos de computação em nuvem [Liu et al. 2011]:

- Infraestrutura como Serviço (*Infrastructure as a Service* - IaaS): nesse modelo, o usuário de nuvem tem acesso aos recursos de infraestrutura, tais como de armazenamento, processamento e virtualização, possibilitando a execução de diferentes

aplicações e processos, mas com a responsabilidade de gerenciar a instalação e a configuração deles.

- Plataforma como Serviço (*Platform as a Service* - PaaS): além de ter acesso aos recursos computacionais, nesse modelo o provedor de nuvem disponibiliza recursos como ferramentas, bibliotecas e linguagens de programação. Dessa forma a responsabilidade do gerenciamento das aplicações é dividida com o usuário da nuvem.
- Software como serviço (*Software as a Service* - SaaS): nesse modelo, o gerenciamento da infraestrutura e das aplicações são de responsabilidade do provedor de nuvem, de forma que o usuário possua acesso às aplicações disponíveis, porém sem o poder de administrá-las ou modificá-las.

Os modelos de computação em nuvem apresentados podem ser disponibilizados em diferentes modelos de organização [Ahmad et al. 2021], tais como:

- Pública: modelo de organização na qual a infraestrutura de nuvem é fornecida para acesso a um público geral. Tem-se, como exemplos, a Amazon Web Services (AWS), a Microsoft Azure e o Google Cloud Platform (GCP), no qual cada nuvem é controlada e administrada por uma empresa específica.
- Privada: refere-se à infraestrutura de nuvem na qual o uso é restrito a uma organização. Como exemplo, pode-se citar uma nuvem de uma empresa específica, utilizada apenas pelos colaboradores internos desta empresa.
- Comunitária: faz referência a um modelo de organização no qual um grupo de usuários com interesse em comum compartilham os recursos da nuvem. Como exemplo, pode-se citar a nuvem de uma universidade pública, que oferece os recursos aos pesquisadores realizarem experimentos científicos.
- Híbrida: nesse modelo, uma combinação dos modelos anteriores é feita (pública, privada e comunitária), havendo uma interação entre elas, por meio de protocolos e de interfaces. Como exemplo, pode-se citar uma empresa que utiliza recursos de nuvem privada e de nuvem pública.

Apesar de diversos benefícios identificados em relação à adoção da computação em nuvem, alguns desafios e barreiras foram identificados e apontados na literatura. Fatores como custo, privacidade, *compliance*, controle e complexidade ainda são mencionados como impeditivos para que a adoção integral ocorra. Em adição, a questão de segurança dos dados e de aplicações hospedadas em ambientes de nuvem são consideradas, atualmente, as principais barreiras, necessitando de avanços nesse domínio [Ahmad et al. 2021].

Cabe destacar, também, o surgimento mais recentemente do termo *fog computing* [Dastjerdi and Buyya 2016]. Esse conceito é compreendido como uma proposta de arquitetura que visa oferecer recursos computacionais (e.g., processamento distribuído, armazenamento, rede) próximos à borda da rede, onde os dados são gerados pelos usuários. Essa abordagem possibilita reduzir a quantidade de dados trafegados pela rede,

atenuando, assim, alguns dos desafios existentes na adoção de modelos de computação em nuvem, como latência e capacidade de banda [De Donno et al. 2019].

3.4.2. Internet das Coisas

No cenário tradicional a Internet é composta por uma infraestrutura global de rede de computadores, na qual pessoas a utilizam para gerar, consumir e compartilhar informações. A Internet das Coisas (*Internet of Things* - IoT) pode, dessa forma, ser compreendida como um conceito que gerou uma mudança de paradigma da Internet [Li et al. 2015]. Nesse cenário, além dos computadores, demais objetos passam a fazer parte da infraestrutura global, gerando, consumindo e interagindo com outras pessoas e objetos, no mundo físico e virtual. Como exemplo desses objetos pode-se citar as etiquetas (*tags*) de identificação por radiofrequência (*Radio-Frequency Identification* - RFID), os sensores, atuadores e dispositivos móveis.

A heterogeneidade é uma das características predominantes em IoT. Como são utilizados diferentes objetos nesse contexto, tais como etiquetas RFID, sensores de temperatura, de movimento, sensores corporais e câmeras de rede sem fio, os dados gerados não são estruturados, contendo diferentes formatos [Jiang et al. 2014]. Cabe ressaltar que, além da heterogeneidade do formato dos dados, deve-se lidar também com uma ampla variedade de padrões de comunicação em que esses são transmitidos (ZigBee, WiFi, 6LowPan, etc.), bem como uma variedade na forma de interação (baseado em eventos, baseado em tempos, etc.).

No contexto de IoT há um grande número de objetos que coletam e transmitem dados por meio de diversos eventos ocorridos, resultando em uma grande expansão no volume de dados. Para suportar o gerenciamento e a análise do grande volume de dados gerados, é essencial a existência de um ambiente de alta escalabilidade. Soluções nas quais os objetos são tratados apenas como objetos periféricos, e os dados são transmitidos para uma entidade central tem limitações em termos de suporte a um grande número de objetos, e conseqüentemente, problemas de desempenho.

Objetos como sensores e atuadores existentes em aplicações de IoT são caracterizados normalmente como dispositivos com baixa capacidade, em termos de: processamento, memória, comunicação, armazenamento e energia. Por tal motivo, uma abordagem muito utilizada é transmitir os dados em sua forma original para uma entidade externa. Porém, como esses dispositivos também possuem limitações em termos de rede, uma solução mais adequada é ter um processo que permita a redução de dados, evitando sobrecarregar a rede. Além desse fator, a redução de dados em diferentes níveis em um cenário de IoT tem como objetivo auxiliar no processamento dos dados, pois busca diminuir a latência do processamento, que deve ocorrer em tempo real.

A computação em nuvem atua como um importante papel para tornar essas funcionalidades possíveis de serem implementadas, dado a necessidade de um sistema que seja escalável e distribuído. Entretanto, embora a adoção desse recurso ofereça maior disponibilidade dos serviços, sua utilização pode afetar o desempenho das aplicações, por aumentar a latência. Nesse contexto, soluções de *fog computing* têm sido exploradas para prover meios para atender ao desempenho das aplicações de IoT [De Donno et al. 2019].

3.4.3. Blockchain

Dentre as tecnologias digitais mencionadas nesse capítulo, a tecnologia blockchain é considerada uma das mais disruptivas, com potencial valor de sua aplicação. Embora seja associado muitas vezes somente ao contexto de criptomoedas, blockchain pode ser utilizado em inúmeras outras aplicações, conforme é apresentado nessa seção.

Atribui-se a Haber e Stornetta, a invenção da tecnologia blockchain, a partir da publicação de um artigo, em 1990, na qual é descrita uma técnica para lidar com a autenticação de documentos digitais, como áudio, texto e imagens [Haber and Stornetta 1990]. A proposta do artigo era apresentar uma técnica capaz de garantir que um documento digital não foi alterado, sem haver a necessidade de uma entidade central para prover essa garantia. Para isso, foram utilizados elementos que formavam um sistema de verificação de *timestamp*. Posteriormente, em 2008, essa tecnologia foi utilizada como um elemento chave para a estrutura do funcionamento da moeda Bitcoin, proposto pela pessoa até hoje desconhecida, de pseudônimo Satoshi Nakamoto [Nakamoto 2008].

Blockchain pode ser compreendido como um conjunto de blocos dispostos em uma lista de registros, em que o vínculo desses são criptografados, resistindo assim, à alteração de seus dados. Nesse contexto, cada bloco é composto por um valor *hash* criptografado do bloco anterior, um *timestamp* e os dados da transação. Além disso, os blocos existentes nessa cadeia são descentralizados e distribuídos, impedindo que um bloco consiga fazer uma alteração sem que afete retroativamente os blocos subsequentes [Rajasekaran et al. 2022]. A partir dessa abordagem, duas características principais podem ser observadas no blockchain:

- **Imutabilidade.** Um dos benefícios de se utilizar blockchain é por ser considerado quase impossível ser alterado os dados mantidos nos blocos. Isso ocorre pela exigência de que qualquer modificação seja realizada somente com um consenso da maioria dos nós da rede, tornando-a assim mais segura e transparente [Rajasekaran et al. 2022]. Dessa forma, existe a confiança de que os dados registrados não foram corrompidos e/ou modificados. Além disso, a imutabilidade também garante que, uma vez que um dado é registrado, ele é enviado para verificação e validação, não podendo ser excluído ou alterado sem que isso reflita nos demais nós [Deepa et al. 2022].
- **Descentralização.** Blockchain é compreendido como um livro-razão distribuído, no qual o gerenciamento das transações ocorre de maneira descentralizada, sem a necessidade de uma entidade central que faça o controle da rede. É essa capacidade de garantir a autenticidade das transações sem uma entidade centralizadora confiável que torna o blockchain uma tecnologia considerada disruptiva [Mattila 2016]. Para tornar isso possível de forma segura, são utilizados protocolos, criptografia e mecanismos de consenso para o registro e a validação dos dados. Essa característica pode resultar em benefícios como a redução de custos, maior transparência e aperfeiçoamento de processos operacionais [Deepa et al. 2022].

A partir dessas características, blockchain tem sido utilizado em diferentes áreas de negócio. A seguir são apresentados exemplos de como as empresas estão utilizando

essa tecnologia.

- Moedas digitais. Após o surgimento do bitcoin, novas criptomoedas surgiram no mercado, também utilizando blockchain para não necessitar de entidades centrais regulamentadoras. Essas moedas são consideradas inspiradas no bitcoin, e foram criadas com o foco na proposição de uma versão otimizada desta moeda. Como exemplo, pode-se citar: ZCach, Litecoin e Morero [Yuan and Wang 2018].
- Contratos inteligentes. A partir de blockchain pode-se obter *scripts* que permitem a criação de modelos de contratos personalizados, facilitando o seu gerenciamento, além de oferecer a segurança provida pelo blockchain. Como exemplo, pode-se citar a plataforma Ethereum, que possibilita a construção de aplicações descentralizadas.
- Gestão da cadeia de suprimentos. Empresas da área de cadeia de suprimentos estão utilizando a tecnologia blockchain para aperfeiçoar o processo de rastreamento de materiais e produtos ao longo da cadeia produtiva. Essa abordagem possibilita obter maior transparência do processo, aumentar a confiança entre os envolvidos em uma determinada operação, bem como garantir a procedência de um material ou produto [Mattila 2016]. Um exemplo é a utilização de blockchain na gestão de suprimentos da indústria alimentícia, possibilitando rastrear o alimento em suas diversas etapas, como o cultivo, a colheita e o transporte.

Apesar do potencial valor identificado na adoção de blockchain, alguns desafios são, também, apresentados na literatura. Um dos principais desafios refere-se à escalabilidade, uma vez que cada nó de uma rede blockchain armazena todos os dados históricos das transações realizadas [Lu 2019]. Entretanto, dado o crescente volume de transações e, conseqüentemente, o crescente volume de dados gerados em um sistema blockchain, torna-se necessário o investimento em pesquisas que possibilitem prover maior escalabilidade ao sistema, resolvendo, assim, questões referentes à alta taxa de latência, elevado consumo de energia e baixa taxa de transferência [Khan et al. 2021].

3.4.4. Inteligência Artificial (IA)

Dentre as tecnologias digitais apresentadas, uma que tem recebido cada vez mais destaque e sendo utilizada para aperfeiçoar e inovar diferentes domínios é a Inteligência Artificial (IA). Conforme define Russel e Norving, IA pode ser compreendido como o estudo e projeto de agentes inteligentes, em que esse agente é um sistema que percebe o seu ambiente e executa ações que maximizam suas chances de sucesso [Russel et al. 2013].

Atualmente um conjunto vasto de aplicações fazem uso de IA, nos mais diferentes domínios, tais como:

- Agentes conversacionais. Consiste em *chatbots* e assistentes conversacionais, voltados à comunicação entre humanos e plataformas computacionais, utilizando recursos de Processamento de Linguagem Natural (PLN), por voz e/ou por texto.

- Sistemas de reconhecimento facial. Solução que utiliza técnicas de visão computacional voltadas ao reconhecimento de pessoas a partir de imagens e vídeos. Tais sistemas são aplicados para diversos fins, tais como o desbloqueio de aparelhos, controle em ambientes restritos, autenticação para operações bancárias, entre outros.
- Veículos autônomos. Sistemas de controle computacional que possibilitam a condução de um veículo, sem a necessidade de um condutor humano. Para isso, são utilizados sensores com recursos de IA, voltados ao processamento de dados diversos, como imagens de câmeras e de radares.

Ao se considerar o tema de IA, insere-se também o de aprendizado de máquina. Esse conceito é compreendido como uma subárea de IA, com foco em possibilitar que sistemas computacionais sejam capazes de aprender de forma automatizada a partir de uma experiência, sem necessitar da programação explícita. Uma das possíveis maneiras de se categorizar os algoritmos existentes em aprendizado de máquina é por três categorias: aprendizado supervisionado, não supervisionado e aprendizado por reforço [Jo 2021]. Na Figura 3.8 são descritas as categorias existentes, bem como um conjunto de exemplos de seus algoritmos.

O aprendizado supervisionado pode ser compreendido como aquele em que o aprendizado é obtido a partir de um conjunto de dados rotulados. São normalmente utilizados para fins de classificação ou regressão. Na classificação, o algoritmo busca atribuir uma determinada classe a uma instância. Para isso, algoritmos de Redes Bayesianas, por exemplo, podem ser utilizados. Na regressão, tem-se como objetivo estimar um valor numérico contínuo para uma determinada instância, podendo ser utilizado, por exemplo, os algoritmos de regressão logística.

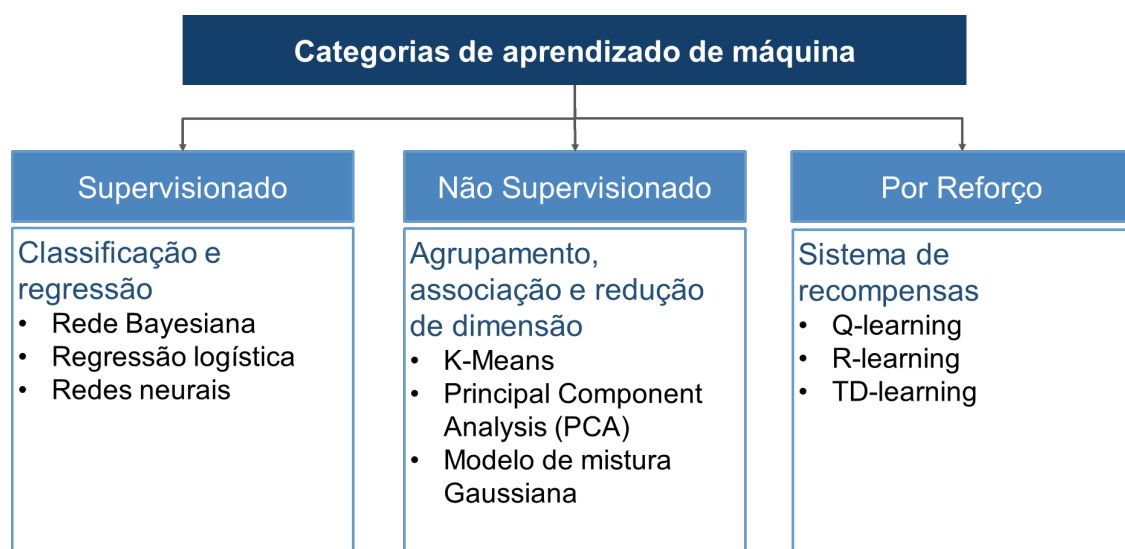


Figura 3.8. Categorias de aprendizado de máquina

O aprendizado não supervisionado possui esse nome por não necessitar de uma base de dados rotulada para o processo de aprendizado. Seu objetivo é agregar bases de

dados não rotuladas, criando grupos que compartilham características em comum, com base em uma métrica de similaridade [Rashidi et al. 2019]. Um exemplo é o uso do algoritmo K-Means para estratégias de agrupamento e o algoritmo de Análise de Componentes Principais (*Principal Component Analysis - PCA*), caso o objetivo seja atuar na redução de dimensionalidades.

Por fim, o aprendizado por reforço refere-se a algoritmos no qual o treinamento é obtido a partir de um sistema no qual parâmetros são ajustados, visando maximizar recompensas e minimizar punições, a partir de uma interação entre um agente e um ambiente [Jo 2021]. Uma possibilidade, nessa categoria, é o uso do algoritmo Q-learning para fazer a modelagem de um agente.

Embora estudos teóricos sobre IA tenham sido realizados muito antes do cenário atual de digitalização, a quantidade massiva de dados gerados atualmente, somado ao crescente poder computacional dos servidores, impulsionou o avanço dessa área de estudo. Nesse contexto, pode-se citar a crescente adoção de algoritmos de aprendizado profundo (*deep learning*), uma área de aprendizado de máquina que utiliza redes neurais artificiais para a resolução de problemas. O principal elemento dos algoritmos de *deep learning* é sua eficiência na aplicabilidade de recursos de álgebra linear, apoiados nos últimos anos por processos de computação de CPU (*Central Processing Unit*) e GPU (*Graphics Processing Unit*). Nesse contexto, diversas ferramentas e bibliotecas tem sido disponibilizadas para esse fim, tais como Keras, TensorFlow e PyTorch.

Apesar dos avanços referentes às aplicações de IA nos últimos anos, um dos principais desafios enfrentados atualmente refere-se ao problema da IA explicável, uma vez que o processo de decisão dos algoritmos são comumente considerados uma “caixa-preta”, não sendo possível interpretar a lógica utilizada para o resultado apresentado. Dessa forma, essa área de estudo visa a identificação de métodos e processos que permitam a compreensão das saídas geradas por algoritmos de IA, podendo assim avaliar fatores como sua taxa de exatidão e possíveis erros que podem resultar em impactos negativos como imparcialidade e discriminação.

Além das tecnologias digitais apresentadas nessa seção, é importante salientar o fato de que outras tecnologias também fazem parte desse contexto, sendo utilizadas pelas empresas em suas estratégias de negócio, em iniciativas para transformação digital e adoção da indústria 4.0. Como exemplo, pode-se citar: tecnologias de realidade virtual e realidade aumentada, computação quântica e tecnologia 5G [Martínez-Caro et al. 2020]. Todas essas tecnologias digitais estão impactando não somente os negócios, mas também toda a sociedade. Baseia-se em mudanças profundas em todas as indústrias. Portanto, é um tema de interesse para diferentes áreas de estudo, como ciência da computação, administração de empresas, sistemas de informação e engenharia elétrica [Hausberg et al. 2018, Vial 2019].

Por fim, é possível analisar que a interação entre tecnologias digitais e big data pode ser vista em dois aspectos principais. Em primeiro lugar, as tecnologias digitais estão promovendo a geração crescente de dados. Dados gerados por humanos em serviços de mensagens, avaliação de produtos em plataformas de *e-commerce*, mídias sociais como Facebook e Twitter, bem como dados gerados por máquinas, como registros de *log* de servidores de computação em nuvem e sensores de IoT estão sendo criados em alta

velocidade, fornecendo informações de contextos internos e externos. Em segundo lugar, tecnologias de big data e técnicas de big data analytics têm fornecido mecanismos para melhorar o processo de tomada de decisões de negócios. Ao extrair percepções de diversos dados, as empresas são capazes de obter um melhor entendimento de seus clientes ou reduzir custos automatizando ou otimizando seus processos [Vial 2019].

3.5. Big Data e Tecnologias Digitais Aplicadas à Economia Circular

Embora existam empresas que já incorporam estratégias de economia circular em seus modelos de negócios, identificar como ela deve ser implantada ainda é considerado um desafio em aberto [Rani and Saha 2021]. A economia circular é considerada um novo paradigma, que requer mudanças em diversos aspectos, exigindo novas abordagens no *design* de produtos, mudança sistêmica na cadeia produtiva e mudanças no comportamento do consumidor em relação ao uso e descarte de um produto [EMF 2014]. Diante do exposto, identifica-se que a transição para a economia circular necessita de educação e conscientização de produtores e consumidores, bem como de modelos de negócios inovadores, que incorporem a sustentabilidade com a proposição de valor [Barros et al. 2020]. Nesse contexto, big data e tecnologias digitais são identificadas como recursos-chave para promover essa transição.

Após apresentada uma contextualização sobre big data e tecnologias digitais, esta seção tem como objetivo descrever como a transição para a economia circular, visando cadeias produtivas mais sustentáveis, podem se apoiar com o uso dessas tecnologias. Para isso, conforme ilustrado na Figura 3.9, a descrição será apresentada a partir de abordagens voltadas aos seguintes aspectos: colaboração; transparência e rastreabilidade; compreensão e tomada de decisão; monitoramento e previsão.

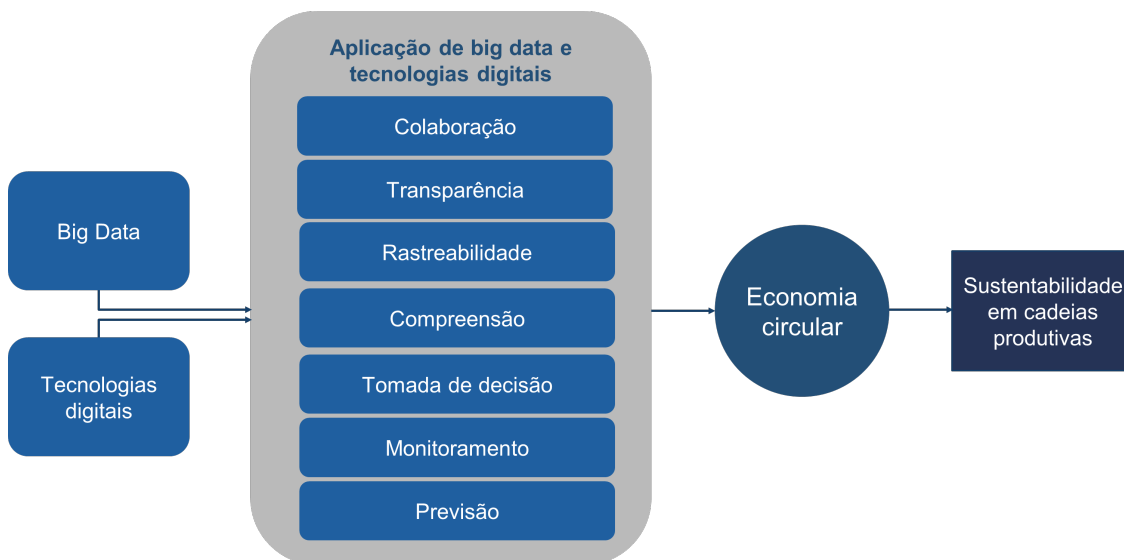


Figura 3.9. Aplicação de big data e tecnologias digitais à economia circular

3.5.1. Estratégias para Colaboração

Técnicas de big data analytics podem permitir que as organizações analisem e extraiam informações úteis de eventos de registros digitais (e.g., registros de *logs* de servidores da Web, atividade de navegação, dados de sensor), bem como dados gerados por humanos (e.g., *feedback*, postagens) [Lehrer et al. 2018]. Essa habilidade provê um conhecimento ampliado sobre o processo do ciclo de vida do produto, resultando em maior transparência e visibilidade de suas ações. Como consequência, a obtenção desses dados pode contribuir para que atuem de forma colaborativa, agregando e compartilhado as informações entre os colaboradores [Garcia-Torres et al. 2021].

Diante desse contexto, empresas que buscam a transição para a economia circular são desafiadas a atuarem de modo colaborativo, permitindo que múltiplos *stakeholders* se envolvam nas estratégias do modelo de negócio. Assim, a transição para a economia circular, a partir da teoria de *stakeholders*, destaca a necessidade de que diferentes partes da cadeia produtiva atuem ativamente na definição da proposta de valor e nas estratégias circulares, de forma que todos possam atuar de forma colaborativa e serem beneficiados com os resultados da estratégia estabelecida [Gupta et al. 2019, Sandvik and Stubbs 2019].

Adicionalmente, ao se considerar a economia circular, a colaboração é considerada um fator-chave, em relação à responsabilidade coletiva. A responsabilidade não deve ser transferida de uma parte para a outra, por exemplo, do vendedor para o consumidor. Ao contrário, identifica-se que, no decorrer do ciclo de vida de um produto, todos os *stakeholders* devem ser responsabilizados. Sem essa responsabilidade coletiva, torna-se inviável garantir que a sustentabilidade será aplicada em todo o ciclo de vida do produto, uma vez que uma decisão tomada em uma etapa impacta diretamente a etapa seguinte [Moorhouse 2020].

A partir de uma revisão sistemática da literatura, Marquesone e Carvalho identificam que tecnologias de big data podem contribuir para a colaboração na economia circular [Marquesone and Carvalho 2022a]. Dentre os benefícios identificados, pode-se citar:

- Auxílio na integração de informações entre *stakeholders*. A capacidade de tecnologias de big data como o Hadoop, somado ao conceito de *data lake*, pode contribuir para que os dados coletados de diferentes fontes ao longo da cadeia produtiva sejam integrados, possibilitando uma análise que incorpore informações obtidas da colaboração entre os *stakeholders*. Ou seja, uma plataforma colaborativa de dados é criada, possibilitando agregar as informações referentes ao ciclo de vida do produto. Nesse contexto, tecnologias de computação em nuvem podem também ser utilizadas, provendo serviços que auxiliem no acesso remoto à plataforma pelos usuários, além de possibilitar a implantação das tecnologias de big data, como o Apache Hadoop, em um ambiente flexível e de alta escalabilidade.
- Melhoria na visibilidade das etapas da cadeia produtiva. A partir de técnicas de big data analytics, as empresas tornam-se habilitadas a analisarem dados obtidos de toda a cadeia produtiva, podendo, assim, obter uma visão dos impactos econômicos, ambientais e sociais, causados ao longo da produção. Esta habilidade contribui na colaboração, uma vez que todo o contexto de produção é avaliado, necessitando

do apoio mútuo para a resolução dos desafios identificados. Também possibilita que os resultados das análises sejam compartilhados com todos os envolvidos, aumentando, assim, a visibilidade da cadeia.

- Auxílio no gerenciamento de redes diretas e redes reversas. Conforme já mencionado, a economia circular inclui estratégias que envolvem todo o ciclo de vida do produto, incluindo a fase de fim de vida. Nesse contexto, a logística reversa desempenha um papel fundamental nas estratégias de circularidade, provendo meios para que os materiais e produtos sejam reaproveitados e utilizados em novos ciclos. Dessa forma, técnicas de big data analytics podem contribuir para a colaboração de dados obtidos tanto nas redes diretas quanto nas redes reversas, para assim, prover percepções referentes à ligação entre elas.

Apesar da identificação de que big data contribui para a colaboração entre *stakeholders* na transição para a economia circular, um desafio relacionado à esse fato é a cultura organizacional. Uma cultura organizacional pode ser definida como um conjunto de crenças e valores compartilhados que dão significado aos membros de uma instituição e lhes fornecem regras de comportamento em sua organização. Esta cultura é capaz de influenciar diversos fatores na organização, como objetivos, estrutura, cooperação, comunicação e tomada de decisão [Sun 2008]. Entretanto, mesmo diante dos benefícios percebidos, a adoção de cultura orientada a dados ainda é considerada um desafio, inibindo o acesso aos dados, até mesmo entre departamentos internos. Normalmente, ainda identifica-se a existência de silos de dados, reduzindo a possibilidade de compartilhamento dos dados e, como consequência, minimizando a chance de extrair informações significativas dos dados. Portanto, não somente os dados relativos às práticas de sustentabilidade e circularidade deixam de ser compartilhados, como também outros dados que poderiam ajudar a organização a aprimorar seus modelos de negócios.

3.5.2. Estratégias para Transparência e Rastreabilidade

Para que a transição para a economia circular ocorra, são identificados alguns obstáculos, resultantes do modelo de processo linear até então adotado. Um desses obstáculos se refere à falta de transparência e rastreabilidade na cadeia produtiva, de forma que todos os *stakeholders*, incluindo os consumidores, tenham visibilidade das práticas adotadas em cada etapa, possibilitando assim monitorar e identificar o impacto ambiental e social em todo o processo produtivo [Moorhouse 2020].

Identifica-se, na literatura, que a falta de transparência pode ocorrer por diversos fatores, como, por exemplo, por motivos estratégicos, na qual a empresa não divulga seus dados com foco em obter vantagem competitiva. Nesse cenário, mesmo a empresa divulgando relatórios de sustentabilidade, ainda é comum que ocorra o que é atualmente denominado *greenwashing*, caso em que os dados são apresentados com vieses, ocultando informações ou criando uma falsa aparência de sustentabilidade.

Entretanto, diante da complexidade existente em toda a cadeia produtiva de uma determinada indústria, ocorre, em determinadas situações, que empresas sequer possuem a aquisição dos dados necessários para compreender seu impacto ambiental e social. Dessa forma, a adoção de tecnologias de big data, técnicas de big data analytics e tecno-

logias digitais podem contribuir para prover meios que possibilitem a captura e a análise desses dados, trazendo maior transparência de todo o processo.

Considerando também o aspecto rastreabilidade, um exemplo de estratégia voltada à cadeia produtiva de uma indústria é a adoção de blockchain. A partir dessa tecnologia, torna-se possível o mapeamento dos impactos ambientais e sociais causados nas diferentes etapas de produção, mantendo um histórico imutável sobre o processo produtivo, permitindo, assim, o rastreamento de cada operação, a sua autenticidade, para, assim, avaliar questões referentes à sustentabilidade. Além disso, essas informações podem ser acessadas a partir de dispositivos de IoT, como a Identificação por Radiofrequência (*Radio Frequency Identification* - RFID), possibilitando que diferentes *stakeholders*, incluindo os consumidores, tenham acesso a tais dados de maneira facilitada, obtendo informações que possam auxiliá-los na decisão de uma compra mais consciente. Esta abordagem também facilita o rastreamento do produto em sua fase pós-consumo, auxiliando nas práticas referentes à logística reversa, uma vez que permite mapear a identificação do local, das características e do estado do produto nesta etapa.

3.5.3. Estratégias para Compreensão e Tomada de Decisão

Outra percepção relacionada à economia circular é que, mesmo empresas que já capturam dados ao longo de toda a cadeia produtiva, essas não possuem uma visão clara de todo o ciclo de vida do produto, resultando em fatores como a obscuridade dos dados e a incerteza das decisões tomadas. Assim, a organização não tem visibilidade suficiente que lhe permita ter uma visão de quais estratégias e processos estão sendo aplicados ao longo do ciclo de vida de um produto, não identificando em quais etapas estão ocorrendo problemas de sustentabilidade e quais devem ser aperfeiçoados. Nesse contexto, a aplicabilidade de big data e tecnologias digitais podem auxiliar essas empresas a reduzirem incertezas durante o ciclo de vida de um produto, obtendo uma maior compreensão sobre esse [Thorisdottir and Johannsdottir 2019].

Soluções guiadas por dados podem ser aplicadas como um fator-chave para auxiliar as empresas a coletar dados e a medir o impacto de suas ações, seu desempenho e sua estratégia de negócios para buscar uma abordagem mais sustentável para a transição para a economia circular [Gupta et al. 2019]. Nesse contexto, a adoção de técnicas de análise descritiva pode prover uma maior compreensão das métricas e dos indicadores de sustentabilidade, possibilitando que a empresa obtenha uma visão mais clara de seus impactos ambientais e sociais.

Conforme apresentado por Marquesone e Carvalho, a adoção de big data pode contribuir na compreensão e no processo de tomada de decisão voltados à economia circular a partir de uma série de fatores, tais como [Marquesone and Carvalho 2022a]:

- No processo de tomada de decisão, obtendo dados que contribuam para a compreensão sobre hábitos e estratégias de produção e consumo de um determinado produto. Para isso, sensores e dispositivos de IoT podem contribuir na captura de dados do processo produtivo.
- Na análise e na compreensão referente à utilização e ao rendimento dos recursos utilizados ao longo das etapas da cadeia produtiva, podendo, assim, identificar pontos

a serem otimizados. Nesse contexto, algoritmos de IA podem também contribuir para gerar análises que relatem o melhor curso de ação em relação a medidas que reduzam a quantidade de recursos utilizados e que gere menor impacto ambiental.

- Na geração de conhecimento referente ao processo de concepção de um produto, avaliando e mensurando o impacto das decisões tomadas durante todo o seu ciclo de vida.
- Na compreensão das condições de trabalho em todas as etapas da cadeia produtiva, a partir da análise integrada dos dados obtidos de fornecedores e colaboradores internos, possibilitando, assim, identificar condições sociais em partes específicas da cadeia que precisam ser solucionadas.
- Na identificação de pontos de desperdício ao longo das etapas da cadeia produtiva, provendo percepções que auxiliem na compreensão de como reduzir tais desperdícios, ou como utilizá-los em novos ciclos de produção.

3.5.4. Estratégias para Monitoramento e Previsão

Ao utilizar modelos preditivos e em tempo real, as empresas podem obter maiores percepções sobre processos e recursos que devem ser compartilhados em um modelo de economia circular [Gupta et al. 2019]. Além disso, a captura de eventos a partir de sensores e dispositivos de IoT permite ampliar a capacidade de compreensão dos dados ao longo da cadeia produtiva. Por exemplo, a análise do impacto ambiental na etapa de distribuição do produto pode ser obtida a partir de sensores que medem a emissão de gases poluentes. Dessa forma, pode-se identificar fatores que podem contribuir para a redução da emissão.

A partir das estratégias apresentadas, identifica-se a necessidade do compartilhamento dos dados com *stakeholders*, ao utilizar big data e/ou tecnologias digitais para a transição de economia circular. Nesse contexto, é importante destacar que a interação com os dados compartilhados pode ocorrer como remetentes e receptores de dados, de acordo com suas necessidades. Cada colaborador pode contribuir com seus dados existentes, para prover mais subsídios de dados a serem comunicados com os demais.

Ao compartilhar dados entre as partes interessadas em uma plataforma, permitindo que cada um atue como emissor e receptor, torna-se possível ampliar o leque de dados disponíveis e aumentar a perspectiva sobre tais dados. Nesse contexto, mesmo que diferentes *stakeholders* recebam os mesmos dados, cada um pode ter uma perspectiva diferente sobre o mesmo, atuando de acordo com suas habilidades.

Por exemplo, ao considerar as grandes empresas atuando como remetentes, essas podem compartilhar as ferramentas e técnicas usadas para extrair, armazenar, processar e analisar os dados usados para avaliar o desempenho em termos de sustentabilidade e atividades da economia circular. Podem, também, compartilhar indicadores obtidos na medição de desempenho da empresa, para que outras empresas e colaboradores tenham uma visão transparente das práticas da empresa. Podem compartilhar o progresso da empresa em relação à economia circular, inspirando outras empresas e instituições a adotarem práticas semelhantes. Em adição, podem compartilhar informações relacionadas ao comportamento do consumidor, obtidas a partir de abordagens como campanhas de *marketing*, análise de mídia social e análise de registros de *log*.

Ao atuar como receptora, tais empresas podem receber dados de outras empresas para obter uma análise de seu desempenho. Podem, também, receber dados de seus fornecedores e de todas as partes envolvidas no ciclo de vida do produto, a fim de compreender as questões de sustentabilidade envolvidas em cada etapa. Também podem receber dados de pesquisadores, instituições e educadores, tendo acesso a experimentos, provas de conceito e estudos de caso que podem contribuir para a formulação de um modelo de negócio circular e para o processo de tomada de decisão.

3.6. Cenário de Aplicação: Economia Circular e a Indústria Têxtil

Uma abordagem que tem sido considerada como impulsionadora para a promoção de uma indústria mais sustentável é o princípio da economia circular [Mungcharoen et al. 2021]. Considerada uma alternativa ao modelo tradicional de economia linear, a economia circular na indústria têxtil e do vestuário busca uma mudança em toda a cadeia produtiva, desde a produção ao consumo de um produto, estendendo seu ciclo de vida, gerando mudanças positivas não somente no âmbito econômico, mas também social e ambiental, por meio de uma abordagem restaurativa e regenerativa no design dos produtos. Dessa forma, a indústria têxtil é avaliada nessa seção, havendo um relato referente aos desafios de sustentabilidade atualmente enfrentados. Na sequência, apresenta-se, também, um estudo inicial de estudos de caso referentes à adoção de economia circular nessa indústria, contendo uma análise sobre a possibilidade de aplicação de big data e tecnologia digitais nessas estratégias.

3.6.1. Desafios de Sustentabilidade da Indústria Têxtil

A indústria têxtil, utilizada como cenário de aplicação nesse estudo, é considerada uma indústria de substancial poder econômico, provendo empregos diretos e indiretos em nível global. Entretanto, identifica-se, nessa indústria, uma série de desafios ambientais e sociais que necessitam ser solucionados, para que essa torne-se sustentável. Considerando esse contexto, em [Marquesone and Carvalho 2022b], doze desafios foram identificados na indústria têxtil. Nesse artigo, as autoras também categorizaram os desafios a partir das dimensões volume, variedade e velocidade dos dados, utilizadas na definição de big data. Seguindo essa mesma abordagem, a seguir são apresentados os desafios identificados, contendo uma descrição sobre esses.

Desafio 1 - Grande volume de roupas produzidas. Há uma estimativa de que 111 milhões de toneladas de fibras têxteis foram utilizadas em 2019, para atender a quantidade de roupas e demais artigos têxteis a serem produzidos, representando um crescimento de 50% em relação a 2008 [Matlin et al. 2020]. Uma mudança significativa observada na indústria têxtil é a pressão por roupas mais baratas ao longo dos anos, resultando em seu uso em períodos mais curtos de tempo. Essa demanda crescente gera o aumento da produção e a quantidade crescente de resíduos têxteis [Koszewska 2019, Brydges 2021]. Este cenário denomina-se atualmente como *fast fashion* e é considerado um dos desafios atuais relacionados à sustentabilidade nesta indústria, tanto nos aspectos ambientais como sociais.

Desafio 2 - Grande volume de matérias-primas extraídas. Embora o grande volume de roupas produzidas possa resultar em crescimento econômico, esse é conside-

rado um desafio de sustentabilidade, devido aos impactos ambientais e sociais causados. Dentre os impactos ambientais, insere-se a contínua extração de matérias-primas para a produção de novas roupas e demais artigos têxteis.

A indústria têxtil é altamente dependente de matéria-prima em sua cadeia produtiva. Um produto têxtil pode ser derivado tanto de recursos naturais, como o algodão, quanto de recursos sintéticos, como o poliéster, obtido a partir de recursos fósseis, oriundos do petróleo. Nesse cenário, ambos os recursos geram impactos ambientais, em diferentes níveis [Chen et al. 2021]. Por exemplo, a produção de algodão resulta em vasta exploração do solo para monoculturas, além de quantidades excessivas de água para seu cultivo. O poliéster, por sua vez, resulta em processos com elevada emissão de gases poluentes, contaminação do solo e a água, além da geração de microplásticos, que são descarregados nos oceanos após cada lavagem, prejudicando a saúde humana e a marinha.

Desafio 3 - Grande volume de gases poluentes emitidos. Estudos apontam que a indústria têxtil é considerada a segunda indústria mais poluente do mundo, ficando atrás somente da indústria de gás e do petróleo [Saha et al. 2021]. Dessa forma, conforme apresentado na definição da Fundação Ellen MacArthur, torna-se necessário uma estratégia de economia circular que considere também o uso de energias renováveis, visando assim a redução da emissão desses gases nas diversas etapas da cadeia produtiva dessa indústria [EMF 2012]. Um estudo apresentado pela empresa de consultoria McKinsey, em 2020, considerando a indústria têxtil, de vestuário e de calçados, foi identificado que 77% das emissões ocorrem nas etapas de produção. Identificou-se, também, que 3% dessas emissões ocorrem na etapa de varejo, 3% na de transporte, 20% no uso do produto e 3% no fim de uso do produto [McKinsey 2020].

Desafio 4 - Grande volume de água utilizada. Identifica-se, na indústria têxtil, que diversas etapas da cadeia produtiva requer uma quantidade elevada de água. Ao todo, estima-se que são consumidos 93 bilhões de metros cúbicos de água anualmente nessa indústria [Ütebay et al. 2020]. Esse cenário é identificado já na etapa de cultivo da matéria-prima, uma vez que o algodão, uma das matérias-primas mais utilizadas, requer um alto consumo de água para a irrigação do solo. Além disso, etapas como o tingimento e a estamparia de tecidos também são caracterizados pelo alto consumo.

Desafio 5 - Grande volume de resíduo gerado. Um outro desafio existente na indústria têxtil refere-se ao final de vida dos produtos têxteis, gerando, atualmente, quantidades elevadas de resíduos, que, em sua grande parte, são incinerados ou despejados em aterros, causando a emissão de gases poluentes, contaminando o solo e a água. Estudos estimam que 92 milhões de toneladas de resíduos têxteis sejam gerados anualmente [Li and Leonas 2021]. Esses resíduos são gerados tanto nas etapas pré-consumo, resultantes de sobras de tecidos e demais materiais têxteis no processo industrial, quanto nas etapas pós-consumo, resultante do descarte de roupas e outros itens têxteis pelos consumidores.

Identifica-se, diante desse desafio, a necessidade de estratégias que possibilitem o reaproveitamento dos materiais, seja por ciclos técnicos ou biológicos. Para isso, medidas voltadas ao compartilhamento de responsabilidade do produtor são também cruciais, para prover ao consumidor mecanismos que o permita e o estimule a prover um destino

adequado às roupas descartadas, como por exemplo, por meio de programas de coleta.

Desafio 6 - Grande variedade de *stakeholders*, etapas e materiais envolvidos ao longo da cadeia produtiva. Embora sejam apresentados três desafios diferentes, esses são integrados, por terem uma relação entre seus impactos. A indústria têxtil possui, ao considerar toda a sua cadeia produtiva, um conjunto vasto de *stakeholders* envolvidos na produção de um item têxtil, como por exemplo: produtores, fornecedores, fabricantes, *designers* e varejistas. Somente na produção de uma jaqueta jeans, por exemplo, é possível a ocorrência de mais de cem operações, envolvendo uma quantidade vasta de material, recursos utilizados e profissionais envolvidos no processo [Marquesone and Carvalho 2022b].

Desafio 7 - Grande variedade de informações desconhecidas e ocultas sobre o processo produtivo e o impacto ambiental causado. A indústria têxtil, além de conter uma ampla variedade de *stakeholders*, etapas e materiais envolvidos ao longo da cadeia produtiva, também é considerada uma indústria fragmentada e geograficamente dispersa, o que torna mais desafiador uma análise do impacto ambiental e social em cada etapa e processo ao longo da cadeia produtiva [Thorisdottir and Johannsdottir 2019]. Nesse contexto, identifica-se a falta de transparência das empresas dessa indústria, não sendo possível mensurar e monitorar o impacto causado por essas. Somado a isso, tem-se também a falta de rastreabilidade, possibilitando a identificação dos impactos de forma isolada, em cada etapa da produção, até o fim de uso do produto.

Desafio 8 - Grande variedade de condições inadequadas de trabalho na indústria. Diversos estudos apontam que, além dos impactos ambientais causados por essa produção excessiva na indústria, há também desafios sociais, referentes às condições inadequadas de trabalho [Saha et al. 2021]. Esse cenário ocorre principalmente em países em desenvolvimento, como Sri Lanka e Bangladesh, que, após o desmantelamento do Acordo Multifibras (*Multi-Fibre Arrangement - MFA*), passou a receber uma grande demanda de produção de artigos têxteis, devido à mão de obra barata nas fábricas [Ernst et al. 2005]. Entretanto, a discussão sobre esse contexto ganhou maior notoriedade em 2013, após o desabamento do edifício Rana Plaza, em Bangladesh, que abrigava diversas fábricas têxteis independentes, resultando na morte dos colaboradores [Marquesone and Carvalho 2022b].

Diante da pressão pela produção cada vez mais rápida e com materiais de baixo custo, identifica-se a existência, até os dias atuais, de trabalhadores em condições de trabalho precárias, como baixos salários, cargas horárias excessivas e fábricas em instalações inadequadas. Ou seja, mesmo sendo somente um desafio social aqui apresentado, ele engloba diversos fatores, tais como o bem-estar humano, a equidade de gênero, acesso à educação e exploração de trabalho infantil [Huq et al. 2014].

Desafio 9 - Grande variedade de componentes químicos utilizados na produção. Além do elevado consumo de água, uma outra preocupação ambiental referente à indústria têxtil se deve à elevada quantidade de componentes químicos utilizados ao longo da cadeia produtiva [Gupta et al. 2022]. Exemplos incluem fertilizantes, pesticidas e corantes, utilizados tanto em tecidos naturais quanto sintéticos. Como resultado, são liberadas substâncias tóxicas no meio ambiente, poluindo o ar, a água e a saúde das pessoas [Marquesone and Carvalho 2022b].

Desafio 10 - Grande velocidade de produção de roupas. Diretamente relacionado

ao desafio 1, identifica-se um aumento crescente na velocidade com que as roupas estão sendo produzidas atualmente. Estimuladas pelo modelo de negócio *fast fashion*, empresas da indústria têxtil e do vestuário estão reduzindo cada vez mais o intervalo entre as coleções, havendo varejistas que apresentam ao mercado novas coleções a cada quatro semanas, estimulando o consumidor a adquirir uma nova peça de roupa, por meio de estratégias de *marketing* [Saha et al. 2021]. Além de haver uma pressão por custos menores na produção, esse curto prazo resulta da dificuldade das empresas em prever de forma precisa a demanda para uma determinada coleção, fazendo com que estoques excessivos sejam gerados, aumentando, assim, a quantidade de roupas descartadas.

Desafio 11 - Grande velocidade de consumo das roupas. Em adição do desafio 10, identifica-se, também, uma redução na quantidade de vezes que as pessoas utilizam uma peça de roupa. Um estudo relata que as pessoas estão utilizando as roupas 50% a menos, quando comparado a duas décadas atrás [Koszewska 2019]. Além da obsolescência percebida, gerada pela introdução constante de novas coleções, identifica-se também, que as pessoas não se sentem mais emocionalmente ligadas a uma peça de roupa, tornando, assim, mais fácil seu descarte. Além disso, estudos também atribuem esse desafio ao fato de que as novas gerações estão perdendo o hábito de reparar uma peça de roupa, sendo mais fácil a compra de um novo item [Diddi and Yan 2019].

Desafio 12 - Grande velocidade de descarte das roupas. Por fim, em consonância com os desafios 10 e 11, está a grande velocidade com que as roupas estão sendo descartadas. Atualmente, considera-se que a roupa é um dos itens mais subutilizados, uma vez que seu descarte ocorre por vezes ainda em boas condições de uso. Como resultado, além do impacto ambiental causado, identifica-se, também, uma perda de valor econômico, uma vez que o uso das roupas descartadas poderia ser prolongado por meio de estratégias de circularidade, ou o material dessas ser utilizado em novos ciclos, como insumos para a produção de novos materiais e produtos.

Na Figura 3.10 é apresentado um resumo dos desafios relatados. Diante do exposto, identifica-se que a transição para a economia circular, visando tornar a indústria têxtil sustentável, é desafiadora, porém necessária. Dado os impactos ambientais e sociais causados, considera-se que o modelo de produção linear nessa indústria está atingindo seus limites, tanto por pressões internas, uma vez que o custo pelos recursos naturais estão cada vez mais elevados, diante da escassez, quanto de pressão externa, conforme a sociedade torna-se mais consciente e exige que as empresas atuem de modo responsável.

Diversas iniciativas estão sendo criadas atualmente para demandar que as empresas na indústria têxtil e da moda revelem suas informações e atuem de forma mais transparente. Essas se referem tanto a iniciativas regulatórias, por meio da imposição de políticas e leis, quanto iniciativas relacionadas à divulgação de relatórios que relatem como as empresas estão se posicionando em relação à transparência de suas ações sustentáveis. Por exemplo, uma iniciativa voltada para o objetivo de aumentar a transparência na indústria da moda por meio dos dados é o *Fashion Transparency Index* (FTI), uma iniciativa criada em 2017, e que a cada ano apresenta uma avaliação em termos de transparência de empresas na indústria da moda, em relação ao seu desempenho sustentável.

Embora o FTI ofereça uma maior compreensão sobre o cenário de sustentabilidade na indústria têxtil e da moda, ao analisar as estratégias de divulgação dos dados de

sustentabilidade das empresas avaliadas pelo FTI, um estudo identificou que a transparência ainda é limitada nessas indústrias, de forma que a maioria das marcas ainda hesitam na divulgação de suas práticas de sustentabilidade, havendo ainda uma cultura voltada à tornar os dados obscuros, impedindo, assim, o avanço para a economia circular na indústria [Jestratijevic et al. 2021].

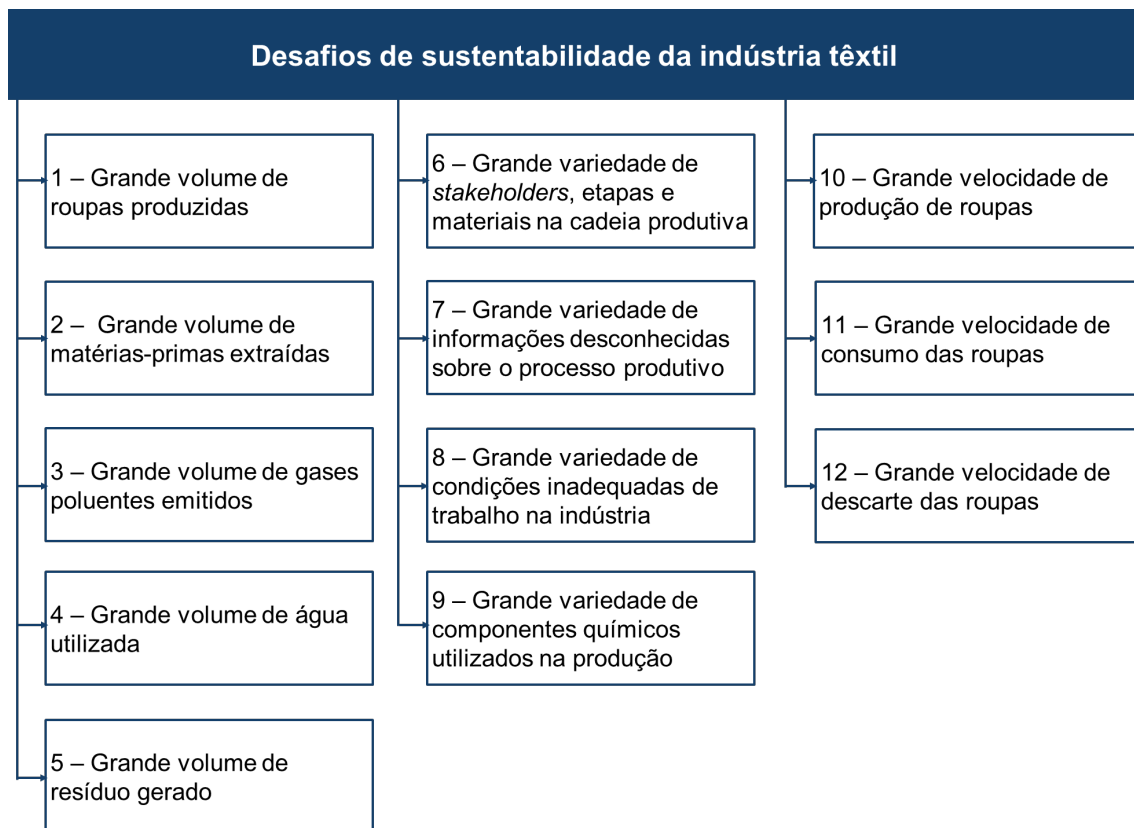


Figura 3.10. Desafios de sustentabilidade da indústria têxtil

WikiRate [WIKIRATE 2021] é uma ferramenta aberta e colaborativa, lançada em 2013, com o objetivo de levantar e responder questões importantes sobre os impactos corporativos. Atualmente, esta ferramenta é utilizada em colaboração com um conjunto diversificado de empresas, universidades e ONGs com foco na coleta e na extração de percepções de dados corporativos. A ferramenta foi desenvolvida a partir de um projeto europeu denominado Collective Awareness Platform for Sustainability and Social Innovation (CAPS), que tinha como objetivo analisar o desempenho ESG das empresas. A partir deste projeto, foi identificada a necessidade de um repositório público aberto de tais dados, aumentando a transparência e permitindo que muitos stakeholders tenham acesso e a possibilidade de extrair *insights* dele.

O Higg Index [SAC 2021], desenvolvido pela Sustainable Apparel Coalition (SAC) em 2011, é definido como uma ferramenta de autoavaliação para medir o desempenho de sustentabilidade de marcas na indústria de vestuário, calçados e têxtil. A coalizão é formada por mais de 40 empresas dos setores mencionados, como Patagonia, Nike e Walmart, que atuam em conjunto para fornecer um padrão de métricas para medir e monitorar práticas sustentáveis. Um dos objetivos do Higg Index é proporcionar a colaboração de

ferramentas e práticas sustentáveis entre seus membros, facilitando o processo de medição, monitoramento e comunicação de seu desempenho. Portanto, as empresas que fazem parte do SAC podem utilizar um conjunto de ferramentas para essa medição, possibilitando uma análise comparativa entre cada empresa [Gonçalves and Silva 2021].

Uma outra ferramenta, voltada especificamente para a economia circular, é a *Circulytics* [EMF 2021], desenvolvida pela equipe de dados e métricas da Fundação Ellen MacArthur, com o objetivo de oferecer o suporte para a transição de modelos de negócios para a economia circular. Essa ferramenta oferece uma solução para ser utilizada por diferentes tipos de indústrias, independente de seu tamanho e complexidade. Dentre as funcionalidades da ferramenta, destaca-se a avaliação do nível de circularidade dos produtos ou serviços de uma determinada empresa. Essa avaliação é realizada a partir de um conjunto de indicadores, denominados *Material Circularity Indicator*.

Dado os desafios apresentados, bem como exemplos de estratégias que visam solucioná-los, ressalta-se a importância de avanços na transição para a economia circular na indústria têxtil. Nesse contexto, o apoio por meio de big data e tecnologias digitais é identificado como elemento habilitador para auxiliar nessa transição.

3.6.2. Modelos de Negócios Circulares da Indústria Têxtil e a Aplicação de Big Data e Tecnologias Digitais

Nessa seção é apresentado um conjunto de cinco estudos de caso referentes a modelos de negócios de economia circular na indústria têxtil, descrevendo como big data e/ou tecnologias digitais podem contribuir para a melhoria das estratégias voltadas à sustentabilidade. Os estudos de caso são obtidos de uma plataforma denominada *Knowledge Hub*¹, que oferece uma biblioteca on-line de livre acesso a milhares de estudos de caso de economia circular. Mais especificamente, os estudos de caso selecionados fazem parte da coleção *World Circular Textiles Day (WCTD)*², contendo estudos de caso específicos da indústria têxtil. A partir desse levantamento, o projeto WCTD visa realizar uma investigação anual do *status* da economia circular na indústria têxtil, até o ano de 2050, onde é esperado que a indústria seja completamente circular.

Estudo de caso 1 - Transparência e rastreabilidade a partir de tecnologias digitais. Conforme apresentado na Seção 3.5.2, a transição para a economia circular requer que empresas sejam mais transparentes e promovam a rastreabilidade de sua cadeia produtiva, para que, assim, essas possam obter e divulgar uma melhor compreensão do impacto ambiental e social causado ao longo do ciclo de vida do produto, além de se comprometer na redução dos impactos identificados. Para contextualizar essa abordagem, esse estudo de caso refere-se à uma marca de moda italiana, especializada na produção de jeans utilizando algodão orgânico, e adotando práticas sustentáveis, como a produção com baixo consumo de água.

O uso de tecnologias digitais foi realizado pela marca, com foco em prover maior transparência e rastreabilidade aos seus consumidores, mostrando a esses detalhes do processo produtivo. Para isso, a empresa realizou uma parceria com uma empresa de

¹<https://knowledge-hub.circle-lab.com/>

²<https://worldcirculartextilesday.com/>

tecnologia chamada Provenance³, que, visando atender aos requisitos de transparência e rastreabilidade, além de oferecer uma melhor experiência ao cliente, desenvolveu um passaporte digital para cada produto da marca, possibilitando ao consumidor obter as informações referentes ao rastreamento do produto a partir de um QR Code gravado na peça de roupa. Ao fazer a leitura do QR Code, o consumidor é destinado à página da marca, onde é apresentado um histórico do processo produtivo em uma linha do tempo, permitindo a esse verificar informações do processo produtivo, tais como a identificação dos componentes utilizados para a produção em cada etapa do processo, os fornecedores parceiros e as certificações obtidas por esses.

Para prover maior confiança dos dados apresentados aos consumidores, foi utilizada a tecnologia blockchain para cadastrar as informações ao longo da cadeia produtiva, permitindo o registro imutável de cada evento. A partir dessa estratégia, a marca fortalece o relacionamento com seus fornecedores, dando maior visibilidade a esses e provendo uma confiança mútua. Possibilita, também, conscientizar os consumidores da marca, estimulando-os a compreenderem como o produto foi produzido e o impacto gerado nessa produção.

Estudo de caso 2 - Serviço de gerenciamento de fim de vida de produto têxtil.

Foi identificado na Seção 3.6.1 que um dos desafios de sustentabilidade na indústria têxtil refere-se à vasta quantidade de resíduos têxteis que são descartados em aterros ou são incinerados, causando severos danos ambientais e prejudicando a saúde humana. Dessa forma, soluções no contexto de economia circular na indústria têxtil devem considerar meios para a redução de resíduos e o gerenciamento do final de vida de um produto têxtil, evitando a ocorrência de impactos ambientais negativos. Nesse cenário, esse estudo de caso apresenta um exemplo de como a indústria têxtil pode atuar na resolução desse desafio.

O estudo de caso refere-se à uma eco-organização francesa, que, em parceria com as empresas de indústria têxtil e calçadista do país, oferece mecanismos para que os resíduos têxteis gerados tanto nas fases de produção quanto na de consumo, tenham um destino adequado, por meio de estratégias de circularidade⁴. Ela atua com foco na política estabelecida na França de Responsabilidade Estendida do Produtor, que determina que as empresas sejam responsáveis pelo gerenciamento dos produtos em fim de vida.

A empresa oferece uma série de iniciativas para contribuir na promoção de um destino adequado aos resíduos têxteis. Por exemplo, a partir de seu site e de um aplicativo móvel, a empresa fornece um serviço no qual as pessoas podem localizar pontos de entrega voluntária de produtos têxteis mais próximos, além de instruções referentes à classificação do produto e ao processo de entrega. A empresa também disponibiliza vídeos, realiza *webinars* e apresentam informações referentes à segunda vida de têxteis e calçados, promovendo maior conscientização e educação tanto aos produtores quanto aos consumidores.

Identifica-se, nesse estudo de caso, que a empresa atua com foco em capturar, integrar e compartilhar dados referentes à gestão do fim de vida dos produtos têxteis. Dessa

³<https://www.provenance.org/case-studies/haikure>

⁴<https://refashion.fr/>

forma, técnicas de big data analytics podem contribuir para que esses dados sejam melhor compreendidos e transformados em informações relevantes para o alcance de maior sustentabilidade na indústria. Além disso, serviços de computação em nuvem e o desenvolvimento de uma API podem facilitar a troca de dados entre os *stakeholders*, uma vez que os serviços providos são oriundos do apoio mútuo entre eles.

Estudo de caso 3 - Serviço de aluguel de jeans. Na Seção 3.2.5 foi apresentado um conjunto de estratégias de circularidade com foco em tornar mais sustentável a produção e o consumo de produtos. Nesse contexto, nesse estudo de caso é apresentado um modelo de negócio circular baseado no aluguel de roupas, ao invés da compra.

O estudo de caso refere-se a uma empresa com base na Holanda, que oferece um serviço de aluguel de jeans, possibilitando que seus clientes façam anualmente a troca por novas peças⁵. Ou seja, ao invés de pagar pela aquisição de uma calça jeans, por exemplo, o cliente paga uma mensalidade ao serviço de aluguel, e, ao final de um ano, o cliente pode decidir se deseja ficar com a calça permanentemente, ou se deseja trocar por uma nova peça. A empresa também oferece serviços de reparo, caso o cliente desejar continuar o uso da peça alugada.

Além de promover a circularidade do jeans a partir do serviço de aluguel, a empresa também é responsável pela fabricação de suas peças, utilizando materiais orgânicos para sua produção. Como estratégia de circularidade, também realiza a reciclagem do jeans para a produção de novas peças.

Os recursos de tecnologias digitais são utilizados tanto para promover uma melhor experiência ao cliente, que pode, a partir do site e do aplicativo da empresa, realizar a busca e o aluguel do jeans, de forma facilitada. Além disso, a empresa também promove informações sobre os impactos ambientais e sociais gerados ao longo da cadeia produtiva, e oferece vídeos e treinamentos para que estudantes e consumidores, visando contribuir para o aumento da educação e da conscientização sobre economia circular e sustentabilidade.

Estudo de caso 4 - Modelo de negócio baseado na troca de roupas. Além da possibilidade de prolongar o uso de um produto e promover a circularidade por meio de serviços de aluguel, conforme apresentado no estudo de caso anterior, esse estudo de caso apresenta uma estratégia de circularidade alternativa, focada no serviço de troca de roupas.

Esse estudo de caso se refere a uma *startup* social holandesa, que visa atuar na indústria têxtil e da moda de forma sustentável, reduzindo o volume de roupas produzidas, a quantidade de recursos naturais extraídos, bem como reduzindo o volume de resíduos têxteis⁶. Para isso, a proposta do modelo de negócio da empresa é baseado na troca de roupas. Os clientes da *startup* entregam peças de roupas usadas para a loja, que então avalia as condições da roupa, entregando ao cliente uma espécie de moeda, chamada *swaps*. A partir disso, o cliente pode utilizar seus *swaps* para escolher outras peças de roupas da loja. Dessa forma, torna-se possível que peças ainda em boas condições de uso continuem sendo utilizadas por outras pessoas, prolongando assim o seu uso. Essa estratégia

⁵<https://mudjeans.eu/>

⁶<https://www.the-swapshop.com/>

também contribui para aumentar a conscientização dos consumidores, uma vez que eles identificam outras alternativas de renovar o guarda-roupa, que não incluía a compra de novas peças.

Além desse modelo de negócio baseado na troca de roupas, a *startup* também realiza outras ações, voltadas ao contexto de economia circular. As roupas que não estão em condições de uso, por exemplo, são enviadas a empresas parceiras, que fazem a reciclagem dos produtos. Roupas em boas condições mas que não são de interesse nas trocas, são doadas para instituições de caridade. Além disso, a empresa também oferece cursos de empreendedorismo social, contribuindo na disseminação do conhecimento de modelos de negócios circulares.

Em relação à adoção de tecnologias digitais e de big data, a *startup* tem buscado oferecer uma melhor experiência ao cliente, possibilitando que esse faça aquisição de peças de roupa em um *e-commerce*, uma vez que esse tenha *swaps* para realizar a troca. Essa também tem realizado parcerias com outras empresas para testar soluções que permitem o rastreamento dos produtos trocados, para assim obterem uma melhor compreensão do fim de vida dos produtos. Além dessas estratégias, identifica-se que técnicas de big data analytics, como a de análise prescritiva, podem contribuir para automatizar o processo de avaliação de um produto a ser trocado, identificando, assim, suas condições, características e valor.

Estudo de caso 5 - Produção de tecidos a partir de matérias-primas recicladas. Além de prolongar o uso de um produto têxtil, a transição para a economia circular inclui, também, estratégias para a reciclagem de resíduos têxteis, para que esses possam ser utilizados em novos ciclos, evitando assim a extração de novas matérias-primas. Há, também, o objetivo de se utilizar matérias-primas que possam ser regeneradas ou restauradas, evitando a escassez de recursos.

Esse estudo de caso refere-se à uma empresa brasileira classificada como empresa B certificada, por ter uma visão socioambiental em seu modelo de negócio. Nesse caso, sua estratégia é, especificamente em promover a sustentabilidade na indústria têxtil⁷. Seu modelo de negócio baseia-se na comercialização de tecidos sustentáveis, produzidos a partir de fios de algodão reciclados e matérias-primas recicladas, como garrafas PET descartadas.

O processo de reciclagem adotado pela empresa dispensa o uso de produtos químicos, bem como o consumo de água. Após a reciclagem, os tecidos são utilizados para a produção de novos artigos têxteis, como roupas, bolsas, estofados e almofadas. Além dos tecidos produzidos a partir de matérias-primas recicladas, a empresa também atua com tecidos produzidos com algodão orgânico certificado. Nesse contexto, a adoção de big data e tecnologias digitais pode contribuir para que a empresa faça um melhor rastreio das matérias-primas adquiridas para sua produção de novos tecidos, bem como provendo o rastreio de seus tecidos aos clientes, que poderão obter uma informação detalhada da origem e do processo utilizado em toda a produção.

Considerando o contexto de economia circular, identifica-se que as tecnologias digitais podem contribuir para a empresa compartilhar com os consumidores as etapas de

⁷<https://ecosimple.com.br/>

seu processo produtivo, para que esses tenham maior visibilidade de como a reciclagem é adotada, bem como dos impactos causados em cada etapa. Além disso, sensores e dispositivos no contexto de IoT podem auxiliar no mapeamento e na coleta dos resíduos, obtidos de empresas parceiras. Entretanto, os estudos de caso apresentados mostram que a adoção desses recursos para modelos de negócios de economia circular ainda está em seus primórdios, havendo uma ampla oportunidade de aperfeiçoamento das estratégias adotadas.

3.7. Considerações Finais

A sociedade e o mundo estão em constante transformação. A era atual é marcada pela quantidade de dados gerados, bem como a introdução de inovações oriundas de big data e tecnologias digitais. Simultaneamente, identificamos, na era atual, uma preocupação com a necessidade de mudar os hábitos de produção e consumo, visando cadeias produtivas mais sustentáveis e consumidores conscientes, para assim obter uma harmonia entre a prosperidade econômica, o bem estar social e a preservação do meio ambiente. Nesse contexto, a economia circular surge como uma proposta promissora para se atingir esse objetivo.

Embora a relação entre big data, tecnologias digitais e economia circular não seja tão clara em um primeiro momento, observa-se que a partir de uma ligação entre esses conceitos, torna-se possível avançar na transição para a sustentabilidade. A partir de um estudo centrado na indústria têxtil, caracterizada por desafios ambientais e sociais ao longo de sua cadeia produtiva, foi apresentado um conjunto de possibilidades referentes à aplicabilidade de big data e tecnologias digitais com foco na transição para a economia circular. Assim, foi possível explorar, a partir desse estudo, diferentes meios com que essa aplicabilidade pode ser realizada, provendo benefícios como a colaboração, a transparência, a rastreabilidade, a previsão e o apoio no processo de tomada de decisão de estratégias circulares.

A partir dessa interseção de temas, pode-se identificar, também, que o alcance de uma cadeia produtiva mais sustentável pode ser apoiada por mudanças culturais nas organizações. A adoção de big data e tecnologias digitais auxilia em mudanças que proporcionam às empresas uma cultura orientada a dados, gerando percepções valiosas e otimizando o processo de tomada de decisão. Ao mesmo passo, investimentos em estratégias de circularidade auxiliam na implantação de uma cultura orientada à economia circular. Dessa forma, os dados e as tecnologias digitais adotadas podem, a partir dessa junção cultural, serem implementados de forma direcionada à sustentabilidade. Nesse contexto, práticas que incentivem a educação e a conscientização dos colaboradores sobre tais temas são essenciais para se alcançar essa mudança.

Entretanto, embora os benefícios do uso de big data e tecnologias digitais para a economia circular tenham sido explorados nesse estudo, há, ainda, a necessidade de se avançar em campos teóricos e práticos, uma vez que há desafios e incertezas sobre essa área de estudo. Dessa forma, sugere-se que a formação de equipes de estudo multidisciplinares, com pesquisadores e demais profissionais com conhecimentos em áreas como a designers, engenharia de produção, engenharia de computação, engenharia de software e ciência da computação, juntamente com demais profissionais, como os especializados na

área de sustentabilidade, podem, de forma colaborativa, identificar meios para a resolução dos desafios apresentados nesse estudo. Em adição, uma vez que a economia circular considera a necessidade de colaboração de todas as partes envolvidas no processo de produção e consumo, a união entre universidades, governantes, formuladores de políticas públicas, instituições públicas e privadas podem acelerar essa transição, promovendo benefícios mútuos a todos os envolvidos, impactando positivamente não somente a geração atual, como também as gerações futuras.

Referências

- [Ahmad et al. 2021] Ahmad, W., Rasool, A., Javed, A. R., Baker, T., and Jalil, Z. (2021). Cyber security in iot-based cloud computing: A comprehensive survey. *Electronics*, 11(1):16.
- [Barros et al. 2020] Barros, M. V., Salvador, R., do Prado, G. F., de Francisco, A. C., and Piekarski, C. M. (2020). Circular economy as a driver to sustainable businesses. page 100006. Elsevier.
- [Benyus 1997] Benyus, J. M. (1997). *Biomimicry: Innovation inspired by nature*. Morrow New York, New York: Quill.
- [Brundtland 1987] Brundtland, G. H. (1987). Our common future—call for action. *Environmental Conservation*, 14(4):291–294.
- [Brydges 2021] Brydges, T. (2021). Closing the loop on take, make, waste: Investigating circular economy practices in the swedish fashion industry. volume 293, page 126245. Elsevier.
- [Cattell 2011] Cattell, R. (2011). Scalable sql and nosql data stores. *ACM Sigmod Record*, 39(4):12–27.
- [Chen et al. 2021] Chen, X., Memon, H. A., Wang, Y., Marriam, I., and Tebyetekerwa, M. (2021). Circular economy and sustainability of the clothing and textile industry. *Materials Circular Economy*, 3(1):1–9.
- [Côrte-Real et al. 2019] Côrte-Real, N., Ruivo, P., Oliveira, T., and Popovič, A. (2019). Unlocking the drivers of big data analytics value in firms. *Journal of Business Research*, 97:160–173.
- [Dastjerdi and Buyya 2016] Dastjerdi, A. V. and Buyya, R. (2016). Fog computing: Helping the internet of things realize its potential. *Computer*, 49(8):112–116.
- [De Donno et al. 2019] De Donno, M., Tange, K., and Dragoni, N. (2019). Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog. *IEEE Access*, 7:150936–150948.
- [Deepa et al. 2022] Deepa, N., Pham, Q.-V., Nguyen, D. C., Bhattacharya, S., Prabadevi, B., Gadekallu, T. R., Maddikunta, P. K. R., Fang, F., and Pathirana, P. N. (2022). A survey on blockchain for big data: approaches, opportunities, and future directions. *Future Generation Computer Systems*.

- [Diddi and Yan 2019] Diddi, S. and Yan, R.-N. (2019). Consumer perceptions related to clothing repair and community mending events: A circular economy perspective. volume 11, page 5306. Multidisciplinary Digital Publishing Institute.
- [Dong and Yang 2018] Dong, J. Q. and Yang, C.-H. (2018). Business value of big data analytics: A systems-theoretic approach and empirical test. *Information & Management*, page 103124.
- [Elkington 1997] Elkington, J. (1997). The triple bottom line. *Environmental management: Readings and cases*, 2:49–66.
- [EMF 2012] EMF (2012). Towards the circular economy: Economic and business rationale for an accelerated transition. <https://ellenmacarthurfoundation.org/>. [Online; Acessado em 13-Abril-2021].
- [EMF 2014] EMF (2014). Towards the circular economy: Accelerating the scale-up across global supply chains. Ellen MacArthur Foundation (EMF) Isle of Wight, UK.
- [EMF 2015] EMF (2015). Towards a circular economy: Business rationale for an accelerated transition. Ellen MacArthur Foundation (EMF) Isle of Wight, UK.
- [EMF 2021] EMF (2021). Circulytics - measuring circularity. <https://www.ellenmacarthurfoundation.org/resources/apply/circulytics-measuring-circularity>. [Online; Acessado em 23-junho-2021].
- [Ernst et al. 2005] Ernst, C., Ferrer, A. H., and Zult, D. (2005). The end of the multi-fibre arrangement and its implication for trade and employment. *ILO Employment Strategy Paper*, 16.
- [Ferrell 2021] Ferrell, O. (2021). Addressing socio-ecological issues in marketing: environmental, social and governance (esg). *AMS Review*, 11(1):140–144.
- [Fuente et al. 2017] Fuente, J. A., García-Sánchez, I. M., and Lozano, M. B. (2017). The role of the board of directors in the adoption of gri guidelines for the disclosure of csr information. *Journal of Cleaner Production*, 141:737–750.
- [Gandomi and Haider 2015] Gandomi, A. and Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. volume 35, pages 137–144. Elsevier.
- [Garcia-Torres et al. 2021] Garcia-Torres, S., Rey-Garcia, M., Sáenz, J., and Seuring-Stella, S. (2021). Traceability and transparency for sustainable fashion-apparel supply chains. Emerald Publishing Limited.
- [Gartner 2022] Gartner (2022). Gartner glossary. <https://www.gartner.com/en/information-technology/glossary/big-data>. [Online; Acessado em 25-Maio-2022].
- [Geissdoerfer et al. 2018] Geissdoerfer, M., Morioka, S. N., de Carvalho, M. M., and Evans, S. (2018). Business models and supply chains for the circular economy. *Journal of cleaner production*, 190:712–721.

- [Geissdoerfer et al. 2017] Geissdoerfer, M., Savaget, P., Bocken, N. M., and Hultink, E. J. (2017). The circular economy—a new sustainability paradigm? *Journal of cleaner production*, 143:757–768.
- [Goldman et al. 2012] Goldman, A., Kon, F., Junior, F. P., Polato, I., and Pereira, R. d. F. (2012). *Apache Hadoop: conceitos teóricos e práticos, evolução e novas possibilidades*, chapter 3, pages 88–136. XXXI Jornadas de Atualizações em Informática (JAI). Sociedade Brasileira de Computação (SBC).
- [Gonçalves and Silva 2021] Gonçalves, A. and Silva, C. (2021). Looking for sustainability scoring in apparel: A review on environmental footprint, social impacts and transparency. volume 14, page 3032. Multidisciplinary Digital Publishing Institute.
- [Graedel 1996] Graedel, T. E. (1996). On the concept of industrial ecology. *Annual Review of Energy and the Environment*, 21(1):69–98.
- [GRI 2021] GRI (2021). Global reporting initiative—sustainability reporting guidelines. <https://www.globalreporting.org/>. [Online; Acessado em 26-Julho-2021].
- [Gupta et al. 2022] Gupta, R., Kushwaha, A., Dave, D., and Mahanta, N. R. (2022). Waste management in fashion and textile industry: Recent advances and trends, life-cycle assessment, and circular economy. *Emerging Trends to Approaching Zero Waste*, pages 215–242.
- [Gupta et al. 2019] Gupta, S., Chen, H., Hazen, B. T., Kaur, S., and Gonzalez, E. D. S. (2019). Circular economy and big data analytics: A stakeholder perspective. *Technological Forecasting and Social Change*, 144:466–474.
- [Haber and Stornetta 1990] Haber, S. and Stornetta, W. S. (1990). How to time-stamp a digital document. In *Conference on the Theory and Application of Cryptography*, pages 437–455. Springer.
- [Härting et al. 2018] Härting, R.-C., Reichstein, C., and Schad, M. (2018). Potentials of digital business models—empirical investigation of data driven impacts in industry. *Procedia Computer Science*, 126:1495–1506.
- [Hausberg et al. 2018] Hausberg, J., Liere-Netheler, K., Packmohr, S., Pakura, S., and Vogelsang, K. (2018). Digital transformation in business research: A systematic literature review and analysis. *DRUID18, Copenhagen Business School, Copenhagen, Denmark*.
- [Hilbert 2022] Hilbert, M. (2022). Digital technology and social change: the digital transformation of society from a historical perspective. *Dialogues in clinical neuroscience*.
- [Huq et al. 2014] Huq, F. A., Stevenson, M., and Zorzini, M. (2014). Social sustainability in developing country suppliers: An exploratory study in the ready made garments industry of bangladesh. *International Journal of Operations & Production Management*.

- [IPCC 2021] IPCC (2021). Climate change 2021: The physical science basis. contribution of working group i to the sixth assessment report of the intergovernmental panel on climate change. *IPCC*.
- [Jestratijevic et al. 2021] Jestratijevic, I., Uanhoro, J. O., and Creighton, R. (2021). To disclose or not to disclose? fashion brands' strategies for transparency in sustainability reporting. Emerald Publishing Limited.
- [Jiang et al. 2014] Jiang, L., Da Xu, L., Cai, H., Jiang, Z., Bu, F., and Xu, B. (2014). An iot-oriented data storage framework in cloud computing platform. *IEEE Transactions on Industrial Informatics*, 10(2):1443–1451.
- [Jo 2021] Jo, T. (2021). *Machine Learning Foundations: Supervised, Unsupervised, and Advanced Learning*. Springer Nature.
- [Khan et al. 2021] Khan, D., Jung, L. T., and Hashmani, M. A. (2021). Systematic literature review of challenges in blockchain scalability. *Applied Sciences*, 11(20):9372.
- [Khan et al. 2019] Khan, N., Naim, A., Hussain, M. R., Naveed, Q. N., Ahmad, N., and Qamar, S. (2019). The 51 v's of big data: survey, technologies, characteristics, opportunities, issues and challenges. In *Proceedings of the international conference on omni-layer intelligent systems*, pages 19–24.
- [Khan et al. 2022] Khan, S. A. R., Piprani, A. Z., and Yu, Z. (2022). Digital technology and circular economy practices: future of supply chains. *Operations Management Research*, pages 1–13.
- [Kirchherr et al. 2017] Kirchherr, J., Reike, D., and Hekkert, M. (2017). Conceptualizing the circular economy: An analysis of 114 definitions. *Resources, Conservation and Recycling*, 127:221–232.
- [Kostoska and Kocarev 2019] Kostoska, O. and Kocarev, L. (2019). A novel ict framework for sustainable development goals. *Sustainability*, 11(7):1961.
- [Koszewska 2019] Koszewska, M. (2019). Circular economy in textiles and fashion—the role of a consumer. In *Circular Economy in Textiles and Apparel*, pages 183–206. Elsevier.
- [Laney et al. 2001] Laney, D. et al. (2001). 3d data management: Controlling data volume, velocity and variety. *META group research note*, 6(70):1.
- [Lee 2017] Lee, I. (2017). Big data: Dimensions, evolution, impacts, and challenges. volume 60, pages 293–303. Elsevier.
- [Lehrer et al. 2018] Lehrer, C., Wieneke, A., vom Brocke, J., Jung, R., and Seidel, S. (2018). How big data analytics enables service innovation: materiality, affordance, and the individualization of service. *Journal of Management Information Systems*, 35(2):424–460.

- [Li and Leonas 2021] Li, J. and Leonas, K. K. (2021). Sustainability topic trends in the textile and apparel industry: a text mining-based magazine article analysis. *Journal of Fashion Marketing and Management: An International Journal*.
- [Li et al. 2015] Li, S., Xu, L. D., and Zhao, S. (2015). The internet of things: a survey. *Information systems frontiers*, 17(2):243–259.
- [Liu et al. 2011] Liu, F., Tong, J., Mao, J., Bohn, R., Messina, J., Badger, L., Leaf, D., et al. (2011). Nist cloud computing reference architecture. *NIST special publication*, 500(2011):1–28.
- [Liu et al. 2021] Liu, Z., Liu, J., and Osmani, M. (2021). Integration of digital economy and circular economy: Current status and future directions. *Sustainability*, 13(13):7217.
- [Lu 2019] Lu, Y. (2019). The blockchain: State-of-the-art and research challenges. *Journal of Industrial Information Integration*, 15:80–90.
- [Marquesone 2016] Marquesone, R. d. F. P. (2016). *Big Data: Técnicas e tecnologias para extração de valor dos dados*. Casa do Código, São Paulo.
- [Marquesone and Carvalho 2022a] Marquesone, R. d. F. P. and Carvalho, T. C. M. B. (2022a). *Modelo de implementação de big data como apoio na transição para a economia circular na indústria têxtil*. PhD thesis.
- [Marquesone and Carvalho 2022b] Marquesone, R. d. F. P. and Carvalho, T. C. M. d. B. (2022b). Examining the nexus between the vs of big data and the sustainable challenges in the textile industry. *Sustainability*, 14(8):4638.
- [Martínez-Caro et al. 2020] Martínez-Caro, E., Cegarra-Navarro, J. G., and Alfonso-Ruiz, F. J. (2020). Digital technologies and firm performance: The role of digital organisational culture. *Technological Forecasting and Social Change*, 154:119962.
- [Matlin et al. 2020] Matlin, S. A., Mehta, G., Hopf, H., Krief, A., Keßler, L., and Kümmerer, K. (2020). Material circularity and the role of the chemical sciences as a key enabler of a sustainable post-trash age. *Sustainable Chemistry and Pharmacy*, 17:100312.
- [Mattila 2016] Mattila, J. (2016). The blockchain phenomenon. *Berkeley Roundtable of the International Economy*, 16.
- [Mazzei and Noble 2019] Mazzei, M. J. and Noble, D. (2019). Big data and strategy: Theoretical foundations and new opportunities. In *Strategy and Behaviors in the Digital Economy*. IntechOpen.
- [McDonough and Braungart 2010] McDonough, W. and Braungart, M. (2010). *Cradle to cradle: Remaking the way we make things*. North point press.

- [McKinsey 2020] McKinsey (2020). Fashion on climate: How the fashion industry can urgently act to reduce its green house gas emission. <https://www.mckinsey.com/industries/retail/our-insights/fashion-on-climate>. [Online; Acessado em 13-Abril-2021].
- [Mikalef et al. 2019] Mikalef, P., Krogstie, J., Pappas, I. O., and Pavlou, P. (2019). Exploring the relationship between big data analytics capability and competitive performance: The mediating roles of dynamic and operational capabilities. Elsevier.
- [Modgil et al. 2021] Modgil, S., Gupta, S., Sivarajah, U., and Bhushan, B. (2021). Big data-enabled large-scale group decision making for circular economy: An emerging market context. volume 166, page 120607. Elsevier.
- [Moorhouse 2020] Moorhouse, D. (2020). Making fashion sustainable: Waste and collective responsibility. *One Earth*, 3(1):17–19.
- [Mungcharoen et al. 2021] Mungcharoen, T., Varabuntoonvit, V., and Poolsawad, N. (2021). *Life Cycle Greenhouse Gas Emissions for Circular Economy*. Springer.
- [Nakamoto 2008] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, page 21260.
- [ONU 2021a] ONU (2021a). Objetivo 12. assegurar padrões de produção e de consumo sustentáveis. <https://brasil.un.org/pt-br/sdgs/12>. [Online; Acessado em 13-Abril-2021].
- [ONU 2021b] ONU (2021b). Objetivos de desenvolvimento sustentável. <https://brasil.un.org/pt-br/sdgs>. [Online; Acessado em 13-Abril-2021].
- [Potting et al. 2017] Potting, J., Hekkert, M., Worrell, E., and Hanemaaijer, A. (2017). *Circular economy: measuring innovation in the product chain*. Number 2544. PBL publishers.
- [Rajasekaran et al. 2022] Rajasekaran, A. S., Azees, M., and Al-Turjman, F. (2022). A comprehensive survey on blockchain technology. *Sustainable Energy Technologies and Assessments*, 52:102039.
- [Ramzan et al. 2019] Ramzan, S., Bajwa, I. S., Ramzan, B., and Anwar, W. (2019). Intelligent data engineering for migration to nosql based secure environments. *IEEE Access*, 7:69042–69057.
- [Rani and Saha 2021] Rani, H. and Saha, G. (2021). Organizations and standards related to textile and fashion waste management and sustainability. In *Waste Management in the Fashion and Textile Industries*, pages 173–196. Elsevier.
- [Ranjan 2019] Ranjan, J. (2019). The 10 vs of big data framework in the context of 5 industry verticals. *Productivity*, 59(4).
- [Rashidi et al. 2019] Rashidi, H. H., Tran, N. K., Betts, E. V., Howell, L. P., and Green, R. (2019). Artificial intelligence and machine learning in pathology: the present landscape of supervised methods. *Academic pathology*, 6:2374289519873088.

- [Rogers and Hudson 2011] Rogers, K. and Hudson, B. (2011). The triple bottom line: The synergies of transformative perceptions and practices for sustainability, with barclay hudson, od practitioner (fall 2011). *OD Practitioner*.
- [Russel et al. 2013] Russel, S., Norvig, P., et al. (2013). *Artificial intelligence: a modern approach*. Pearson Education Limited London.
- [SAC 2021] SAC (2021). The higg index. <https://portal.higg.org/>. [Online; Acessado em 28-Junho-2021].
- [Saha et al. 2021] Saha, K., Dey, P. K., and Papagiannaki, E. (2021). Implementing circular economy in the textile and clothing industry. *Business Strategy and the Environment*, 30(4):1497–1530.
- [Sandvik and Stubbs 2019] Sandvik, I. M. and Stubbs, W. (2019). Circular fashion supply chain through textile-to-textile recycling. *Journal of Fashion Marketing and Management: An International Journal*.
- [Sawadogo and Darmont 2021] Sawadogo, P. and Darmont, J. (2021). On data lake architectures and metadata management. *Journal of Intelligent Information Systems*, 56(1):97–120.
- [Sun 2008] Sun, S. (2008). Organizational culture and its themes. *International Journal of Business and Management*, 3(12):137–141.
- [Thorisdottir and Johannsdottir 2019] Thorisdottir, T. S. and Johannsdottir, L. (2019). Sustainability within fashion business models: A systematic literature review. *Sustainability*, 11(8):2233.
- [UNFCC 2018] UNFCC (2018). Earth's annual resources budget consumed in just 7 months. <https://unfccc.int/news/earth-s-annual-resources-budget-consumed-in-just-7-months>. [Online; Acessado em 13-Abril-2021].
- [Ütebay et al. 2020] Ütebay, B., Çelik, P., and Çay, A. (2020). Textile wastes: Status and perspectives. In *Waste in Textile and Leather Sectors*. IntechOpen.
- [Vial 2019] Vial, G. (2019). Understanding digital transformation: A review and a research agenda. *The Journal of Strategic Information Systems*.
- [WIKIRATE 2021] WIKIRATE (2021). Wiki rate. <https://wikirate.org/>. [Online; Acessado em 28-Junho-2021].
- [Yuan and Wang 2018] Yuan, Y. and Wang, F.-Y. (2018). Blockchain and cryptocurrencies: Model, techniques, and applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(9):1421–1428.
- [Zaharia et al. 2016] Zaharia, M., Xin, R. S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M. J., et al. (2016). Apache spark: a unified engine for big data processing. *Communications of the ACM*, 59(11):56–65.

Capítulo

4

Visitando na teoria e na prática o Cartesi RollUps: para além das limitações da Blockchain, uma solução de futuro para aplicativos descentralizados

Antonio A. de A. Rocha, Arthur A. Vianna, Bruno T. Gondim,
Eduardo B. Loivos, Rayan G. O. J. Lima

Abstract

Polarizations, wars and the risk of arbitrary censorship are consequences of the centralization of decision-making in the hands of a few. Technology exists to solve human needs, such as civil liberties. In this sense, web 3.0 seeks in decentralization the neutralization of arbitrary actions. Within this perspective, this work proposes to expose one of the most promising technologies - Cartesi Rollups - capable of elevating Blockchain technology to the status of standard for future internet applications, minimizing its scalability limitations. Practical examples of how to develop and implement such decentralized applications will also be exposed.

Keywords: *Blockchain, Sharding, scaling, decentralized ledger, Cartesi, rollups.*

Resumo

Polarizações, guerras e riscos de censuras arbitrárias são consequências da centralização da decisão nas mãos de poucos. A tecnologia existe para resolver necessidades humanas, como a liberdade civil. Neste sentido, a web 3.0 busca na descentralização a neutralização de ações arbitrárias. Dentro desta perspectiva, este trabalho propõe expor uma das tecnologias mais promissoras - o Cartesi Rollups - capaz de alçar a tecnologia Blockchain ao status de padrão das aplicações na internet do futuro, minimizando suas limitações de escalabilidade. Também serão expostos exemplos práticos de como desenvolver e implementar tais aplicações descentralizadas.

Palavras-chaves: *Blockchain, Sharding, escalabilidade, razão descentralizada, Cartesi, rollups.*

4.1. Introdução

A expressão “*In Blockchain We Trust*” vem sendo utilizada para ressaltar a confiança na segurança dessa solução tecnológica. Isto porque, nela as transações entre indivíduos e a transferência de valores é garantida por meio de algoritmos matemáticos e criptográficos. A Blockchain é uma tecnologia de núcleo que possibilita que grandes grupos de pessoas cheguem a um acordo (também definido como, consenso) e registrem transações permanentes, sem uma autoridade central.

O termo (Bloco = block + chain = cadeia) tem origem no seu funcionamento, onde cada bloco validado é criptograficamente selado ao bloco anterior, formando uma cadeia de blocos cada vez maior, diretamente relacionado na construção de uma economia digital justa, inclusiva, segura e democrática. Por isso, a Blockchain é descrita como uma máquina de confiança, não é apenas um *ledger* distribuído em larga escala, mas também como uma trilha de auditoria imutável. Nela, cada bloco é incorporado em todos os seguintes, impossibilitando a alteração da história de seu conteúdo.

A tecnologia Blockchain também é, portanto, um banco de dados distribuídos, sendo praticamente invulnerável a falhas e adulterações. Suas múltiplas utilidades descolam-se da tecnologia de criptomoedas Bitcoin, para a qual foi criada. Antes do desenvolvimento da tecnologia Blockchain, os registros contábeis eram mantidos em bancos de dados centralizados e não públicos. As pessoas precisavam confiar na idoneidade do banco de dados para ter certeza de que não haveria nenhuma alteração nos registros (saldos e transações da conta, por exemplo). Com a Blockchain, os dados são distribuídos entre todos os participantes, com total transparência e descentralização. Logo, torna-se desnecessário confiar em uma terceira parte para que os dados contábeis sejam registrados corretamente e não haja perigo de fraudes. Podemos concluir que o modelo de solução Blockchain assemelha-se a um “livro-razão”, ou seja, uma base de informações com entrada de diversos dados e transações. Todas as informações imputadas e contidas na ferramenta são compartilhadas entre vários usuários.

O processamento desta base de dados é feito em blocos, “de tempos em tempos”, criando um código de verificação a cada bloco processado. Estes códigos de verificação são criados com base nos blocos processados anteriormente, fazendo com que a Blockchain seja uma solução de alta confiabilidade, pois, uma vez adulterado um bloco, isso impactará nos demais blocos processados. Por isso, tecnologias Blockchain encontram aplicações em soluções que existe desconfiança mútua pelas partes envolvidas, são exemplos delas: Estabelecimento de contratos; Registro de propriedade intelectual; Estabelecimento de Identidade digital; Prontuário médico; Cartórios digitais; Operações cambiais imediatas; Sistema de voto digital; e, Auditabilidade [Revoredo 2019].

4.2. Blockchain: conceitos fundamentais

A Blockchain oferece propriedades que proveem benefícios às aplicações e sistemas baseados nesta tecnologia. Conforme definido em [Rebello et al. 2019], as principais propriedades da Blockchain são:

- **Descentralização.** A Blockchain é executada de maneira distribuída, sem a necessidade de um intermediário confiável para a troca de ativos, através do estabeleci-

mento de consenso entre todos os participantes da rede;

- **Imutabilidade.** Os dados armazenados em uma corrente de blocos são imutáveis. Não é possível modificar ou recriar qualquer dado incluído na corrente de blocos. Toda atualização na corrente de blocos é realizada de forma incremental;
- **Irrefutabilidade.** Os dados são armazenados na corrente de blocos em forma de transações assinadas, que não podem ser alteradas devido à propriedade de imutabilidade da corrente de blocos. Portanto, o emissor de uma transação jamais pode negar sua existência;
- **Transparência.** Todos os dados armazenados na Blockchain são acessíveis por todos os participante da rede. Permitindo que todos os participantes possam verificar, auditar e rastrear os dados inseridos na corrente de blocos para encontrar possíveis erros ou comportamentos maliciosos;
- **Disponibilidade.** As correntes de blocos são estruturas replicadas em cada participante da rede e, portanto, a disponibilidade do sistema é garantida mesmo sob falhas, devido à redundância de informações;
- **Anonimidade.** Os usuários e nós mineradores de uma Blockchain são identificados por chaves públicas ou identificadores únicos que preservam suas identidades. Ainda, é possível utilizar uma chave pública em cada transação, evitando a rastreabilidade do usuário e conferindo um grau a mais de anonimidade.

4.2.1. Função Hash

A função hash é uma função matemática que tem como entrada uma string de tamanho variável e a mapeia em uma string de tamanho fixo. A conversão tem como objetivo gerar uma identidade única, resistente a colisões. Uma função hash é resistente à colisão, se houver baixa probabilidade de encontrar dois valores de entrada distintos, que possuam um mesmo valor de saída (Figura 4.1). Dado que mapear uma grande quantidade de dados para um universo menor, quanto menor a probabilidade de encontrar hash iguais de forma proposital, melhor a função de hash.

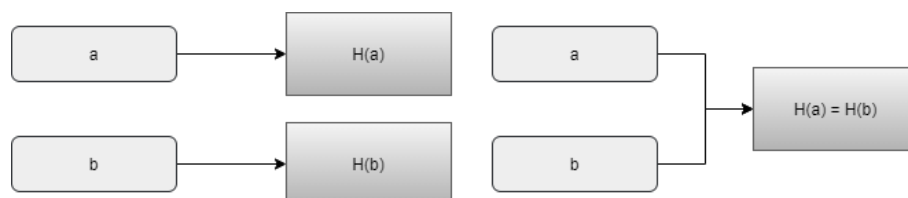


Figura 4.1. Exemplo de colisão entre dois hashes

Uma característica importante do hash é que, uma pequena variação na entrada resulta em uma grande variação na saída. Como pode ser visto na Figura 4.2, mudar uma uma letra de maiúscula para minúscula gera uma saída completamente diferente. Não deve ser possível adicionar conteúdo “a” um texto sem modificar sua saída. Se uma função $H(a)$ resulta a string fixa “y”, encontrar uma string “b” tal que concatenada com “a”

resulte em $H(a \parallel b) = y$, em um curto espaço de tempo, não deve ser possível. Da mesma forma, se uma entrada tem uma parte gerada de forma aleatória, do hash resultante não pode ser escolhido um hash específico. Outra propriedade da função hash é: Dada uma saída, não é possível determinar a entrada que a originou. Em um espectro pequeno ou repetido de entradas, uma vez vista a solução de para uma dessas entradas, passa a ser possível identificá-la através de seu hash. É possível ocultar a entrada usando um nonce, ou um valor secreto, concatenado a entrada original.

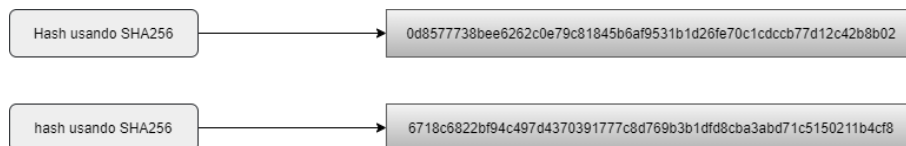


Figura 4.2. Exemplo hash usando SHA256

4.2.2. Estrutura e Cadeia de bloco

A estrutura de bloco consiste, principalmente, em um conjunto de dados. No universo das criptomoedas, os dados de um bloco representam as transações que ocorrem em um determinado período de tempo. Além dos dados, um bloco possui seu índice ou altura, e um campo de nonce. Uma vez formado um bloco, uma função de hash é usada para gerar uma assinatura para o mesmo, garantindo que os dados registrados nele não serão alterados. Uma função de hash comum para esta finalidade é a SHA de 256 bits usada pelo Bitcoin.

Ao criar um bloco e assiná-lo, os dados são verificados e apenas as transações válidas entram em um bloco. Para dificultar que um usuário mal-intencionado valide um bloco com dados forjados, é necessário um esforço para assinar o bloco. As formas mais comuns de esforço usadas são *Proof of Work (PoW)* e *Proof of Stake (PoS)*, mais detalhadas à frente.

A função hash garante que qualquer alteração em um bloco seja facilmente detectada. Para garantir a integridade dos dados, é necessário que um bloco esteja ligado ao próximo através do hash. Assim, para alterar um bloco, deve-se alterar todos os blocos posteriores. Os blocos são conectados através de um campo adicional contendo o hash do bloco anterior. O primeiro bloco, conhecido como bloco gênese, possui esse campo com o valor 0 em todos os caracteres.

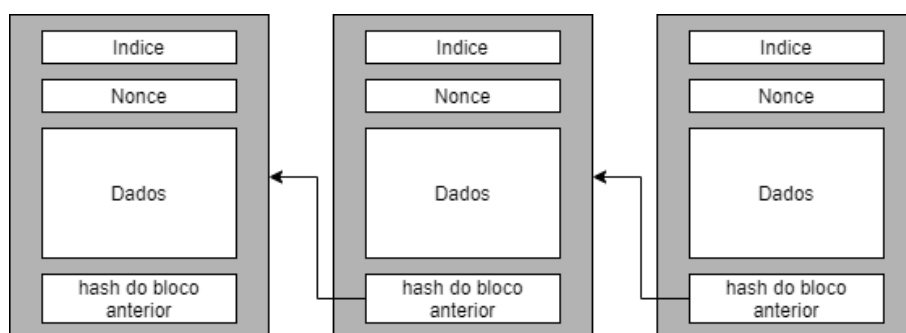


Figura 4.3. Modelo de uma Cadeia de Blocos (Blockchain)

Todos os usuários possuem uma cópia da Blockchain, e sempre que um minerador confirma um bloco ele transmite sua solução na rede. Devido a latência ou a ação de um nó mal intencionado, duas soluções podem ser aceitas em partes diferentes da rede gerando um *fork* na Blockchain. Uma vez identificado o *fork*, os nós consideram a cadeia mais longa como a verdadeira. Logo, para falsificar os dados de um bloco da cadeia, um usuário malicioso precisa, no caso de prova de trabalho, de poder computacional suficiente para tornar seu *fork* mais longo que a Blockchain verdadeira.

4.2.3. Consenso

Plataformas Blockchain usam consenso descentralizado para manter a consistência em uma máquina de estado distribuída. Esta pode ser usada para realizar pagamentos completamente descentralizados ou mesmo cálculos de Turing completos, tornando realidade a criação de aplicações descentralizadas. É papel do consenso em sistemas de Blockchain garantir que todos os nós confiáveis em uma rede Blockchain executem as mesmas atualizações de estado na mesma ordem [Muratov et al. 2018].

Há dois tipos bem comuns de algoritmo de consenso, os baseados em prova e os em voto. No conceito básico de algoritmo de consenso baseado em prova, entre os muitos nós que se juntam à rede, o nó que executa a prova terá o direito de acrescentar um novo bloco à corrente e receber a recompensa. Os principais algoritmos com base em prova são *Proof of Work*, *Proof of Stake* e alguma variação desses [Pahlajani et al. 2019]. Já os algoritmos de consenso baseado em votação, além de manter o livro-razão, todos os nós da rede teriam que verificar juntos as transações ou blocos. Eles se comunicam uns com os outros, antes de decidir anexar ou não os blocos propostos à cadeia. Dentro de quase todas essas variantes, os algoritmos requerem que um número mínimo de nós aceitem o mesmo bloco proposto para anexá-lo à cadeia [Pahlajani et al. 2019]. Alguns desses modos de consenso são descritos a seguir:

Proof of Work (PoW): O uso de um sistema como PoW impede que usuário utilize seu poder computacional para gerar um spam de determinada informação. No cenário da Blockchain, o PoW impede que diversos blocos sejam criados sequencialmente pelo mesmo usuário. O PoW envolve a busca de um valor de nonce que quando processado por um hash, como com SHA-256, o *output* começa com um determinado número de bits zero. O trabalho médio necessário é exponencial, de acordo com número de bits zero necessários, e pode ser verificado executando um único hash. [Nakamoto 2008]

Proof of Stake (PoS): Normalmente pergunta-se, se nós devem manter o consumo de energia para ter uma criptomoeda descentralizada. Portanto, demonstrar que a segurança de criptomoedas ponto a ponto não precisa depender de consumo de energia, é um marco importante tanto teórico quanto tecnologicamente [King and Nadal 2012]. O PoS aproveita o fato de que a posse de moeda é proporcional ao tempo que um nó participa da rede. Desta forma demonstra a sua confiabilidade e assegura a honestidade do mesmo.

Proof of Elapsed Time (PoET): O consenso PoET atua como uma loteria, na qual, cada nó deve esperar um período de tempo randômico, e o primeiro a concluir este tempo de espera ganha o bloco. O próprio nó deve gerar um valor de tempo pelo qual ele irá ficar em *sleep mode*, o primeiro nó a despertar, ou seja, aquele que tiver o menor período de tempo de espera, irá fazer o commit do bloco para a Blockchain e realizar o brod-

cast das informações relevantes [Jake Frankenfield 2020]. Geralmente, alguma prova de que ele não adulterou o processo é necessária. Para garantir que o nó não gere valores de tempo pequenos para sempre ganhar o bloco, os nós que participam deste consenso precisam executar esta geração de valores em um *TEE (Trusted Execution Environment)*. TEEs são ambientes de execução que têm uma superfície de ataque extremamente pequena e são equipados com procedimentos de verificação criptográfica que podem fornecer atestados verificáveis externamente e evidências de adulteração [Corso 2019].

Consenso baseado em voto: Nesse tipo de consenso, todos os nós mineradores devem votar pela aprovação ou rejeição de um bloco, e atingir um consenso antes de adicionar o novo bloco à corrente. Como consequência, todos os nós mineradores devem ser conhecidos e identificados. O protocolo de consenso garante que a adição de um bloco à corrente ocorre de forma sincronizada para todas as réplicas. Isto é, as réplicas adicionam a mesma sequência de blocos na corrente. Dentre os modelos baseados em voto, os tolerantes a falhas bizantinas (Byzantine Fault Tolerant - BFT) são particularmente interessantes. A ideia principal dos protocolos BFT é eleger um líder por uma rodada, que determina e propõe um novo bloco aos participantes do consenso. Os protocolos BFT promovem uma camada de confiança a mais em comparação a protocolos tolerantes apenas a falhas por parada, pois toleram comportamento malicioso na rede. No entanto, geralmente necessitam de mais réplicas, quando comparado com os protocolos que toleram falhas apenas por parada, para tolerar a mesma quantidade de falhas [Rebello et al. 2019].

4.2.4. Bitcoin x Ethereum

Como o protocolo Bitcoin é “Open Source”, qualquer um poderia pegar seu protocolo, bifurcar (modificar o código) e iniciar sua própria versão de dinheiro eletrônico. Essa qualidade da Blockchain Bitcoin possui um código aberto que contribuiu para que, ao longo dos anos, o protocolo Bitcoin fosse modificado centenas de vezes para criar versões alternativas do Bitcoin que são mais rápidas ou mais anônimas. Percebeu-se que o protocolo blockchain subjacente possibilitava que pessoas desconhecidas, ou que não confiavam entre si, realizassem qualquer tipo de transação de valor, e não apenas dinheiro, sem quaisquer intermediários.

Começam a surgir, então, projetos que buscavam usar a tecnologia Blockchain para transferências, sem intermediários, de outros tipos de valor. Também ganhou corpo a ideia de se afastar da blockchains de propósito único, para criar um protocolo onde qualquer tipo de transação, sem validadores tradicionais de confiança, fosse possível. Então, ao perceber que as adaptações da Blockchain Bitcoin não eram satisfatoriamente eficientes nem flexíveis, Vitalik Buterin introduziu a ideia de dissociar as funcionalidades blockchain e iniciou o projeto da Blockchain Ethereum [Wood 2014] em 2014.

Ao contrário da Blockchain Bitcoin, que é uma Blockchain de propósito único com um único contrato inteligente, a Blockchain Ethereum é projetada como uma rede de computadores descentralizada na qual qualquer tipo de contrato inteligente pode ser programado, permitindo qualquer tipo de troca direta de valor. Outras Blockchains surgiram como solução, pois o surgimento da Ethereum inspirou projetos de Blockchain mais recente (como NEO, EOS, CARDANO, CHAINLINK, QTUM, STELLAR, GOCHAIN, entre outros. Além do aspecto tecnológico, fatores técnicos, econômicos e legais também

serão relevantes para avaliar a viabilidade de uma Blockchain [Revoredo 2019].

4.2.5. Um novo mundo para aplicações da Blockchain

Acredita-se que, as máquinas serão os clientes do futuro, presumindo que, mais cedo ou mais tarde, terão carteiras integradas. As primeiras tentativas nesse sentido já foram feitas e com o Ethereum, em que é possível implantar carteiras com contratos inteligentes. Embora a aplicação como moeda e sistema financeiro sem intermediários seja a aplicação mais famosa da tecnologia de Blockchain, várias outras possibilidades e aplicações podem ser vislumbradas.

Uma Organização Autônoma Descentralizada, ou DAO, é uma organização ou empresa teórica operada por código em vez de pessoas. Os DAOs criam uma maneira de as organizações ou empresas serem estruturadas de forma menos hierárquica, argumentam os defensores, com os investidores direcionando diretamente a direção das empresas, em oposição aos líderes designados. Os defensores do DAO acreditam que o Ethereum pode dar vida a essa ideia futurista. Ethereum é a segunda maior criptomoeda por capitalização de mercado e é a maior plataforma para usar a tecnologia por trás da criptomoeda - Blockchain - para usos além do dinheiro. A ideia é que, se o Bitcoin pode eliminar os intermediários nos pagamentos online, a mesma tecnologia ou uma tecnologia comparável pode fazer o mesmo pelos intermediários nas empresas? E se organizações inteiras pudessem existir sem um líder central ou CEO comandando o show?

Com isso, será possível criar múltiplas cópias de segurança: Redundância de informações é algo fundamental para evitar perdas de dados, e nada tem mais backups do que algo controlado por um Blockchain. Afinal, cada um dos mineradores e até mesmos outros membros da rede (como o caso dos fullnodes do Bitcoin) possuem uma cópia completa, atualizada e integral de todos os blocos e, portanto, todas as informações registradas por ele, tais como:

- Registros virtualmente inalteráveis: o hash que valida um bloco é formado pelas informações registradas em um bloco e hash do bloco anterior, que foi gerado pelos dados dele e seu antecessor, e por aí vai. Assim sendo, as informações em um Blockchain são incrementais e acumulativas, armazenadas de forma a não ser alteradas ou apagadas.
- A autenticidade de documentos a garantir a autenticidade de documentos é algo custoso, especialmente no Brasil, e ser caro não garante que o processo é à prova de falhas ou fraudes, pois sempre que houver um fator humano que possa ser corrompido e subverter o sistema, a fraude será possível. A startup brasileira OriginalMy surgiu com uma proposta: registrar documentos em um Blockchain, beneficiando-se de, pelo menos, duas características importantes: o fato de que o registro do documento não possa ser alterado ou removido e a transparência que um Blockchain público provê, permitindo a verificação deste registro sem burocracias.

Também será possível uma maior proteção dos direitos autorais, caso um sistema notarial de algum país fosse implementado integralmente em um único Blockchain. Além

de mitigar fraudes por adulteração ou remoção de informações, todos os documentos poderiam ser verificados e confrontados, evitando a fraude do “vidente que registra sua previsão em cartório”: algumas pessoas fazem dezenas de previsões aleatórias, como “prever” celebridades mortas em acidentes aéreos, por exemplo. Ao registrar tais previsões em dezenas de cartórios diferentes (cada previsão em um cartório diferente), quando o previsto se torna um fato, basta apresentar a previsão que “acertou” o acontecimento, desprezando todas as demais. Algo semelhante aconteceu na Copa do Mundo de 2014, quando uma conta de Twitter chamada “@fraudefifa” quis provar que a FIFA manipulava os resultados da competição. Curiosa, no entanto, foi a maneira pela qual a conta decidiu “provar” sua tese, realizando postagens com todas as possibilidades de confrontos e vencedores, dias antes das partidas. Quando dois times realmente jogavam e um destes ganhava, bastava remover do Twitter todas as postagens desfavoráveis.

Surgiram outras iniciativas de autenticação de documentos na Internet, como é o caso do site LexisNexis, que também registra os documentos de maneira a comprovar sua autoria futura. Nesse contexto, a gigante Kodak decidiu não investir na tecnologia em razão da grande receita que a empresa adquiria na revelação de filmes fotográficos tradicionais. A ideia é criar uma plataforma de gerenciamento de direitos de imagem da qual os fotógrafos possam registrar seus trabalhos em um Blockchain, que impedirá que tal registro seja alterado ou apagado. Desta maneira, o registro da foto em um Blockchain poderia comprovar um eventual plágio fotográfico, permitindo identificar o autor da obra com facilidade.

Existem estudos de Implementação do Smart Governance, votação 2.0. Além de garantir a privacidade, o contrato traz transparência ao processo de apuração pois, ao verificar as condições para as quais os votos foram apurados, é possível garantir que 1 pessoa = 1 voto. Ao utilizar um Blockchain como infraestrutura para que o sistema de votação seja realizado, reduzem - se drasticamente quaisquer fraudes que possam hoje ser realizadas em sistemas eleitorais tradicionais.

As Blockchains e smart contracts são tecnologias em ascensão. Enquanto os ativos digitais são usados por uma parcela muito pequena da população mundial, a adoção da Blockchain é inicial em outros campos e ainda certamente encontrarão inúmeros casos de uso da tecnologia ainda desconhecidos ou com desenvolvimento ainda muito incipientes [Eichmann 2018].

4.3. Aplicações Descentralizadas

A criação de contratos inteligentes trouxe consigo uma nova proposta de arquitetura para aplicações, chamada de DApp. A sigla DApp vem do inglês “Decentralized Application”, que tem como tradução “Aplicação Descentralizada”. Ao contrário do que pode parecer a princípio, o termo “descentralizada” presente na sigla DApp não diz respeito a um back-end ou processamento descentralizado. Afinal, uma aplicação não precisa rodar na Blockchain para ser descentralizada. Há algum tempo já existem bancos de dados distribuídos. Nesse caso, a palavra descentralizada remete a ideia de que a aplicação não possui uma única entidade que controla o seu funcionamento. Não apenas isso, para que uma aplicação seja considerada um DApp, ela deve atender os seguintes critérios, segundo [Johnston et al. 2014]:

- A aplicação deve ser completamente open-source, deve operar de forma autônoma, sem que nenhuma entidade controle a maioria de seus tokens, e seus dados e registros de operação devem ser armazenados criptograficamente em uma cadeia de blocos pública e descentralizada, ou seja, armazenada em uma Blockchain pública;
- A aplicação deve gerar tokens de acordo com um algoritmo padrão ou conjunto de critérios e possivelmente distribuir alguns ou todos os tokens no início de sua operação. Esses tokens devem ser necessários para o uso da aplicação e qualquer contribuição dos usuários deve ser recompensada com pagamento de tokens da aplicação;
- A aplicação pode adaptar o seu protocolo em resposta às melhorias propostas e feedback do mercado, mas todas as mudanças devem ser decididas por consenso da maioria de seus usuários.

Dado que tokens constituem o critério do segundo ponto, é importante termos uma definição para tal. A finalidade de um token é permitir o acesso a uma aplicação. Por exemplo, um indivíduo deve possuir um número de Bitcoins para poder realizar qualquer transação na rede Bitcoin. Os tokens nos DApps não representam nenhum ativo subjacente, não dão direito a um dividendo e nenhum patrimônio é representado por meio deles. Embora o valor de um token no DApp possa aumentar ou diminuir ao longo do tempo, os tokens não são títulos de capital.

Com base na definição apresentada para tokens e nos critérios definidos para que uma aplicação seja considerada um DApp, vemos que o conceito de “descentralizada” é mais amplo do que pode parecer a princípio.

4.3.1. Bitcoin visto como um DApp

Além de ser um sistema de dinheiro eletrônico ponto a ponto, o Bitcoin também é um aplicativo com o qual os usuários podem interagir. Por conta disso e com base nos critérios definidos, nós podemos afirmar que o Bitcoin é um DApp porque:

- O Bitcoin é open-source, nenhuma entidade (governo, empresa ou organização) controla o Bitcoin e todos os registros relacionados ao uso do Bitcoin são abertos.
- O Bitcoin gera seus tokens, os Bitcoins, com um algoritmo predeterminado que não pode ser alterado, e esses tokens são necessários para o funcionamento do Bitcoin. Os mineradores de Bitcoin são recompensados com Bitcoins por suas contribuições na segurança da rede Bitcoin.
- Todas as alterações no Bitcoin devem ser aprovadas por um consenso majoritário de seus usuários por meio do mecanismo de Proof of Work.

4.3.2. Classificação

A forma padrão usada para se classificar um DApp é de acordo com a Blockchain utilizada por ele, uma Blockchain própria ou de outro DApp. A partir dessa forma de classificação, surgem três tipos:

- **Tipo 1:** DApps que possuem a sua própria Blockchain. Bitcoin é o exemplo mais famoso de um DApp tipo I, mas Litecoin e outras “alt-coins” também são do mesmo tipo.
- **Tipo 2:** DApps que utilizam a Blockchain de um DApp do tipo 1. DApps do tipo 2 são protocolos e possuem tokens que são necessários para o seu funcionamento. Um exemplo de DApp desse tipo é o Omni Layer [OmniLayer 2020], que é uma plataforma para troca de assets ou moedas digitais que faz uso da Blockchain do Bitcoin. Dessa forma o Omni Layer busca expandir o Bitcoin criando novas funcionalidades, mas no fundo todas as transações do Omni são transações Bitcoin.
- **Tipo 3:** DApps que utilizam o protocolo de um DApp do tipo 2. DApps do tipo 3 também são protocolos e possuem tokens que são necessários para o seu funcionamento. Um caso seria uma aplicação que faz uso de um DApp tipo 2 para emitir suas moedas, como ocorre com o SAFE Network (SafeCoin) e o Omni.

4.3.3. Smart Contracts

O termo “Smart Contract” foi criado por Nick Szabo em 1996, antes mesmo do surgimento da primeira Blockchain, o Bitcoin, em 2008. Eles não tinham nada a ver com DAOs, ou trocas descentralizadas ou qualquer um desses tipos de construções que as pessoas tendem a pensar quando ouvem o termo. O conceito é na verdade bem mais simples do que os sistemas e aplicações que vem à mente quando pensamos em contratos inteligentes, uma vez que ele é derivado da ideia de contrato em si, que é: um conjunto de promessas acertadas em um “encontro de mentes”, é a forma tradicional de formalizar um relacionamento. Note que, este conceito vai além de relações comerciais, podendo se estender à política e também a relacionamentos pessoais, como o casamento.

A partir do conceito de contrato, os contratos inteligentes são definidos da seguinte forma, de acordo com o criador do termo [Szabo 1996]. “Novas instituições e novas formas de formalizar as relações que compõem essas instituições são agora possibilitadas pela revolução digital. Eu chamo esses novos contratos de inteligentes, porque eles são muito mais funcionais do que seus ancestrais inanimados baseados em papel. Nenhum uso de inteligência artificial está implícito. Um contrato inteligente é um conjunto de promessas, especificadas em formato digital, incluindo protocolos dentro dos quais as partes cumprem essas promessas.”

Dada a definição, ao desenvolver contratos inteligentes é necessário cumprir 4 objetivos para garantir que o contrato é funcional. [Szabo 1996]

- **Observabilidade:** As pessoas (ou coisas) envolvidas no contrato devem ser capazes de ver que a outra parte está cumprindo corretamente os termos do contrato e ser capazes de provar que elas mesmas estão cumprindo corretamente com a outra parte.
- **Verificabilidade:** Todas as partes de um contrato precisam da capacidade de provar ao árbitro escolhido do contrato que ele foi executado corretamente ou que uma parte (ou partes) violou suas obrigações no contrato. O árbitro em questão pode ou não ser uma terceira parte confiável.

- **Privacidade:** O contrato deve ser estruturado da forma mais privada possível. A quantidade de informações privadas sobre o contrato ou as partes envolvidas que são divulgadas além delas para o público ou outros terceiros deve ser mantida no mínimo necessário para a execução do contrato.
- **Aplicabilidade:** Deve haver algum mecanismo para garantir que as coisas sejam executadas corretamente, mesmo no caso de uma ou mais partes violar suas obrigações sob os termos do contrato e, também, o contrato deve ser estruturado de forma a tornar a necessidade de coerção para seu cumprimento muito improvável. Os contratos devem encorajar as partes a cumprir voluntariamente suas obrigações sob os termos.

4.3.4. Bitcoin Smart Contracts

A rede Bitcoin funciona como um árbitro distribuído gigante, impondo a execução adequada de contratos inteligentes sem depender de uma única autoridade central para fazê-lo. Ele fornece um mecanismo para que os contratos sejam observáveis, verificáveis e aplicáveis. A única qualidade de um contrato que ficou aquém historicamente é a privacidade - todos os termos dos contratos inteligentes do Bitcoin são públicos para todos verem. No entanto, preserva as identidades reais daqueles envolvidos em contratos. Em novembro de 2021 houve um soft-fork do Bitcoin onde foi ativado o Taproot, “raiz principal” em tradução livre, é uma melhoria proposta por Gregory Maxwell em 2018, e visa ocultar as cláusulas de um contrato, a menos que seja necessário aplicá-las, para isso faz uso de um protocolo de assinatura chamado MuSig [Maxwell et al. 2019].

Os contratos inteligentes que podem ser desenvolvidos no Bitcoin são escritos na linguagem Script, muitas vezes chamada de Bitcoin Script, que é uma linguagem do paradigma orientada a pilha(stack-oriented ou stack-based). Com isso, essa linguagem além de possuir um paradigma pouco usual os contratos desenvolvidos utilizando-a são pouco expressivos devido à limitações dela.

4.3.5. Ethereum Smart Contracts

No Ethereum, existem dois tipos de contas: contas de usuário e contas de contrato inteligente. As contas de usuário representam participantes, incluindo callers (que chamam funções de contratos inteligentes), deployers (que implantam contratos inteligentes no Ethereum) e miners (cujos nós trabalham para contribuir para o livro-razão). As contas de contrato inteligente representam o contrato inteligente que é um tipo de programa que é salvo e pode ser executado em Blockchains, também chamado de chaincode (código na corrente). Ethereum é a primeira Blockchain que fornece linguagem de programação Turing-completa para desenvolver contratos inteligentes [Wu et al. 2021].

As transações podem ser classificadas em duas dimensões, a do dado enviado ou do iniciador da transação. Na dimensão do dado, temos dois tipos, transferência de Ether, que representa a transferência de uma conta de usuário para outra, e execução de contrato, que representa a chamada de um método do contrato com seu input e Ether para pagar a taxa de execução de tal método. Na dimensão do iniciador, também temos dois tipos, transações externas, que são aquelas iniciadas por contas de usuários, e transações

internas, que são aquelas iniciadas por contas de contratos inteligentes. A Figura 4.4 resume os tipos de contrato.

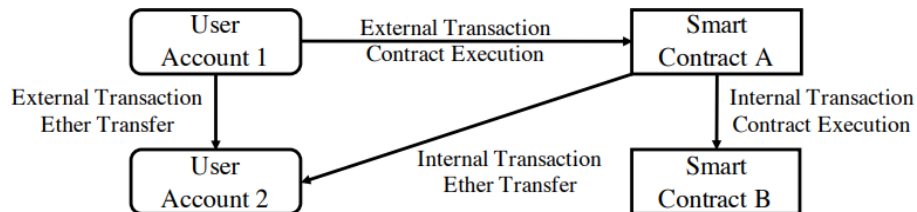


Figura 4.4. Tipos de Contrato no Ethereum [Wu et al. 2021]

O Blockchain Ethereum fornece recursos de computação e armazenamento por meio do mecanismo de contratos inteligentes. Portanto, os DApps que utilizam o Ethereum como Blockchain podem implantar contratos inteligentes para usar tais recursos. Em teoria, todos os processos e dados de um DApp baseado em Blockchain devem ser manipulados e armazenados na Blockchain para pura descentralização. No entanto, devido ao gargalo de desempenho da Blockchain, que será estudado em detalhes na próxima seção, os DApps atuais geralmente implementam apenas partes de sua funcionalidade na Blockchain. Como resultado, três tipos de arquiteturas são adotados pelos DApps Ethereum, tais arquiteturas estão dispostas na Figura 4.5.

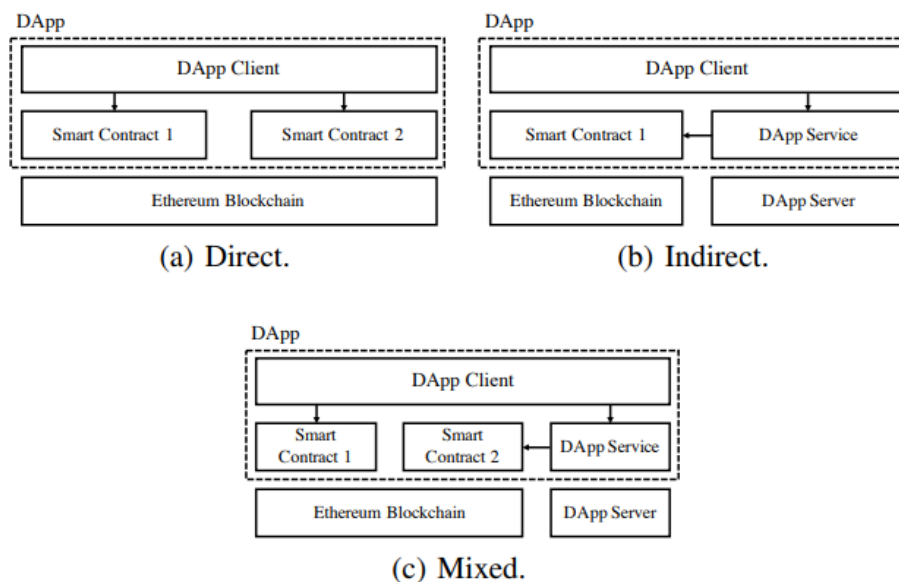


Figura 4.5. Tipos de Arquiteturas Ethereum DApp [Wu et al. 2021]

No Ethereum os contratos são escritos na linguagem Solidity, na qual definimos contratos, que é basicamente uma classe, funções e eventos. Uma vez pronto, o código-fonte do contrato inteligente é compilado em bytecode para ser implantado no Ethereum. Após a implantação, o contrato inteligente receberá um endereço. Qualquer conta pode realizar a implantação de um contrato, inclusive uma conta de contrato pode implantar

outros contratos, chamados de contratos filhos, a implantação feita dessa forma recebe o nome de “factory patterns”.

Todas as transações no Ethereum acarretam em uma taxa, chamada “gás”, usada para custear a operação da rede, em outras palavras, recompensar os mineradores pelo poder computacional cedido. No contexto dos contratos inteligentes o custo se dá de duas maneiras, na implantação e na execução, que no final das contas também são transações. A implantação do contrato pode ser vista como uma chamada a uma função especial construtora do contrato, portanto, o custo de implantação pode representar a complexidade do contrato. Já o custo da execução depende das instruções a serem executadas e se existem ou não chamadas a métodos de outros contratos, isto é, transações internas.

Visando reduzir os custos de implantação e execução dos contratos, na Blockchain Ethereum, o gás total de um bloco é limitado. Um contrato inteligente complexo pode custar muito gás e inviabilizar a sua implantação, ou seja, o bloco não conterá a transação. Além disso, quanto maiores forem os custos de execução do contrato, menor será o rendimento das execuções dele e mais tempo os usuários terão que aguardar pelas confirmações das execuções. Porém, tal abordagem não barateou o custo de operação, ela apenas impediu que ele atingisse determinados valores, sendo assim, o custo continua alto e supondo que haja muito recurso disponível para manter um determinado contrato inteligente, ele será limitado pelo total de gás permitido no bloco.

4.4. Soluções de escalabilidade na Blockchain

Escalabilidade é um termo utilizado em sistemas, que diz respeito à capacidade de um sistema crescer, tendo como intenção atender mais usuários ou adicionar mais funcionalidades. Sendo assim, um sistema é dito escalável quando o seu desempenho aumenta proporcionalmente com o seu poder computacional.

Podemos dividir escalabilidade em duas vertentes, escalabilidade horizontal e escalabilidade vertical. Na escalabilidade horizontal aumentamos o poder computacional do sistema adicionando nós ao sistema, ou seja, adicionando uma nova máquina. Já na escalabilidade vertical aumentamos o poder computacional do sistema melhorando um nó existente, como por exemplo, adicionando mais memória RAM. Porém há sistemas em que o aumento no poder computacional não acarreta um aumento no desempenho do sistema, nestes casos é necessário rever a arquitetura.

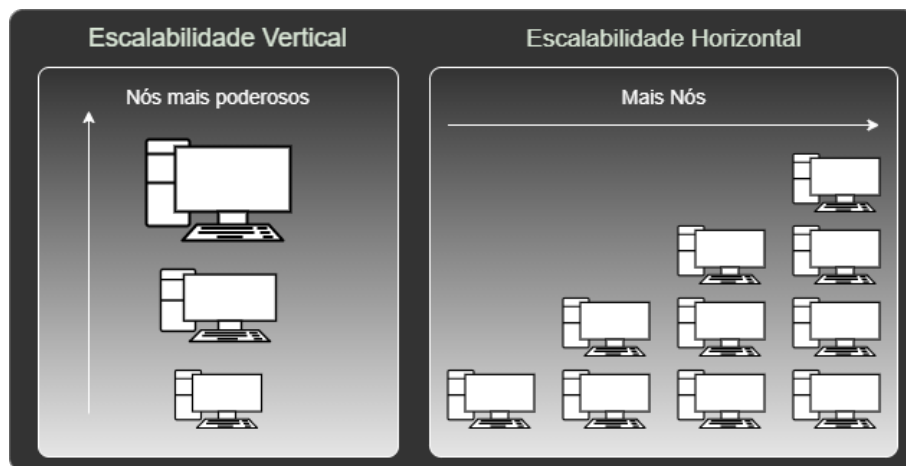


Figura 4.6. Tipos de Escalabilidade

Soluções de Blockchain tradicionais, como por exemplo Bitcoin e Ethereum, apresenta problemas de escalabilidade, no que diz respeito ao tempo para confirmar uma transação e vazão (quantas transações são confirmadas por segundo). Em especial, quando comparado a sistemas de cartões de crédito. Esses, por sua vez, são capazes de processar mais de duas mil transações por segundo. Nos sistemas de cartões de crédito, a escalabilidade pode ser tanto vertical, quanto horizontal, pois trata-se de um sistema centralizado. Portanto, a entidade central pode analisar e investir recursos para melhorar o desempenho da forma que julgar conveniente.

Como a Blockchain é uma rede P2P, a sua escalabilidade vai se dar principalmente da forma horizontal, no qual novos validadores (mineradores) vão se juntar à rede e o poder computacional do sistema irá aumentar. Em se tratando de prova de trabalho, o modelo que é diretamente afetado pelo poder computacional, aumentar a capacidade de um nó não gera ganho para a rede, apenas uma vantagem para este nó em detrimento dos demais. Para os demais modelos, essa desigualdade é irrelevante para a escalabilidade, uma vez que um nó possui apenas um voto ou uma maior probabilidade de ser sorteado.

Nas Blockchains tradicionais, apesar de termos esse aumento de poder computacional, nós não temos um aumento proporcional no desempenho do sistema. No caso do Bitcoin, nem mesmo há um aumento, o desempenho da rede se mantém constante, independentemente do poder computacional. Levanta-se, então, um questionamento sobre o porque a Blockchain tradicional não escalar. Para responder esta pergunta, devemos observar as decisões de arquitetura desta classe de Blockchain. Decisões na arquitetura de sistemas são feitas com base em tradeoff. Tratando-se de Blockchain e performance, devemos olhar basicamente três pontos principais: a quantidade de blocos que cada nó armazena, o algoritmo de consenso e o tamanho dos blocos da cadeia. Para a análise dos pontos mencionados, teremos como caso de estudo a Bitcoin, dado que, outras Blockchains que se encaixam na categoria de Blockchain tradicional foram muito influenciadas por ela.

O primeiro ponto diz respeito à quantidade de blocos que cada nó armazena, ou seja, o quanto da cadeia de blocos cada nó guarda. No caso da Bitcoin, cada nó armazena toda a cadeia de blocos. Isto traz como benefícios, uma maior disponibilidade das

informações, uma vez que ela está replicada em todos nós da rede. Porém, traz também desvantagens como uma maior dificuldade de manter a coerência destes dados e um desafio de performance, pois toda vez que for feita uma consulta estaremos consultando a base de dados completa.

Como segundo ponto temos o algoritmo de consenso. Como mencionado anteriormente, o algoritmo de consenso é responsável por manter a consistência dos dados armazenados pelos nós da rede. Essa tarefa se torna consideravelmente mais lenta uma vez que cada nó armazena a cadeia completa. No caso da Bitcoin o algoritmo de consenso utilizado é o Proof of Work, que é executado por máquinas que são chamadas de mineradores. Fundamentalmente, ele funciona como um puzzle computacional, no qual deve-se simplesmente buscar por um valor que faça parte do conjunto de soluções do puzzle. Esta ideia que faz com que o algoritmo de consenso funcione, mas também é o seu maior problema, pois acabamos tendo um desperdício de poder computacional.

Quando um minerador na Bitcoin descobre a solução para uma prova, ele envia uma mensagem *Broadcast* para avisar aos outros nós da rede que aquela prova foi resolvida e o bloco pode ser fechado. Entretanto, existe um tempo de propagação dessa mensagem e, enquanto isso, um outro nó pode encontrar uma outra solução para a prova. A partir daí podemos acabar gerando duas cadeias de blocos diferentes, dado que os mineradores podem pegar um conjunto de transações diferentes para tentar formar um bloco. Este cenário é chamado de *fork*. É importante ressaltar que, o *fork* pode ser enxergado como uma anomalia, pois a existência de nós com cadeias de blocos diferentes vai contra a ideia de consenso. Esta situação é resolvida simplesmente adotando a maior cadeia como a correta. Assim temos novamente um desperdício computacional, pois, apenas um *fork* é escolhido para representar a cadeia, desperdiçando todo o poder computacional investido na construção dos outros *forks*.

Por fim, o terceiro ponto temos que, na Bitcoin foi definido um tamanho máximo de 1MB para os blocos. Este tamanho para o bloco parecia viável quando a Bitcoin foi concebida. Com o passar dos anos, a criptomoeda se popularizou muito e por conta disso a quantidade de transações aumentou também, fazendo com que haja uma demora ainda maior ao confirmar as transações. É necessário que um bloco seja fechado contendo todas as transações, e como os blocos são muito pequenos se comparados ao volume de transações disponíveis, acaba formando-se um grande volume de transações à espera de um bloco para incorporá-las.

Com base nesses três pontos, é possível entender porque a Bitcoin não escala. Nesta temos um atraso de confirmação das transações, que se dá justamente por conta do tempo necessário para os mineradores solucionarem o puzzle do Proof of Work. Também por conta do tamanho dos blocos, esse tempo de confirmação é maior que 10 minutos. Além da demora para a primeira confirmação da transação, temos a situação dos *forks*, que podem fazer com que o *fork* que já tinha processado determinada transação seja descartado em detrimento de um que ainda não a processou. No fim das contas a Bitcoin só é capaz de processar apenas aproximadamente 4 transações por segundo [L. 2019]. Isso destoa muito, se comparado a um sistema de cartão de crédito que atualmente é capaz de processar aproximadamente 1700 transações por segundo [L. 2019], como no caso de sistemas de cartão de crédito atuais. Além de uma vazão muito maior, sistemas de cartão

de crédito também possuem um tempo de confirmação de transação menor, apresentando, então, alta vazão e baixa latência. Tratando-se de criptomoedas e Blockchain tem-se como meta um desempenho tão bom quanto o dos sistemas de cartões de crédito.

Para reduzir a disparidade com os sistemas de cartão de crédito, processar transações com uma taxa próxima a eles, soluções de Blockchain usam uma combinação de técnicas *On-chain* ou *layer-1* e *Off-chain* ou *layer-2*. As soluções *On-chain* são propostas que alteram o protocolo padrão da Blockchain visando a escalabilidade. Um exemplo desse modelo de proposta é a técnica de *Sharding*. As soluções *Off-chain* são implementadas fora da rede principal adicionando camadas de processamento que independem do núcleo da Blockchain. Os Rollups são técnicas *Off-chain* [Zhou et al. 2020a, Tianchen et al. 2021].

4.4.1. *Sharding*

À medida que a popularidade da Blockchain cresce, o mesmo acontece com o volume transacional gerenciado pela rede e a carga de trabalho é ajustada para manter a dificuldade. Se pensarmos em uma Blockchain como um banco de dados compartilhado, à medida que mais e mais dados são adicionados à rede, precisa-se encontrar novas maneiras de processar todos esses dados com eficiência e rapidez. É aí que o *sharding* pode ajudar. [Mearian 2019]

O *sharding* não é uma técnica nova, ele foi originalmente desenvolvido para melhorar a performance de bancos de dados muito grandes, e somente, recentemente, passou a ser explorado como solução para escalabilidade na Blockchain. A técnica consiste na fragmentação ou divisão horizontal de bancos de dados, permitindo que processem mais transações por segundo. O *sharding* divide o sistema em partições menores, conhecidas como “*shards*”. Cada fragmento é composto por seus próprios dados, tornando-o distinto e independente quando comparado a outros fragmentos, permitindo um melhor manuseio dos mesmos, tornando-os menos pesados e mais fáceis de operar.

O *sharding* pode ajudar a reduzir a latência ou lentidão de uma rede, pois divide uma rede Blockchain em fragmentos independentes. No entanto, existem algumas questões de segurança em torno do *sharding* que podem ser exploradas por adversários. Ao subdividir os nós em *shards*, o poder de *hashing* de cada grupo diminuirá consideravelmente. Isso gera um problema de segurança ao permitir que um agente mal-intencionado execute um ataque com mais facilidade. Uma situação que coloca em risco a segurança e a integridade das informações.[bit2me 2020]



Figura 4.7. Tradeoff em Sistemas Distribuídos

Devemos ter em mente o tradeoff que estamos fazendo ao implementar uma Blockchain que faz uso de *sharding*, pois como ilustrado na Figura 4.7, é impossível atingir os três extremos. O *sharding* por sua vez, tende à escalabilidade e descentralização, introduzindo assim, desafios de segurança à Blockchain. Temos então, dois principais desafios introduzidos pelo *sharding* [Wang et al. 2019]:

1. Distribuir os nós de maneira uniforme nos *shards*, de forma a garantir que a maioria deles seja honesta com alta probabilidade;
2. Como garantir que um adversário não obtenha vantagem significativa, enviesando as operações ou criando identidades Sybil, que é quando um único nó consegue burlar a política de identidade e se passar por vários simultaneamente [Douceur 2002].

Outro problema inserido na Blockchain tradicional com a implementação de partições ocorre porque os nós, quando atribuídos a um subgrupo, não têm acesso ao saldo dos nós que pertencem aos outros subgrupos. Tornando necessário criar um protocolo para resolver as transações entre os subgrupos.

Portanto, para uma Blockchain se manter segura e funcional, enquanto faz *sharding*, ela deve, em geral, ser composta por cinco componentes [Wang et al. 2019], que são:

1. **Identity establishment and committee formation:** Para se juntar ao protocolo, cada nó precisa estabelecer uma identidade, como uma chave pública, um endereço IP e uma solução de prova de trabalho (PoW). Cada nó é então atribuído a um comitê correspondente à sua identidade estabelecida. Nesse processo, o sistema precisa prevenir a identidade Sybil. No entanto, uma Blockchain permissionada não requer este processo;
2. **Overlay setup for committees:** Uma vez que os comitês são formados, cada nó se comunica para descobrir as identidades de outros nós em seu comitê. Para uma Blockchain, uma sobreposição de um comitê é um subgrafo totalmente conectado contendo todos os membros do comitê. Normalmente, este processo pode ser feito com um protocolo de fofoca (*gossip protocol*);
3. **Intra-committee consensus:** Cada nó dentro de um comitê executa um protocolo de consenso padrão para concordar com um único conjunto de transações. Nesse

processo, todos os membros honestos devem concordar com o bloco proposto em seu comitê;

4. **Cross-shard transaction processing:** A transação deve ser confirmada atômica-mente em todo o sistema. Para transações cross-shard, os shards relacionados precisam obter consistência. Normalmente, esse processo requer um tipo de transação de “retransmissão” para sincronizar entre os fragmentos relacionados;
5. **Epoch reconfiguration:** Para garantir a segurança dos shards, os shards precisam ser reconfigurados, a cada período de tempo chamado epoch, exigindo uma aleatoriedade. Essa aleatoriedade será usada na próxima epoch.

As soluções que utilizam Sharding podem ainda ser divididas em duas categorias, Partial Sharding e Complete Sharding, dependendo de como a proposta aborda o processamento, armazenamento e comunicação [Hong et al. 2021]. No modelo de partição parcial, os grupos trabalham em conjunto para gerar uma cadeia única compartilhada por todos. Enquanto no completo, cada grupo possui uma cadeia de blocos distinta. Por fim, na categoria outras soluções, iremos analisar propostas que destoam dos padrões vistos nas outras duas. A Figura 4.8 ilustra a diferença entre partial sharding e complete sharding.

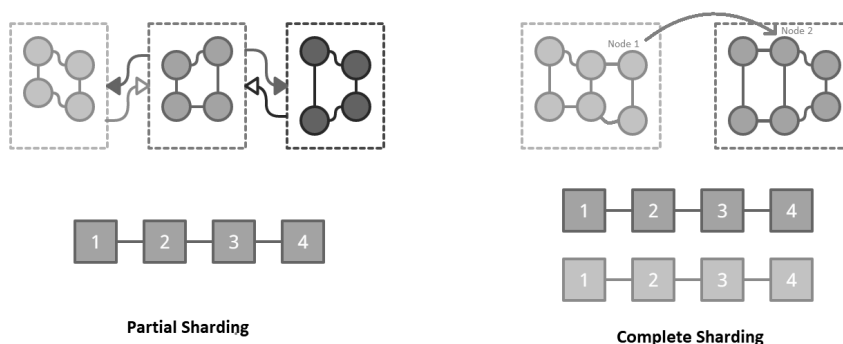


Figura 4.8. Estrutura dos modelos de *sharding*

4.4.1.1. Partial Sharding

As primeiras propostas de *sharding* tentam resolver o problema da escalabilidade dividindo o poder computacional da rede, criando grupos capazes de minerar blocos de forma independente. Porém mantêm a estrutura base da Blockchain onde todos os nós da rede possuem uma cópia exata da Blockchain. Uma vez que cada nó recebe e armazena informações completas do sistema, não existem transações *cross-shard* no sistema, mas os nós ainda sofrem de armazenamento pesado e sobrecarga de largura de banda [Hong et al. 2021]. Uma vez que a fragmentação ocorra em apenas uma das três dimensões processamento, armazenamento e comunicação esse modelo é classificado com *partial sharding*.

4.4.1.2. Complete Sharding

Sharding completo são aqueles que atingem a fragmentação nas bases computação, armazenamento e comunicação. A divisão do poder computacional permite, assim como no modelo parcial, aumentar o número de blocos minerados ao permitir que as partições minerem simultaneamente. Para o armazenamento o objetivo é diminuir o tamanho da cadeia armazenada em cada nó pois a mesma cresce indefinidamente. O consenso baseado em BFT é muito presente nos modelos que visam a escalabilidade pois as provas demandam tempo e esforço, impedindo que muitos blocos sejam minerados. Esses modelos de consenso requerem um grande volume de comunicação e começam a perder sua eficiência em redes maiores. A fragmentação da comunicação busca resolver esse problema.

4.4.2. Soluções *Off-chain*

A estrutura descentralizada da Blockchain oferece benefícios para algumas aplicações do cotidiano. Por exemplo um serviço de carona, como o Uber, contaria com uma redução nas taxas devido a ausência de um intermediário [Sá 2022]. As soluções de *layer-2* são um conjunto de técnicas com objetivo de aumentar a velocidade e vazão das transações. Além disso essas técnicas reduzem o gas das transações que seria um grande empecilho para o surgimento de aplicações integradas à Blockchain.

O Rollup é uma técnica que consiste em empacotar e compactar dados de transações, chamados de lote (batch), off-chain e, em seguida, armazená-los on-chain, Rollup é um termo geral para a tecnologia de Layer-2 que adota esse framework. Esta é uma solução que teve como origem o Sidechain e foi evoluindo até culminar no estado atual, como mostra a Figura 4.9 [Tianchen et al. 2021].

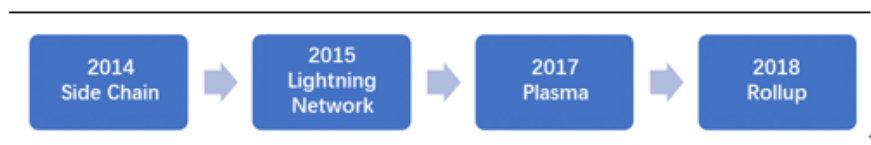


Figura 4.9. Evolução do Rollup [Tianchen et al. 2021]

Side Chains são cadeias que processam dados em uma velocidade maior ao reduzir a descentralização e segurança. Uma vez que a side chain atua de forma independente, a segurança da rede principal é preservada. Rootstock [Lerner 2015] é uma side chain associadas ao Bitcoin responsável por habilitar smart contracts na rede. Polygon usa Proof of Stake para processar transações de forma mais rápida e guarda snapshots dos resultados no Ethereum.

A Lightning Network é uma rede executada sobre a Bitcoin. Essa rede permite que dois usuários abram um canal de troca par a par ao aportarem moedas na rede principal. Os usuários passam a poder trocar livremente até o limite aportado. No Bitcoin são registrados somente as transações de abertura e fechamento do canal.

Plasma é um framework de cadeias secundárias que atuam sobre a rede Ethereum. O Plasma difere das Side Chains pois é contruída sobre um contrato inteligente. Cada cadeia da Plasma é um contrato customizado para um propósito. [Binance 2020]

4.4.2.1. Rollups

A técnica de rollups consiste em acumular as transações em lotes antes de adicioná-los a Blockchain. As transações são processadas fora da rede principal, compactadas em um lote e então adicionadas a Blockchain. Um smart contract registrado na rede mantém os lotes em uma árvore Merkle. Toda vez que um novo lote é adicionado, caso o lote gere uma nova árvore, novo hash, a árvore é registrada na Blockchain.

Os dados processados fora da rede principal devem ser verificados antes de serem adicionados para preservar a confiabilidade. A forma de verificar os dados gerou duas categorias para a técnica de Rollups, uma baseada em Fraud Proof (Optimistic Rollups) e outra baseada em Validity Proofs (Zero Knowledge Rollups) [Buterin 2021].

No Optimistic Rollup, o contrato de rollup acompanha todo o histórico de estado das raízes e o hash de cada lote. Se alguém descobrir que um lote gerou uma raiz incorreta, eles podem publicar uma prova na Blockchain, provando que o lote foi calculado incorretamente. O contrato verifica a prova e reverte esse lote e todos os lotes posteriores.

No Zero Knowledge Rollup, cada lote inclui uma prova criptográfica chamada ZK-SNARK (por exemplo, usando o protocolo PLONK), que prova que a raiz gerada pelo lote é o resultado correto da execução do lote. Não importa quão grande seja o cálculo, a prova pode ser verificada muito rapidamente na cadeia.

4.5. Cartesi

Continuando com soluções de escalabilidade na Blockchain, nesta seção, nos aprofundaremos em uma delas, a Cartesi. A solução é apresentada e são detalhadas suas principais tecnologias, a Cartesi Machine e o Cartesi Rollups, que serão utilizadas na criação dos DApps na seção seguinte. A principal referência para esta seção é a documentação oficial, disponível no site oficial da Cartesi [Cartesi 2022].

A Cartesi é uma plataforma de layer-2 para o desenvolvimento de aplicações descentralizadas. Ela permite que DApps sejam desenvolvidos utilizando linguagens de programação convencionais ao oferecer um sistema operacional Linux acoplado a uma infraestrutura Blockchain. Por exemplo, contratos inteligentes para a Ethereum devem ser escritos utilizando Solidity e, ao utilizarmos a Cartesi podemos escrever com linguagens mais familiares, como Python ou C++. Além disso, diversos softwares, ferramentas, bibliotecas e serviços utilizados para a criação de aplicações tradicionais ficam a disposição do desenvolvedor, garantindo um ambiente de desenvolvimento familiar, diminuindo a barreira de entrada para a criação de aplicações descentralizadas e livrando os desenvolvedores das limitações e especificidades impostas por cada Blockchain.

Na Cartesi, tem-se duas tecnologias principais. A primeira é a Cartesi Machine, uma máquina virtual que permite computação verificável usando um sistema operacional Linux. A segunda é o Cartesi Rollups, que fornece uma estrutura geral para criação de DApps com a combinação da Cartesi Machine com Optimistic Rollups.

Assim, podemos dizer que a Cartesi busca:

- Oferecer um sistema operacional completo para aplicativos Blockchain;

- Solucionar o problema de escalabilidade usando o Cartesi Rollups para suportar cálculos complexos;
- Possibilitar o desenvolvimento de DApps de qualquer complexidade usando ferramentas de desenvolvimento convencionais em cima de redes Blockchain estabelecidas, como Ethereum, Polygon, Avalanche e BNB Smart Chain.

4.5.1. Cartesi Machine

O componente mais importante para a solução Cartesi, a Cartesi Machine foi criada para que desenvolvedores possam utilizar ferramentas tradicionais para a criação de aplicativos descentralizados escaláveis.

Diferentemente das VM tradicionais, a Cartesi Machine não se destina a ser interativa. Em vez disso, ela deve inicializar, abrir o sistema operacional, executar algum software predefinido e depois parar [Cartesi 2022].

Os desenvolvedores costumam trabalhar com cadeias de ferramentas que operam num nível de abstração mais alto. Essas cadeias de ferramentas os isolam de detalhes de hardware irrelevantes e até mesmo dos detalhes de um determinado sistema operacional. Inventar uma nova arquitetura ad-hoc exigiria a portabilidade de uma cadeia de ferramentas e sistema operacional. Em vez disso, as Cartesi Machines são baseadas em uma arquitetura comprovada para a qual uma cadeia de ferramentas padrão e um sistema operacional já são acessíveis.

Para que a Cartesi Machine seja verificável, a Blockchain deve, portanto, hospedar uma implementação de referência de toda a arquitetura. Se for confiável, essa implementação deve ser facilmente auditável. Para isso, tanto a arquitetura quanto a implementação devem ser abertas e relativamente simples. Juntos, esses requisitos apontam para o RISC-V [Teixeira and Nehab 2018]. O emulador de Cartesi Machine implementa o ISA RV64IMASU da RISC-V e é escrito em C/C++ com dependências POSIX restritas às instalações de terminal, processo e mapeamento de memória. O RISC-V garante estabilidade simulando hardware real, podendo trabalhar em um nível mais baixo, com mais ferramentas de desenvolvimento.

Escalabilidade

A Escalabilidade nas Cartesi Machines existe, uma vez que elas podem processar quantidades praticamente ilimitadas de dados e em um ritmo 4 ordens de magnitude mais rápido. Isso é possível porque as Cartesi Machines são executadas off-chain, livres da sobrecarga imposta pelos mecanismos de consenso usados pelas Blockchains. Em um cenário típico, uma das partes envolvidas em um DApp executará a Cartesi Machine off-chain e relatará seus resultados à Blockchain. Diferentes partes não precisam confiar umas nas outras porque a plataforma Cartesi inclui um mecanismo automático de disputa para as Cartesi Machines. Todos os interessados repetem o cálculo off-chain e, caso seus resultados não concordem, entram em uma disputa, que o mecanismo garante ser sempre vencida por uma parte honesta contra qualquer parte desonesta, isso só é possível pois:

- As Cartesi Machines são independentes — elas funcionam isoladas de qualquer influência externa na computação;
- As Cartesi Machines são reprodutíveis — Duas partes realizando a mesma compu-

tação sempre obtêm exatamente os mesmos resultados;

- As Cartesi Machine são transparentes — Elas expõem todo o seu estado para inspeção externa.

Produtividade

O desenvolvimento de software moderno envolve a combinação de dezenas de componentes prontos para uso. A criação desses componentes exigiu o esforço conjunto de uma comunidade mundial ativa ao longo de várias décadas. Todos eles foram desenvolvidos e testados usando conjuntos de ferramentas bem estabelecidos (linguagens de programação, compiladores, linkers, profilers, depuradores, etc.), e contam com vários serviços fornecidos por sistemas operacionais modernos (gerenciamento de memória, multitarefa, sistemas de arquivos, rede, etc). Ao disponibilizar um sistema operacional com suporte a tais ferramentas, a Cartesi Machine é capaz de preservar essa produtividade no ambiente de desenvolvimento de DApps.

Caso fosse necessária a utilização massiva de ferramentas desenvolvidas especificamente para desenvolvimento em Blockchain e o abandono de ferramentas já conhecidas, o fator econômico seria negativo a ponto de possivelmente tornar inviável a adoção de nova tecnologia. Além da economia em se utilizar ferramentas que já se tenha domínio, também obtém-se ganho no quesito segurança, utilizando tecnologias exaustivamente testadas por diversos desenvolvedores.

Arquitetura

A arquitetura da Cartesi Machine pode ser vista de diferentes perspectivas, dependendo de qual nível estamos, são elas:

- **Host Perspective:** Este é o ambiente fora do emulador Cartesi Machine. É mais relevante para desenvolvedores configurando Máquinas Cartesi, executando-as ou manipulando seu conteúdo. Inclui a API do emulador em: C++, Lua, gRPC e a interface de linha de comando.
- **Target Perspective:** Este é o ambiente dentro da Cartesi Machine. Ele engloba a arquitetura RISC-V da Cartesi, bem como a organização do sistema operacional Linux embutido que é executado em cima dela. É mais relevante para os programadores responsáveis pelos componentes do DApp que são executados off-chain, mas devem ser verificáveis. A cadeia de ferramentas de compilação cruzada e as ferramentas usadas para construir o kernel do Linux e os sistemas de arquivos raiz do Linux incorporados também são importantes nessa perspectiva, embora sejam usadas no host;
- **Blockchain Perspective:** Esta é a visão que os contratos inteligentes têm das Máquinas Cartesi. Consiste quase exclusivamente na manipulação de hashes criptográficos do estado das Máquinas Cartesi e partes das mesmas. Em particular, usando apenas operações de hash, a Blockchain pode verificar asserções sobre o conteúdo do estado e obter o hash de estado que resulta de modificações no estado.

4.5.2. Cartesi Rollups

Os Cartesi Rollups são a versão da Cartesi de Optimistic Rollups. O objetivo é mover a maior parte da computação para fora da Blockchain, usando-a como provedora de

dados para a execução off-chain das aplicações. Pode-se encontrar mais detalhes em [Moura 2021, Argento 2021]. Essa solução possui componentes em layer-1 e em layer-2. Na layer-1, temos os contratos inteligentes dos Cartesi Rollups. Eles fazem a mediação dos contratos inteligentes de propriedade externa com os componentes da layer-2, onde a computação é realizada pelos Cartesi Nodes.

O Cartesi Node é o componente de layer-2 responsável por executar os cálculos das aplicações utilizando uma Cartesi Machine. O Cartesi Node faz o uso de um middleware para ler dos dados dos contratos inteligentes em layer-1 e fornecê-los à Cartesi Machine, dessa forma este middleware estabelece a comunicação entre os layers 1 e 2. Temos dois tipos de Cartesi Nodes e eles diferem na forma como interagem com os contratos de rollup em layer-1, são eles: validadores e usuários. Um Cartesi Node validador é responsável por garantir que os dados enviados para a Blockchain sejam confiáveis, lutando contra validadores desonestos para assegurar a que reivindicações honestas sejam cumpridas. Já um Cartesi Node usuário não realiza essas validações, ele se preocupa apenas em avançar o estado de sua máquina, consumindo os dados da Blockchain.

Ao invés de verificar cada nova atualização do estado da Cartesi Machine, os Cartesi Nodes validadores fazem isso fim de cada *epoch*. Um *epoch* é um conjunto de entradas que sejam agrupadas que seguem o mesmo ciclo. Com a verificação dessa forma, evitamos uma interação excessiva com a Blockchain. O back-end da aplicação lê a carga de entrada e a processa.

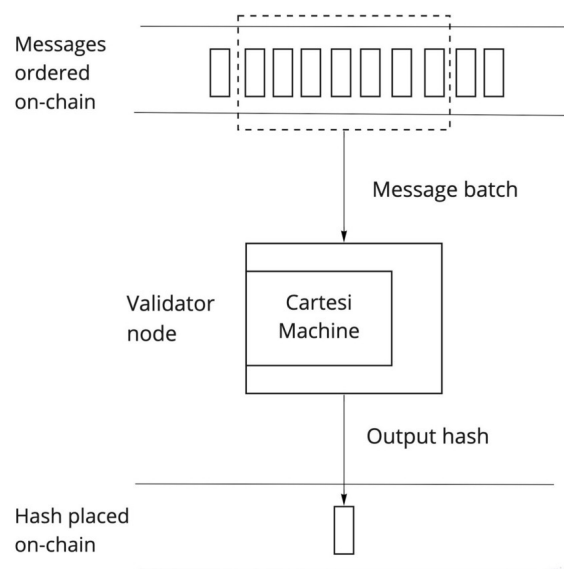


Figura 4.10. Agrupamento das transações e armazenamento do hash de saída [Moura 2021]

Seguindo o protocolo do Cartesi Rollups, quando é processado o conteúdo de cada epoch, um Cartesi Node validador insere na Blockchain o hash que representa o estado do contrato de layer-2 e, então inicia-se o período de desafio. Caso não ocorram disputas o estado representado pelo hash é assumido. Caso contrário, o mecanismo de disputa é acionado para garantir que a reivindicação correta seja aplicada.

Devido a natureza determinística das Cartesi Machines, qualquer Cartesi Node interessado no cálculo da aplicação pode perceber um comportamento desonesto de um outro Node alegando uma saída falsa para a computação.

Diferentemente de como ocorre as transações envolvendo contratos inteligentes de layer-1 na Ethereum, no Cartesi Rollups, as Cartesi Machines são criadas especificamente para executar contratos únicos de layer-2, ao invés de processar todas as transações como faz a EVM. Assim, cada contrato tem seu conjunto específico de Cartesi Nodes validadores, permitindo um ganho de escalabilidade e mantendo a segurança desde que tenhamos pelo menos um validador honesto.

Vouchers e Notices

A Cartesi Machine pode responder e interagir com contratos em layer-1 com os chamados *vouchers*. Um voucher é uma combinação de um endereço de destino e uma carga em bytes. Em sua execução, um voucher envia uma mensagem para o endereço de destino com a carga como parâmetro. Além dos vouchers, a Cartesi Machine também pode produzir *notices*. Uma notice é uma carga em bytes que é enviada para fins informativos.

4.5.2.1. Componentes em layer-1

Na Blockchain, o Cartesi Rollups utiliza contratos inteligentes para mediar a relação dos componentes em layer-2 com outros contratos inteligentes e contas externas. A seguir, apresentamos uma breve descrição de cada dos componentes de layer-1.

Contrato de Entrada

Este contrato recebe as entradas do usuário para a execução da computação. O conteúdo das entradas é sempre guardado em *calldatas* na Blockchain, mantendo um único hash para cada entrada de uma epoch ativa. Este hash resume tanto a entrada em si quanto seus metadados. As entradas não são interpretadas em layer-1, é papel da Cartesi Machine, em layer-2, detectar se há algum problema no conteúdo de cada entrada. Dessa forma, os recursos de execução em layer-1 são economizados.

Contrato de Saída

O contrato de saída pode tanto verificar a validade do conteúdo de uma notice, quanto executar um voucher. Este módulo garante que apenas as saídas sugeridas pela layer-2 e finalizadas na layer-1 possam ser executadas. Ele faz isso exigindo que uma prova seja enviada com a chamada de execução.

Portal

O Portal é a forma de transferir ativos da layer-1 para a layer-2. Com os ativos em layer-2, pode-se transacioná-los de forma menos custosa, utilizando entradas simples que são entendidas pela lógica Linux. Pode-se pensar no Portal como uma conta bancária, de propriedade da máquina off-chain.

Validator Manager

Este componente é responsável pelo gerenciamento das reivindicações, conceder

permissões e aplicar punições.

Dispute Resolution

Este módulo tem o papel de mediar as interações entre os Cartesi Nodes validadores interessados na disputa. Uma disputa ocorre quando dois os mais validadores reivindicam atualizações de estados diferentes para uma mesma epoch.

Cartesi Rollups Manager

Este componente realiza a sincronização entre os demais componentes. Ele é responsável pela mediação entre o componente Dispute Resolution e o Validator Manager. Além disso, ele notifica todos os outros módulos qualquer alteração de fase.

4.5.2.2. Componentes em layer-2

Como citado anteriormente, a layer-2 do Cartesi Rollups é constituída de Cartesi Nodes. Agora exibiremos uma pequena descrição de alguns componentes internos de um Cartesi Node.

State Server

Este é um serviço com papel de assegurar que os outros componentes tenham acesso a uma visão consistente da Blockchain. Ele monitora todas as atividades relevantes dos contratos inteligentes do Cartesi Rollups, guardando as informações que são emitidas pelas Blockchain.

Server Manager

Este componente administra a Cartesi Machine, enviando entradas e obter as saídas produzidas. É responsável por iniciar e parar a máquina conforme apropriado, bem como fornecer uma API para os outros módulos para consultar o estado da máquina.

Rollups Dispatcher

Este módulo é responsável por interpretar o estado atual dos contratos inteligentes da Cartesi Rollups, ele informa o Gerenciador de Servidores sobre quaisquer entradas recebidas e, no caso do Validator Nodes, também envia transações correspondentes a reclamações de atualização do estado. Ele também lidará com quaisquer disputas caso surjam.

Rollups Indexer

Este serviço obtém informações da layer-1 pelo State Server e os dados produzidos pelo back-end da aplicação para consolidar o estado de um Cartesi Node.

Query Server

Para que usuários e clientes recuperem vouchers e notices produzidos pelo back-end da aplicação, este componente provê uma API GraphQL para consulta do estado do Cartesi Node.

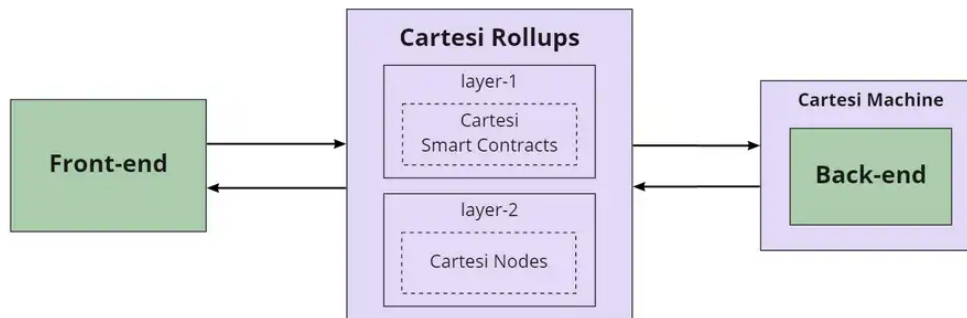


Figura 4.11. Componentes dos Cartesi Rollups [Cartesi 2022]

Perceba, na Figura 4.11, que, de forma distinta às aplicações convencionais, em um DApp Cartesi, não temos a comunicação direta entre front-end e back-end. Ao invés disso, o front-end envia as entradas para o framework Cartesi Rollups, que as dispõe para as instâncias de back-end executadas em cada Cartesi Node. Após o processamento do back-end, as saídas retornam ao Cartesi Rollups que as verifica e, possivelmente, as corrige e as disponibiliza para o front-end. Essa comunicação é realizada por um conjunto de APIs HTTP da Cartesi.

Utilizando uma série de *endpoints* HTTP o back-end comunica ao Cartesi Rollup que qualquer processamento ou inicialização anterior foi concluída, e que o back-end está pronto para lidar com a próxima solicitação, que por sua vez, pode enviar uma consulta sobre o estado atual do aplicativo ou fornecer uma entrada que será processada pelo back-end e atualizará seu estado. Assim, o back-end pode acessar endpoints HTTP para informar os resultados da computação realizada.

O front-end da aplicação também precisa acessar o Cartesi Rollups para enviar solicitações do usuário e recuperar as respostas correspondentes computadas pelo back-end. Essa comunicação pode ocorrer de diferentes formas:

- **AddInput:** uma transação regular na layer-1 para enviar os dados de entrada para os contratos inteligentes de rollup.
- **QueryOutputs:** envia uma consulta à layer-2 para recuperar notíes e vouchers.
- **InspectState:** envia uma consulta a um Cartesi Node para receber o estado de um aplicativo.
- **ExecuteVoucher:** uma transação regular na layer-1 solicitando a execução de contrato inteligente de rollup.

4.6. Cartesi Rollups DApp: Uma Abordagem Hands On

Com base no que foi visto a respeito do Cartesi Rollups, a arquitetura de um DApp Cartesi e como ocorre a comunicação do front-end com o back-end, vamos para o desenvolvimento prático de uma aplicação. Parte importante deste desenvolvimento é a idealização do DApp e, para tal, utiliza-se o modelo de desenvolvimento exposto na Figura 4.12, seguindo a linha de desenvolvimento da esquerda para a direita. [Cartesi 2022]

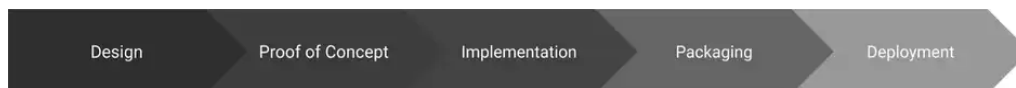


Figura 4.12. Ciclo de vida do DApp [Cartesi 2022]

4.6.1. Modelo de Desenvolvimento

O modelo proposto visa aproximar o desenvolvimento de um DApp do de uma aplicação "tradicional", ou seja, não descentralizada. O objetivo de tal abordagem é tornar o desenvolvimento familiar, mesmo para aqueles que nunca desenvolveram um DApp, permitindo a utilização de ambientes de desenvolvimento mainstream na maior parte do processo. Com isso em mente, nos aprofundaremos em cada uma das etapas deste modelo.

Design

Inicialmente, define-se o design conceitual da aplicação, especificando o escopo principal da lógica de front-end e back-end, bem como os requisitos gerais de tecnologia.

Proof of Concept

Etapa na qual implementa-se uma prova de conceito visando garantir a viabilidade de execução da lógica de back-end pretendida e pilhas de tecnologia dentro de uma Cartesi Machine. Este estágio pode envolver a compilação cruzada de uma biblioteca para o sistema operacional Linux RISC-V da Cartesi e a verificação de que o desempenho da execução do software pretendido na Cartesi Machine é satisfatório.

Implementation

Este é o estágio em que a lógica de front-end e back-end é realmente implementada, representando a maior parte do trabalho no ciclo de vida de desenvolvimento do DApp. Nesta fase, o desenvolvedor do DApp fará uso do Cartesi Rollups Host Environment para implementar os módulos front-end e back-end do DApp. Ele executará a aplicação em Modo Host, onde a lógica do back-end é executada nativamente em localhost. Este modo de execução é importante para que os desenvolvedores possam testar e depurar as aplicações utilizando as ferramentas na qual já estão acostumados.

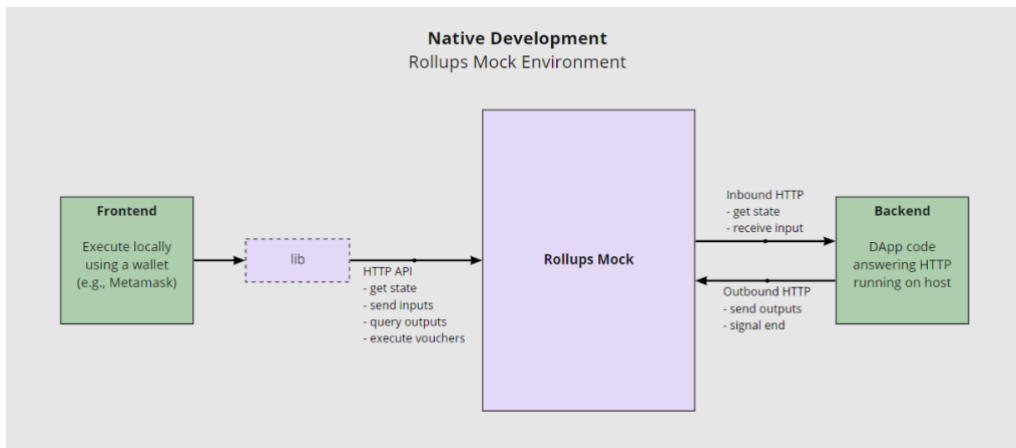


Figura 4.13. Modo Host

Packing

Nesta etapa, a parte de back-end é empacotada para rodar dentro de uma Cartesi Machine, para ser executada por um nó real da Camada 2 (ou seja, não uma versão de desenvolvimento na qual o back-end é executado diretamente no host). A viabilidade de executá-lo dentro de uma Cartesi Machine já deve ter sido acessada durante o Estágio 2, e agora veremos apenas se a implementação completa do DApp também roda satisfatoriamente. Este modo de execução é chamado Modo de Produção. A execução neste modo é fundamental, pois emula a execução da aplicação da forma como será quando for implementada. Isso significa que qualquer código escrito em linguagens compiladas, como C++ ou Rust, deve ser compilado para a arquitetura RISC-V da Cartesi Machine. Nesse modo, a computação realizada pelo back-end é reproduzível e, portanto, verificável, permitindo uma execução verdadeiramente confiável e descentralizada. [Cartesi 2022]

Como o back-end continuará usando a mesma API HTTP de antes, ela permanece inalterada. A única diferença é que desta vez ele irá interagir com um serviço rodando dentro da própria Cartesi Machine, ao invés de se comunicar diretamente com a infraestrutura do Rollups Host como no Estágio 3.

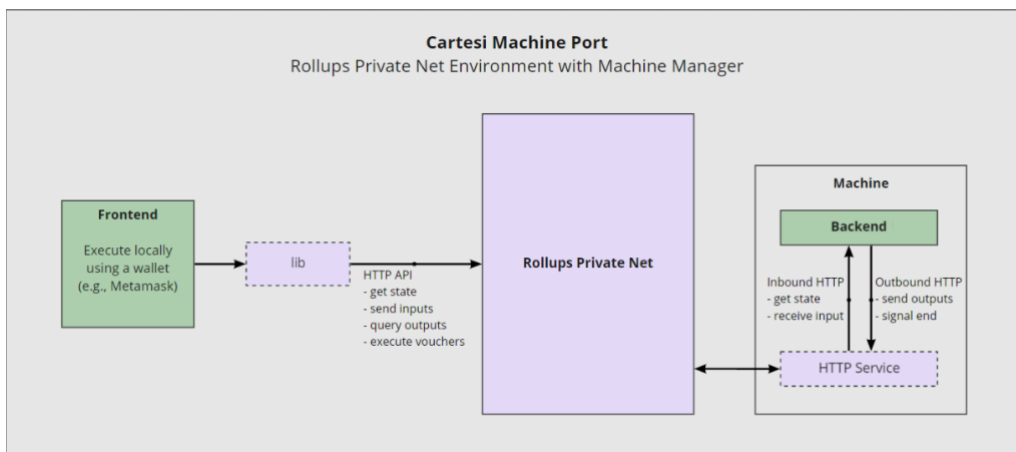


Figura 4.14. Modo de Produção

Deployment

Nesta etapa, a parte de back-end é empacotada para rodar dentro de uma Cartesi Machine, para ser executada por um nó real da Camada 2 (ou seja, não uma versão de desenvolvimento na qual o back-end é executado diretamente no host). A viabilidade de executá-lo dentro de uma Cartesi Machine já deve ter sido verificada durante o Estágio 2, e agora veremos apenas se a implementação completa do DApp também roda satisfatoriamente.

Como o back-end continuará usando a mesma API HTTP de antes, ela permanece inalterada. A única diferença é que desta vez ele irá interagir com um serviço rodando dentro da própria Cartesi Machine, ao invés de se comunicar diretamente com a infraestrutura do Rollups Host como no Estágio 3.

4.6.2. Ferramentas Necessárias

Antes de iniciar a criar aplicações vamos instalar algumas ferramentas úteis que serão necessárias para o desenvolvimento dos DApps Cartesi.

Docker

Docker é uma plataforma de código aberto utilizada para criar e administrar containers (ambientes isolados), onde suas aplicações podem ser executadas. Além disso, podemos compartilhar estes containers com outros usuários de Docker. A Cartesi o utiliza para distribuir a estrutura do Cartesi Rollups. Para instalá-lo siga os passos descritos no site oficial.

Docker Compose

Docker Compose é um gerenciador de containers da Docker. Com ele, podemos iniciar múltiplos containers simultaneamente e definir a forma como eles interagem uns com os outros. Com o Compose, podemos instanciar localmente um ambiente completo de Cartesi Rollups, para que possamos testar os DApps usando apenas sua máquina de desenvolvimento. Para adicionar o Docker Compose à sua máquina siga as instruções do site oficial.

Node.js e NPM

Node.js é um ambiente de execução JavaScript com inspirações no interpretador V8 do Google que nos permite executar códigos JavaScript sem depender de um navegador web. Ele normalmente é distribuído com o gerenciador de pacotes NPM. Pode-se encontrar os passos para a instalação no site oficial do Node.js.

Yarn

Yarn é um gerenciador de pacotes e o utilizamos para adicionar os pacotes aos nossos DApps Cartesi. Para instalá-lo acesse o site oficial do Yarn.

4.6.3. Criando o ambiente

A seguir são expostas as instruções para baixar e iniciar a operação de um Dapp. O primeiro passo é baixar o repositório e reconstruir o back-end, executando os seguintes comandos em um terminal Linux.

```
1 git clone <endereco_do_repositorio>
2 cd rollups-examples/<nome_do_Dapp>
3 make machine
```

O primeiro comando baixa uma cópia do repositório no diretório atual, o segundo comando altera o diretório atual para o diretório do Dapp, e o terceiro comando executa uma série de Makefiles e Shellscripts para construir o sistema de arquivos do Dapp e move-lo para dentro da Cartesi Machine, que é criada na pasta também recém criada chamada **machine**.

Como as interações com a Cartesi Machine serão por meio de comandos Node.js, é preciso operar a partir da pasta **contracts**.

```
1 cd contracts/
2 yarn
```

O Yarn irá instalar o HardHat e demais pacotes necessários para interagir com o contrato, que está na estrutura de simulação do HardHat.

4.6.3.1. Escolhendo o Modo de Execução

Para simular a arquitetura Cartesi, um conjunto de containers docker entra em operação. Há 2 modos de operação. O Modo host, utilizado para desenvolvimento, e o modo de produção. Os comandos para aciona-los, respectivamente, são os 2 a seguir.

É importante ressaltar que eles não operam simultaneamente. Ou se esta em um ambiente ou em outro. E para aciona-los pelo comando docker deve-se estar no diretório **<nome do Dapp>**.

```
1 docker-compose up --build
1 docker-compose -f docker-compose.yml -f docker-compose-host.yml up --
  build
```

Ao final, ou antes de alternar entre ambientes, deve-se pressionar as teclas Ctrl+C e esperar que os containers sejam desativados, feito isso, é possível também utilizar os comandos abaixo para deletar os volumes criados pelos containers. O primeiro comando é para o modo de produção e o segundo é para o modo host.

```
1 docker-compose down -v
1 docker-compose -f docker-compose.yml -f docker-compose-host.yml down -v
```

4.6.4. LGPD-Compliant Datasets Dapp

Este exemplo, que chamamos de LGPD Datasets Dapp, expõe como implementar uma plataforma de provisionamento de modelos de aprendizado de máquinas ciente da procedência dos dados, com base na plataforma Amnesia.[Stach. et al. 2020]. Para alcançar este objetivo, é utilizado um banco de dados já criado com uma tabela chamada medical no SQLite, dentro da Cartesi Machine. O passo a passo da instalação e operação do Dapp mostra como construir e interagir com o respectivo aplicativo Cartesi Rollups que atua de

forma transparente sobre os dados pessoais, buscando cumprir requisitos da LGPD - Lei Geral de Proteção de dados, ou da sua equivalente internacional, a GDPR - General Data Protection Regulation.

Por um lado é possível realizar operações predefinidas para criar, atualizar e excluir registros, atuando como titular dos dados. Por outro lado, também se pode operar como controlador/processador de dados para realizar consultas e construir novos conjuntos de dados para distribuição ou processamento específico. O titular dos dados e o controlador/processador de dados são atores dentro da definição da LGPD/GDPR.

O exemplo destaca alguns aspectos da LGPD/GDPR, como a garantia do titular dos dados de não ter mais seus dados em um conjunto de dados ou a garantia de seu direito de saber como seus dados estão sendo usados, bem como a garantia dos consumidores finais de estarem utilizando dados de origem legítima. Direito ao esquecimento e outros aspectos de segurança não estão presentes neste Dapp conceitual. Este Dapp foi elaborado a partir do SQLite Dapp, disponível em <https://github.com/cartesi/rollups-examples/tree/main/sqlite>.

Ao final, restará claro que através das propriedades do Cartesi Rollups de registrar todos os estados da máquina, é possível verificar quais dados foram utilizados em cada subconjunto de dados disponibilizados para propósitos específicos, como, por exemplo, o treinamento de um modelo de aprendizado de máquina que precisa ser retreinado caso um dos registros de sua base de dados de treinamento seja alterado por solicitação do usuário titular dos dados.

Contexto

Ao se demonstrar que a viabilidade técnica de determinado projeto respeita exigências jurídicas, a adoção da tecnologia é revestida de legitimidade e ganha mais notoriedade de empresas e governos [Governo Federal 2021] que possuem rígidas regras de controle, conformidade(*compliance*) e transparência. Este diferencial pode ser uma vantagem competitiva frente a tecnologias que ignoram estes aspectos.

Escalabilidade de Blockchains é o principal desafio encarado por esta tecnologia atualmente. [Zhou et al. 2020b] A Cartesi oferece à sociedade uma solução do tipo *Layer-2* para este desafio. Otimiza-se a computação com operações *off-chain* entre usuários e otimiza-se o armazenamento de dados *on-chain* através de extensivo uso de *hashes*.

Com a Cartesi Machine como base de dados é possível obter pleno controle sobre inserções, atualizações e exclusões. Utilizando o próprio sistema de arquivos da máquina virtual, arquivos .txt ou .csv, ou algum banco de dados leve, como o SQLite, é possível construir uma Cartesi Machine que opere conforme as necessidades da aplicação atendendo aos requisitos da LGPD/GDPR.

Na máquina *off-chain* estará a maior parte possível da lógica de operação e controle, bem como a base contendo dados pessoais do(s) usuário(s).

As entradas de dados serão requisições para inclusão, leitura, atualização ou exclusão de dados, e os dados propriamente ditos. Embora operações de autenticação, autorização e controle de acesso sejam necessárias para uma implementação em um ambiente

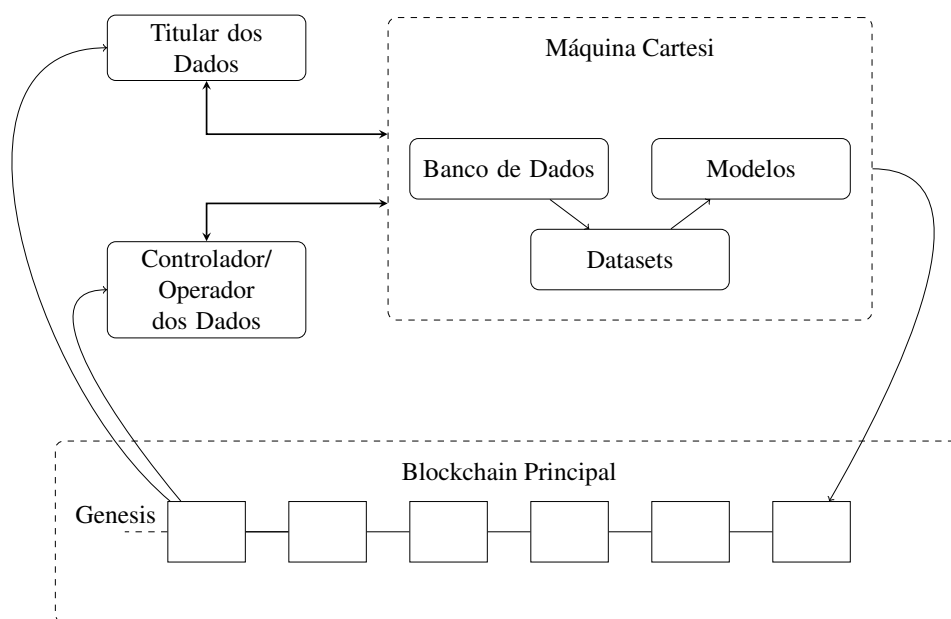


Figura 4.15. LGPD-Compliant Datasets Dapp

de produção real, suas dinâmicas não são tratadas no presente exemplo.

A saída de dados compreende a confirmação da operação e os dados afetados. Após a cada operação, o novo estado é salvo, e novas operações serão sobre o novo estado salvo. A não utilização um estado antigo é o que vai ao encontro do direito ao esquecimento do usuário em um sistema totalmente descentralizado da forma mais segura possível atualmente.

Mais especificamente no contexto do presente exemplo, a Cartesi Machine armazena dados pessoais para fins de treinamento, validação e teste de modelos de aprendizado de máquina. O diferencial desta implementação é o caráter sob-demanda em que o modelo pode ser treinado. Em respeito ao direito do usuário de ser esquecido, o modelo poderia ser totalmente retreinado quantas vezes fossem necessárias nos casos em que um dos titulares dos dados pessoais revogasse o direito de operação de terceiros sobre eles.

Esta abordagem contextualizada pode dar origem a uma nova arquitetura de armazenamento de dados descentralizados de longo prazo e processamento em tempo real em conformidade com requisitos legais de privacidade, como é proposto na Amnésia [Stach. et al. 2020].

Além de propiciar transparência à todos os envolvidos, esta abordagem também fornece maior garantia de que os dados revogados não estão sendo disponibilizados aos antigos interessados. Quem cede os dados tem a garantia das suas corretas utilizações, e quem usa os dados tem garantias de procedência e consentimento atualizado do titular dos dados.

Na Figura 4.15, o **titular dos dados** e o **controlador/operador dos dados** leem na Blockchain a referência ao contrato que permite interagir com os dados.

Caso a ação seja desencadeada pelo **titular dos dados**, este poderá adicionar,

atualizar e deletar seus dados no **banco de dados**, e o **controlador/operador dos dados** terá garantia de procedência dos dados atualizados, bem como seus **modelos** atualizados.

Caso a ação seja desencadeada pelo **operador dos dados**, este poderá ler os conjuntos de dados pertinentes (**conjunto de dados, ou datasets**) e utiliza-los para treinar novos **modelos** de aprendizado de máquina caso o **titular dos dados** tenha concedido autorização para os novos propósitos do treinamento.

Em ambos os casos, ao final, após consenso entre as partes, o estado da **Cartesi Machine** é atualizado e seu hash é salvo na Blockchain.

O presente exemplo tem como foco a implementação focada nos componentes **banco de dados e datasets** da **Cartesi Machine**.

Por motivos de praticidade e pelo próprio propósito do atual construtor de Dapps da Cartesi, as ações das partes interessadas são realizadas de forma assíncrona, ou seja, em uma operação o titular dos dados atual, em outra, o controlador/operador atua. Cada operação corresponde a uma transação completa na Cartesi Machine, auditável e verificável.

Após o **banco de dados/datasets** serem atualizados, a leitura para utilização dos dados pode ser precedida de 2 dos 4 filtros de seleção de dados, a zona de filtro horizontal e a vertical, conforme arquitetura proposta pela Amnesia. [Stach. et al. 2020]

A zona de filtro horizontal equivale a um operador de seleção (σ), removendo origens ou limitando por período de tempo. Este filtro remove registros inteiros.

A zona de filtro vertical equivale a um operador de projeção (π), removendo atributos dos registro(colunas).

Os dados gerados a partir da aplicação dos filtros são armazenados nos **datasets**, que serão utilizados para treinar os **modelos**.

Limitações do direito ao esquecimento

A busca pela conformidade à LGPD/GDPR encontra uma barreira em particular quando se trata da tecnologia Blockchain. Por ser um livro razão, os dados nela inseridos são “eternos” - o que exclui o direito do usuário de exigir o esquecimento dos seus dados por quem os controla e opera. [Truong et al. 2020] [Daudén-Esmel et al. 2021]

Assim, atualmente, a única alternativa para buscar garantir o direito do usuário da Blockchain ao esquecimento é alocar os dados pessoais fora da Blockchain. Dentro desta alternativa, há 2 caminhos possíveis. Ou se armazena os dados pessoais em recipiente centralizado com opção de exclusão - comprometendo a descentralização; ou se armazena em outro sistema descentralizado, como o IFPS - InterPlanetary File System [Benet 2014] que, devido ao seu protocolo de funcionamento, não é possível garantir que todos os nós do sistema deletem determinado dado eventualmente já adquirido.

Uma possível futura alternativa seria a utilização de criptografia completamente homomórfica para realizar queries e outras operações sobre dados criptografados dentro da Cartesi Machine [Gentry 2009]. Como os dados estariam sempre criptografados,

a sua remoção de conjunto de dados para usos específicos poderia ser entendido como esquecimento de tais dados, para fins de processamentos posteriores.

Direito ao esquecimento em Blockchain é um assunto controverso e pouco definido, e não será tratado neste exemplo. Embora haja debates sobre o que é ou não um dado pessoal, ou quando um dado deixa de ser pessoal, o fato é que o quanto mais difícil é recuperar um dado que se deseja esquecer, mais garantida está a satisfação do usuário ao olhos da LGPD/GDPR, que transfere controle absoluto ao usuário sobre seus dados pessoais¹. Dados cifrados cuja chave privada não esteja disponível para uso podem ser considerados dados inacessíveis ou esquecidos.[Stach. et al. 2020] Em contrapartida, é possível que um dia haja possibilidade de recuperar as informações criptografadas mesmo sem conhecimento da chave privada. [Shor 1994] [Shor 1997]

4.6.4.1. Front-End

Nesta aplicação, o front-end tem o papel de oferecer uma melhor interface para o usuário e uma forma de inserir os comandos SQL mais confortável.

Ele foi desenvolvido utilizando Flask, um microframework, escrito em Python, para aplicações web. Além disso, utilizamos alguns módulos do Flask e a biblioteca web3.py.

A biblioteca web3.py é utilizada no desenvolvimento de DApps em Python. Ela é utilizada para facilitar a interação com contratos inteligentes, obter informações da Blockchain, criar transações e muito mais.

O front-end segue o padrão de arquitetura MVC (Model, View e Controller). Nesta implementação, o diretório `templates` é responsável pela componente View da arquitetura. A árvore abaixo exhibe os principais arquivos criados para o funcionamento do front-end da aplicação.

```
front_end
├── config
│   └── connection.py
├── controllers
│   ├── addInput.py
│   └── getNotice.py
├── models
│   └── forms.py
├── static
│   ├── ABI
│   └── bootstrap-5.1.3-dist
└── templates
    ├── addinput.html
    ├── getnotice.html
    └── index.html
```

¹salvo exceções de interesse público

dapp.py

A seguir, descrevemos o papel de cada componente da árvore para a aplicação.

Models

Na pasta `models` temos apenas um arquivo, `forms.py`. Nele, utilizamos o módulo WTForms do Flask para criar os campos para o preenchimento de dados do usuário, que serão utilizados no arquivo `sqlite.py`. O campo `Statement` é onde o usuário insere o código SQL e o campo `Epoch` é onde ele define o epoch no qual quer obter as informações registradas.

View: Templates

Neste diretório, temos os arquivos HTML para exibição das páginas da aplicação. O arquivo `base.html` serve como um template base para os outros arquivos, que são uma extensão dele. Os arquivos `addinput.html` e `getnotice.html` são para a exibição das páginas onde o usuário insere, respectivamente, o statement SQL e o epoch. O arquivo `index.html` exibe a página inicial da aplicação.

Controllers

Na pasta `controllers` estão os arquivos que executam a lógica da aplicação. O arquivo `addInput.py` é responsável por executar o método `addInput`, que envia o statement SQL para o back-end da aplicação. O arquivo `getNotice.py` é usado para realizar consultas utilizando o GraphQL, ele recebe o epoch dado pelo usuário e envia uma query para o GraphQL, e seu resultado é, em seguida, usado para exibir uma resposta ao usuário.

Além disso, temos o arquivo `connection.py` que estabelece a conexão com o back-end da aplicação. E no diretório `static`, temos a pasta com as ABIs dos contratos de Input do Cartesi Rollups e a pasta `bootstrap-5.1.3-dist` com distribuições `bootstrap` para a estilização das páginas da aplicação.

4.6.4.2. Back-End

A Cartesi atualmente possui alta frequência de atualizações significativas de seu código-fonte e documentação, incluindo *scripts* para criação de modelo de Dapp com nome personalizado e respectivo conjunto de pastas e arquivos. Este e demais exemplos estão na url <https://github.com/cartesi/rollups-examples>. O LGPD Datasets está disponível em https://github.com/brunogondim/Dapp-Cartesi-sqlite_extended.

Em versões mais recentes do kit de desenvolvimento Cartesi é possível criar um Dapp modelo básico a partir da execução de um script shell. Entretanto, nada impede o desenvolvedor de seguir a partir de um dos exemplos já prontos, como foi este caso. Enquanto o desenvolvimento do Front-End permite maior liberdade ao desenvolvedor, o Back-End possui um conjunto de pastas e arquivos padrão.

Partindo da estrutura básica, onde o que muda de uma aplicação para a outra é basicamente o arquivo python, `sqlite.py`, dentro do diretório `server`, altera-se o código

python para atender o propósito da nova solução. Deve-se ressaltar aqui que, embora o exemplo seja sobre a linguagem de programação *python*, é possível teoricamente utilizar qualquer linguagem que rode sobre Linux.

Quando o servidor, seguindo a lógica contida no arquivo *.py*, recebe uma requisição *POST* na url terminada em *advance/*, executa o conteúdo da mensagem como comando SQL, abrindo uma conexão com o banco de dados *SQLite* para efetuar uma operação comum de banco de dados.

Caso os dados no banco de dados sejam alterados, eles ficarão salvos no estado final da Cartesi Machine, quando o processamento for finalizado. É esta evolução de estados da máquina que a permite funcionar como banco de dados persistente.

Caso seja uma mera consulta, o estado da máquina também será atualizado, pois ocorrem modificações no arranjo da memória e no processador, bem como a marca do tempo. Os dados consultados são transmitidos por uma requisição *POST* assíncrona enviada do servidor para a porta padrão de saída da Cartesi Machine, com a url contendo */notice* no final. Esta ação ocorre antes da resposta final à requisição inicial. O servidor encerra sua ação respondendo por *POST* a url de final */finish*.

Os dados estarão disponíveis para consulta na Blockchain em momento posterior à finalização da transação, ou seja, o recebimento dos dados depende de um novo comando.

O banco de dados

Para este exemplo, o banco de dados foi criado e populado fora da Cartesi Machine. Isto não é um requisito, mas caso o banco seja previamente criado, assim como qualquer outro eventual arquivo ou pasta que faça parte do back-end, deve ser referenciado nos arquivos Makefile e shell script, caso contrário, no momento da construção da máquina, eles não serão copiados para dentro da máquina. Dentro do diretório *server*, a linha `$(DAPP_FS_BIN): sqlite.py run.sh data.db` do arquivo *Makefile* deve ser alterada e a linha `cp ./data.db $DAPP_FS` do arquivo *build-dapp-fs.sh* deve ser incluída para que o script de criação da máquina considere os novos arquivos.

Um aspecto crucial para possibilitar utilização de banco de dados na Cartesi Machine é o tamanho do sistema de arquivos *.ext2* criado durante o comando *makefile*. Seu tamanho deve ser proporcional ao tamanho que se espera armazenar na máquina. Desta forma, o sistema de arquivos foi customizado para 16MB, alterando-se o parâmetro *-b* para o valor 16384 no arquivo *build-dapp-fs.sh* a linha:

```
1 genext2fs -f -i 512 -b 16384 -d $DAPP_FS $DAPP_FS_BIN
```

4.6.4.3. Interagindo com a aplicação

A interação com o Dapp pode ocorrer por linha de comando ou por seu Front-End. De preferência por este último.

O comando a seguir representa uma possível requisição à rede Blockchain, pelo HardHat, que por sua vez aciona o contrato referente ao Cartesi *on-chain*, que, em se-

guida, aciona a Cartesi Machine *off-chain*. É na parte *off-chain* que o núcleo do Dapp opera. Os comandos Hardhat devem ser acionados dentro do diretório **contracts**.

```
1 npx hardhat --network localhost sqlite:addInput --input "INSERT INTO
  Medical VALUES ('35', 'male', '32.4', '0', 'no', 'southeast',
  '10000.0000' )"

```

Neste Dapp conceitual, todas as operações possíveis se resumem a operações CRUD. Elas representam as ações dos titulares dos dados e dos controladores/operadores dos dados. O resultado das queries não são disponibilizadas na hora. Após a execução, deve-se aguardar alguns minutos e executar o comando abaixo, para retornar o resultado das queries.

```
1 npx hardhat --network localhost sqlite:getNotices --epoch 0 --payload
  string

```

O que vai disponibilizar os *datasets* em conformidade com a LGPD são os filtros sobre a tabela principal, com seu resultado salvo em novas tabelas específicas, que poderão ser usadas para treinar e validar modelos de aprendizado de máquina e outros tipos de processamento. A ciência do titular dos dados sobre quais dados estão sendo usados para cada propósito, praticamente em tempo real, bem como seu controle de exclusão e alteração são as principais vantagens dessa abordagem. A seguir um exemplo de filtro vertical. No primeiro comando cria-se a tabela. No segundo, ela é preenchida.

```
1 npx hardhat --network localhost sqlite:addInput --input "CREATE TABLE
  Vertical_Filter (age text, bmi text, charges text)"
2 npx hardhat --network localhost sqlite:addInput --input "INSERT INTO
  Vertical_Filter (age, bmi, charges) SELECT age, bmi, charges FROM
  Medical"

```

Lembrando que tais queries podem tanto ser comandadas pelo comando HardHat quanto pelo Front-End.

Para executar o front-end, com o back-end em operação, navegue até o diretório `front_end` e instale as dependências com o comando a seguir.

```
1 pip install -r requirements.txt

```

Prosseguindo, definimos a variável de ambiente `FLASK_APP` e rodamos o front-end. Para isso, execute os comandos abaixo.

```
1 export FLASK_APP=dapp
2 flask run

```

Agora, com o front-end em operação, vamos utilizá-lo para enviar os comando SQL para o back-end, ao invés de utilizarmos a linha de comando do terminal. Ele roda na página `localhost:5000`, acesse-a e use a barra de navegação para acessar a página `AddInput`. Nessa página, temos o campo de preenchimento `Statement`, que o usuário utiliza para inserir os comandos SQL. Por exemplo, ao interagirmos com a aplicação diretamente pela linha de comando, podemos usar o comando a seguir.

```
1 npx hardhat --network localhost sqlite:addInput --input "INSERT INTO
  Medical VALUES ('35', 'male', '32.4', '0', 'no', 'southeast',
  '10000.0000' )"

```

Utilizando o front-end, ao invés do comando acima, podemos simplesmente inserir o comando SQL a seguir no campo Statement da página AddInput.

```
1 INSERT INTO Medical VALUES ( '35', 'male', '32.4', '0', 'no', 'southeast', '10000.0000' )
```

Para obtermos os resultados das queries, pelo front-end, acessamos a página GetNotice e inserimos o epoch correspondente a notice que queremos no campo de preenchimento Epoch.

Limitações do Dapp

Embora a tecnologia Cartesi disponha de um sistema operacional Linux como base para a pilha tecnológica do contrato inteligente a ser desenvolvido, propiciando o uso de qualquer tecnologia que opere sobre o Linux, hoje ainda não é possível se utilizar de todo o arsenal tecnológico compatível com Linux na Cartesi de maneira prática. Cada componente que necessite de instalação deve estar presente no sistema de arquivos raiz previamente desenvolvido pela Cartesi. Atualmente a alteração do sistema de arquivos raiz da Cartesi Machine para ser utilizado com a solução de Roll-ups ainda não está disponível ao público, impossibilitando o uso de demais pacotes que precisam ser compilados para a arquitetura da Cartesi.

A obtenção dos dados consultados é realizada em uma operação posterior a solicitação. A solicitação altera o estado da máquina, mas a obtenção dos dados consultados não. Desta forma não é possível registrar dentro da máquina por quem e quando os dados são lidos.

Garantir a privacidade dos dados é um dos principais desafios de se utilizar este banco de dados distribuídos. Como todos os estados anteriores são inspecionáveis, eventuais chaves privadas salvas na máquina podem ser acessadas, mesmo se excluídas.

Como todos os dados inseridos na máquina ficarão inspecionáveis para sempre, os dados pessoais devem ser inseridos já anonimizados na Cartesi Machine, dificultando a confirmação da procedência e controle para posteriores alterações.

A efetiva garantia de que o controlador/operador dos dados irá respeitar a vontade do titular dos dados de ter seus dados alterados ou excluídos, ou apenas utilizados para um propósito específico previamente autorizado está além da capacidade de controle sobre a Blockchain e sua 2ª camada Cartesi. Ela depende de procedimentos extra-sistema. Mas esta limitação tecnológica não é exclusividade nem da tecnologia Blockchain nem da camada 2. Um dado, uma vez visto, não pode mais ser "desvisto". Qualquer dado que seja inserido em um sistema centralizado ou trafegue em rede pública pode nunca mais ser "esquecido", mesmo que seja excluído, pois pode ter sido copiado para outro sistema.

Possíveis soluções às limitações

Uma possível solução para garantir que os dados pessoais serão utilizados conforme acordado é implementar o treinamento do modelo e sua utilização operando dentro da própria Cartesi Machine. Teoricamente é viável treinar modelos dentro da máquina, já

que esta opera Linux, e Linux possui diversas bibliotecas para treinamento de modelos. A utilização efetiva dos modelos vai depender da dinâmica de cada negócio.

O desafio da autenticação do titular dos dados anonimizados para futuras alterações ou exclusões, e até mesmo para evitar duplicidade de registros, por quem de fato é o titular dos dados pode ser encarado utilizando identidades auto-soberanas. Embora atualmente haja diferentes noções do seu conceito e imprecisão sobre o alcance das suas aplicações, este tipo de identidade vem se mostrando cada vez mais uma forma promissora de interação entre as partes interessadas nos dados pessoais na internet do futuro [Ferdous et al. 2019].

4.6.5. IoT Rollups DApp

Como mencionado anteriormente, Blockchain é especialmente útil quando se envolve desconfiança mútua, por conta disso, seu uso é muito vantajoso no contexto de serviços públicos. O transporte público é um destes serviços que carece de confiança, basta se perguntar: como saber se o serviço está cumprindo com o cronograma acordado?

O cronograma mencionado diz respeito a rota que o veículo deve percorrer e o horário que ele deve estar em cada ponto de parada para pegar ou deixar passageiros. Portanto, esse cronograma pode ser visto como um contrato entre a prefeitura e a empresa que fornece o serviço. Para que seja possível verificar que as empresas estão fazendo a sua parte no acordo, ou seja, cumprindo o cronograma, os veículos devem ser capazes de fornecer dados geoespaciais (GPS).

Para que os veículos forneçam dados de GPS, eles devem estar equipados com dispositivos IoT (Internet of Things). Esses dispositivos, usualmente pequenos, são equipamentos que se conectam à internet e juntos formam um ecossistema no qual determinada aplicação roda fazendo uso deles de alguma forma. Eles geralmente funcionam como sensores ou atuadores. Como sensores, esses dispositivos têm o objetivo de gerar dados de medição, por exemplo, temperatura, umidade e proximidade. Como atuadores, esses dispositivos tem o objetivo de realizar mudanças de estado, por exemplo, abrir uma porta ou janela, quando solicitado. No contexto em questão, eles seriam sensores e forneceriam a coordenada do veículo.

Um fato relevante a ser destacado é que dispositivos IoT geralmente funcionam bem com Blockchain, devido a natureza descentralizada inerente ao uso de tais dispositivos, sendo assim, a criação de um Cartesi DApp que envolve dispositivos IoT é um passo natural. Desse modo, a aplicação proposta a seguir é um Cartesi DApp que faz uso de dispositivos IoT para solucionar o problema levantado a respeito do transporte público.

Visando solucionar o problema, o DApp a ser desenvolvido tem que ser capaz de verificar se o serviço de transporte público, nesse caso o ônibus, está cumprindo o seu cronograma. Caso seja comprovado que determinado veículo está descumprindo o cronograma, seja por desvio de rota ou por atraso, deve-se gerar uma multa. As multas geradas devem estar disponíveis em um dashboard online para que o público possa consultá-las.

Para simplificar a implementação, considera-se que os dispositivos que estão nos veículos fornecem dados confiáveis. A informação fornecida é composta pela coordenada do veículo acompanhada da timestamp do momento da medição. O DApp recebe essa

informação e no back-end consulta um banco de dados que possui o cronograma daquele veículo, utilizando essas duas informações é feita a verificação. As multas são geradas na forma de “/notice”, que posteriormente é consultado usando GraphQL para gerar o dashboard público. A Figura 4.16 contém um diagrama que representa o DApp e as subseções a seguir detalham o front-end, back-end e como interagir com a aplicação.

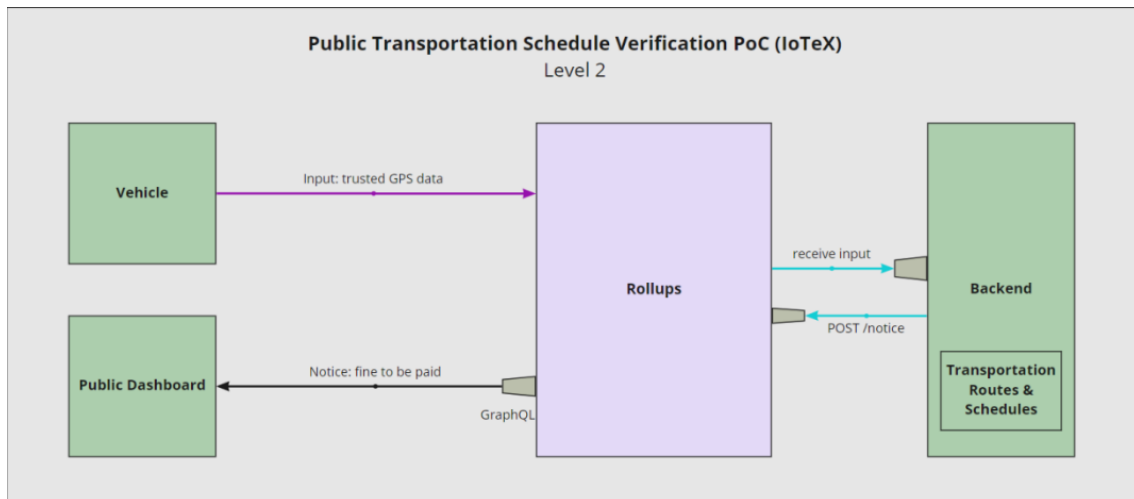


Figura 4.16. Arquitetura do Dapp

4.6.5.1. Front-End

Como mencionado anteriormente, a aplicação deve fornecer um dashboard para consulta do público, mas esse não é o único papel do front-end nesse DApp. O front-end é fornecido por um servidor web desenvolvido em NodeJS, e é responsável também por receber os inputs e repassá-los à Blockchain utilizando Web3, que é um novo padrão de interação na internet que se apoia em tecnologias descentralizadas como a Blockchain.

Para a construção do front-end a biblioteca Web3 utilizada foi a Web3.js, devido ao javascript ser a linguagem nativa do NodeJS. O servidor Web desenvolvido segue a arquitetura MVC, que é composta por 3 componentes (Model, View e Controller) que dão nome ao padrão. A seguir é explicado, de forma sucinta, o papel de cada componente.

- **Model:** É usado pelo controller para acessar o banco de dados. Contém a lógica de negócio, especificações de determinado dado e a lógica que será aplicada a ele.
- **View:** É por onde os usuários interagem com a aplicação, sendo composto apenas por templates HTML que são preenchidos dinamicamente de acordo com o dado que é passado para este template.
- **Controller:** Interage com o Model e fornece as respostas, oriundas de algum back-end, e funcionalidade para o View. O papel do controller é basicamente realizar o “meio de campo” entre o View, que é por onde o usuário interage com a aplicação, e o Model, que contém a lógica de negócio.

Cada componente corresponde a um diretório, como mostra a estrutura de diretórios abaixo, onde tem-se código javascript. Nessa estrutura observa-se também a presença de outros 3 diretórios. Em “config”, tem-se a configuração de acesso ao back-end da aplicação, em “routes” são definidas as rotas e em “public” temos assets usados no view, bibliotecas, arquivos estáticos, entre outros. Agora, tendo ciência deste padrão de design, visitaremos os componentes criados para o front-end utilizado neste DApp.

```
front_end
├── config
│   └── eth-connection.js
├── controllers
│   └── home.js
├── models
│   └── blockchain-model.js
├── public
│   ├── ABI
│   ├── assets
│   └── bootstrap-5.1.3-dist
├── routes
│   └── home.js
├── views
│   ├── form.ejs
│   └── index.ejs
└── app.js
```

Model

No model temos o “blockchain-model.js”, onde são definidas 3 funções:

- `getNoticePage`: Realiza consultas usando graphql para o container “query_server”, com o objetivo de recuperar as multas geradas (/notices), o resultado é posteriormente exibido no dashboard.
- `getAccounts`: Utilizando o módulo de conexão à Blockchain Hardhat definido em “config”, realiza uma consulta para recuperar as contas existentes.
- `addInput`: Utilizando o módulo de conexão à Blockchain Hardhat definido em “config”, executa o método `addInput` do contrato de input para enviar o input para o back-end do DApp.

View

No View temos os dois templates HTML, “index.ejs” e “form.ejs”. O “index.ejs” é a página que possui o Dashboard para a consulta das multas, o “form.ejs” é a página por onde pode-se enviar o cronograma de uma linha de ônibus para o back-end do DApp.

Controller

No controller temos o “home.js”, onde são definidas 3 funções:

- `homePage`: Usando a função `getNoticePage` do model, recupera os dados e encaminha para a view “`index.ejs`”. É acessado através da rota “/” e renderiza o dashboard.
- `formPage`: Usando a função `getAccounts` do model, recupera os dados e encaminha para a view “`form.ejs`”. É acessado através da rota “/form” e renderiza a página do formulário.
- `submit`: Executa a função `addInput` do model. É acessado através da rota “/submit” e envia um JSON de resposta para o cliente que acessou a rota informando se o `addInput` foi executado com sucesso ou não.

4.6.5.2. Back-End

O back-end da aplicação, que é o código executado dentro da Cartesi Machine em Production Mode, foi desenvolvido em Python e seus arquivos encontram-se no diretório *server*. O back-end recebe dois tipos de input, um deles é o cronograma de uma linha de ônibus e o outro é o dado enviado pelo dispositivo IoT em um veículo. Dependendo do input recebido, o back-end deverá processá-lo de forma diferente.

Caso o input recebido seja um cronograma de uma linha de ônibus, o processamento consiste em salvar o cronograma em um banco de dados SQLite do DApp. Este input possui o identificador da linha de ônibus, a rota que é feita por ele, a coordenada das paradas e o cronograma de cada viagem que aquela linha faz. Vale lembrar também que se o back-end estiver sendo executado em Production Mode, o banco de dados ficará dentro da Cartesi Machine.

Caso o input recebido seja um dado enviado por um veículo, o processamento consiste em a partir do identificador da viagem, que também é fornecida pelo veículo, consultar o banco de dados de cronogramas para recuperar a rota e o cronograma daquela viagem. Com os dados recebidos pelo input e os dados recuperados do banco é possível verificar se o veículo está na rota e se o veículo está atrasado ou não.

O código Python foi dividido em três módulos que têm seus nomes e funcionalidades mostrados a seguir.

- `db_manager.py`: É o módulo que gerencia a conexão ao banco de dados SQLite, é responsável pela criação das tabelas utilizadas, caso não existam, inserções e consultas diversas.
- `util.py`: É o módulo que contém funções de cálculo e conversão. Conversão de string para hexadecimal, cálculo da distância entre duas coordenadas e funções que fazem uso do cálculo de distância para verificar a rota e o atraso.
- `iot_dapp.py`: É a main do back-end e importa os outros dois módulos. Seu papel é receber os inputs verificando qual é o tipo, um cronograma ou um dado enviado por um veículo, em seguida utilizar as funções dos outros dois módulos para processá-lo corretamente.

O fluxograma da Figura 4.17 resume o fluxo do processamento realizado pelo back-end.

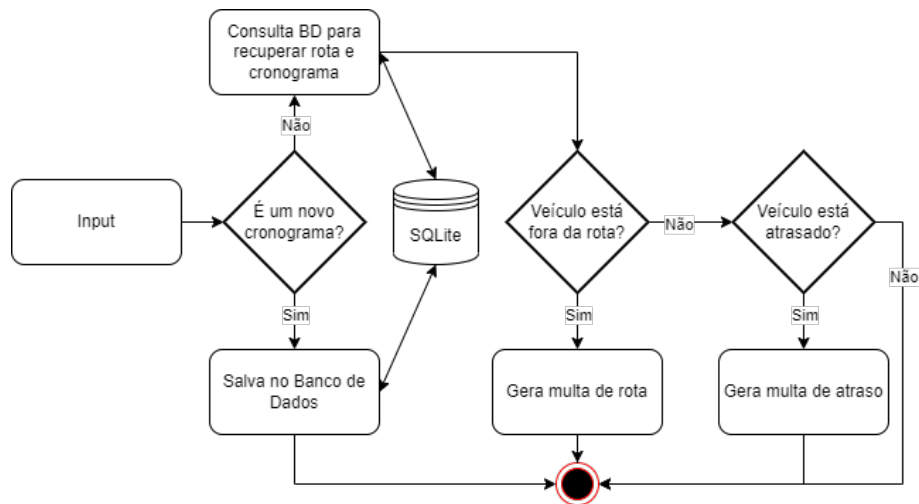


Figura 4.17. Fluxograma do Back-End

4.6.5.3. Interagindo com a aplicação

Antes de interagir com a aplicação, precisamos executá-la e isso envolve rodar o back-end e o front-end. Como já foi visto anteriormente o procedimento para executar o back-end, veremos nesta subseção apenas como executar o front-end antes de interagir com a aplicação. Para o front-end deste DApp é necessário executar os seguintes passos:

1. Navegar até o diretório *front_end*;
2. Instalar as dependências utilizadas executando o comando **npm install**;
3. Rodar o back-end em Host Mode ou Production Mode em outro terminal;
4. Executar o front-end após o back-end estar rodando, utilizando o comando **node app.js**.

Uma vez cumpridas todas as etapas, o DApp estará rodando e o front-end pode ser acessado em `http://localhost:3000`, nessa página inicial temos o dashboard, que no momento deve estar vazio. Vimos na seção back-end que o DApp trabalha com dois tipos de input, o cronograma e o dado enviado por um veículo. Sabendo que não faz sentido processar o dado de um veículo que não possui um cronograma cadastrado, é mostrado primeiro como cadastrar um cronograma e somente depois como enviar o dado de um veículo.

Cadastrando um cronograma

Para cadastrar um cronograma devemos acessar a página de formulário fornecido pelo front-end, para acessá-la basta clicar no link “form” na barra de navegação ou acessar `http://localhost:3000/form`. Uma vez na página de formulário, escolha o endereço de onde a transação deve ser enviada, em seguida carregue um arquivo que contenha o cronograma desejado, o diretório `data_demo` possui dois arquivos de exemplo, por fim clique em submit.

Enviando dados de um veículo

Para o envio de dados de um veículo não existe uma página com interface para isso, apenas uma rota para a qual deve-se enviar tais dados, a rota é `http://localhost:3000/submit`. Portanto, deve-se fazer um POST para essa rota, tendo no corpo da requisição um JSON com os dados. Abaixo temos exemplos de envios usando o comando `curl`.

```
1 curl -H "Content-Type: application/json" -d '{
2 "fromAddress": "0x14dC79964da2C08b23698B3D3cc7Ca32193d9955",
3 "data": { "bus_id": "18C", "trip_id": "18C;1",
4 "lat": 57.828261, "lon": 26.535419,
5 "ts": "2022-03-25 07:45:50" }
6 }'
7 http://localhost:3000/submit
```

Listing 4.1. Enviando dados da linha 18C (Veículo fora da rota)

```
1 curl -H "Content-Type: application/json" -d '{
2 "fromAddress": "0x14dC79964da2C08b23698B3D3cc7Ca32193d9955",
3 "data": { "bus_id": "18C", "trip_id": "18C;1",
4 "lat": 57.82847892, "lon": 26.53362055,
5 "ts": "2022-05-04 07:48:30" }
6 }'
7 http://localhost:3000/submit
```

Listing 4.2. Enviando dados da linha 18C (Veículo atrasado)

Caso deseje-se enviar outros dados, é importante ressaltar que o JSON enviado no corpo da requisição deve possuir os seguintes campos:

- `fromAddress`: O endereço de onde a transação deve ser enviada
- `data`: O payload que será processado pelo back-end do DApp
 - `bus_id`: Identificador da linha de ônibus.
 - `trip_id`: Identificador da viagem, é o `bus_id` concatenado com o inteiro correspondente ao número da viagem.
 - `ts`: momento da medição, está no formato `%Y-%m-%d %H:%M:%S`
 - `lat`: latitude medida no momento "ts".
 - `lon`: longitude medida no momento "ts".

4.7. Conclusões e considerações finais

Nesta seção iremos revisitar diversos dos conceitos vistos até o momento, procurando ressaltar os pontos mais importantes a respeito dos mesmos, seguindo a ordem em que aparecem no texto.

4.7.1. Blockchain

Podemos resumir Blockchain como um livro razão digital, ou seja, uma ferramenta de auditabilidade que veio para oferecer segurança e transparência em transações envolvendo duas partes que não confiam uma na outra, sem a necessidade de uma terceira parte confiável. Sobre essa definição destacam-se os termos “transparência” e “transação”, a Blockchain é capaz de oferecer transparência porque é auditável, mas ao mesmo tempo mantém um certo grau de privacidade, pois uma pessoa ou empresa não precisa se identificar ao criar uma carteira no Bitcoin, por exemplo, e ainda pode criar quantas quiser. Já o termo transação, é genérico e pode assumir diversos papéis, pode simplesmente representar uma transação monetária, ou a execução de um complexo método de um contrato inteligente com diversas transações internas. Um ponto a ser destacado é que existe uma terceira parte confiável na Blockchain, mas esse papel não cabe a um indivíduo ou entidade, a rede como um todo realiza esse papel através de algoritmos de consenso.

Mas a Blockchain, tal como qualquer sistema distribuído, não consegue alcançar os três vértices do triângulo de tradeoffs exibido na Seção 4.4. E como a Blockchain envolve criptoativos na forma de tokens, as soluções não desejam abrir mão da descentralização, visando evitar que determinada entidade seja capaz de manipular tais ativos, e nem da segurança, este por motivos ainda mais óbvios. Sendo assim, a primeira limitação que nos deparamos na tecnologia Blockchain é a escalabilidade, vale ressaltar que isso ocorre apenas para Blockchains públicas, como o Ethereum e o Bitcoin, uma vez que uma Blockchain privada está abrindo mão da descentralização ela deve ser capaz de obter um desempenho melhor nos outros dois pontos.

4.7.2. Escalabilidade

Por conta da dificuldade mencionada em abrir mão da descentralização e da segurança, os olhos e esforços dos desenvolvedores daqueles que contribuem com a tecnologia Blockchain voltaram para a escalabilidade. No texto destacamos duas das principais linhas de proposta nesse segmento, Sharding e Rollups. Porém, o sharding acarreta em alguns desafios difíceis de serem resolvidos de forma ótima:

- Distribuir os nós nos Shards de forma uniforme e imprevisível, para evitar que um shard fique com poucos nós e também evitar que algum atacante mal intencionado preveja os nós que compõem os Shards.
- Criar um controle de identidade com o objetivo de impedir ataques sybil.
- No caso do Complete Sharding também surge a necessidade de executar algoritmos de consenso intra-shard e inter-shard, tendo como objetivo manter a coerência das transações entre shards.

Dado estes pontos, o Rollups se mostra uma boa alternativa devido a sua premissa de acumular as transações em lotes e não gerar problemas com o nível de complexidade

daqueles presentes no sharding.

4.7.3. DApps

O problema de escalabilidade mencionado por si só acaba inviabilizando diversos tipos de DApps que envolvem a necessidade de um tempo de resposta rápido ou que devem atender um número de usuários muito grande.

Outro fator que muitas das vezes inviabiliza a criação de determinados DApps é o custo, o custo de implantação e execução dos contratos inteligentes também inviabiliza a criação de DApps que envolvem uma computação muito complexa.

Como DApps são constituídos de um ou mais contratos inteligentes, é importante ressaltar a dificuldade de desenvolvê-los, uma vez que possuem linguagem própria, variando de Blockchain para Blockchain, como visto para o Ethereum e o Bitcoin, e ainda a ausência de um ambiente de desenvolvimento adequado com debuggers e outras funcionalidades.

4.7.4. Cartesi Rollups

A estratégia de Optimistic Rollups implementada pela Cartesi aborda as questões de escalabilidade e custo. As transações podem ser processadas em uma taxa maior fora da Mainnet e custo de transação e diluído entre todas as transações do lote. A segurança da Cartesi Rollups é garantida por fraud proofs, as transações são confirmadas se não forem contestadas. A Cartesi Machine oferece computação verificável, duas execuções com os mesmas entradas geram sempre o mesmo resultado.

Além de habilitar a verificação do fraud proofs a Cartesi Machine possui uma arquitetura RISC-V com sistema operacional Linux. Ao mover a maior parte de sua lógica DApp para rodar dentro de Máquinas Cartesi, os desenvolvedores podem usar as linguagens e ferramentas de sua escolha.

Assim, podemos resumir o Cartesi Rollups como uma solução para o problema de escalabilidade das Blockchains, enquanto oferece um ambiente familiar para o desenvolvimento de DApps.

Referências

- [Argento 2021] Argento, F. (2021). Rollups: On-chain. <https://medium.com/cartesi/rollups-on-chain-d749744a9cb3>.
- [Benet 2014] Benet, J. (2014). Ipfs - content addressed, versioned, p2p file system.
- [Binance 2020] Binance, A. (2020). What is ethereum plasma? <https://academy.binance.com/en/articles/what-is-ethereum-plasma>. Accessed on 2022-05.
- [bit2me 2020] bit2me (2020). O que é sharding? <https://academy.bit2me.com/pt/o-que-é-sharding/>.
- [Buterin 2021] Buterin, V. (2021). An incomplete guide to rollups. <https://vitalik.ca/general/2021/01/05/rollup.html>. Accessed on 2022-04.

- [Cartesi 2022] Cartesi (2022). Cartesi Documentation. <https://www.cartesi.io/en/docs/>.
- [Corso 2019] Corso, A. (2019). *Performance analysis of proof-of-elapsed-time (poet) consensus in the sawtooth blockchain framework*. PhD thesis, University of Oregon.
- [Daudén-Esmel et al. 2021] Daudén-Esmel, C., Castellà-Roca, J., Viejo, A., and Domingo-Ferrer, J. (2021). Lightweight blockchain-based platform for gdpr-compliant personal data management. In *2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP)*, pages 68–73.
- [Douceur 2002] Douceur, J. R. (2002). The sybil attack. In *International workshop on peer-to-peer systems*, pages 251–260. Springer.
- [Eichmann 2018] Eichmann, K. (2018). The future client of an energy utility company will be a machine. <https://medium.com/future-energy-ventures/machine-economy-a-decentralized-future-that-is-enabled-by-autonomous-machine-to-machine-e497b90f13c1>.
- [Ferdous et al. 2019] Ferdous, M. S., Chowdhury, F., and Alassafi, M. O. (2019). In search of self-sovereign identity leveraging blockchain technology. *IEEE Access*, 7:103059–103079.
- [Gentry 2009] Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. In *STOC '09*.
- [Governo Federal 2021] Governo Federal, M. d. E. (2021). Guia de requisitos mínimos de segurança e privacidade para aplicativos móveis - lgpd. https://www.gov.br/governodigital/pt-br/seguranca-e-protecao-de-dados/guias/guia_seguranca_apps.pdf.
- [Hong et al. 2021] Hong, Z., Guo, S., Li, P., and Chen, W. (2021). Pyramid: A layered sharding blockchain system. *IEEE INFOCOM*.
- [Jake Frankenfield 2020] Jake Frankenfield, S. G. A. (2020). Proof of elapsed time (poet) (cryptocurrency). <https://www.investopedia.com/terms/p/proof-elapsed-time-cryptocurrency.asp>.
- [Johnston et al. 2014] Johnston, D., Yilmaz, S. O., Kandah, J., Bentenitis, N., Hashemi, F., Gross, R., and Mason, S. (2014). The general theory of decentralized applications. *DApps*, URL-<https://cryptochainuni.com/wp-content/uploads/The-General-Theory-of-Decentralized-Applications-DApps.pdf>.
- [King and Nadal 2012] King, S. and Nadal, S. (2012). Ppcoin: Peer-to-peer crypto-currency with proof-of-stake. *self-published paper*, August, 19:1.
- [L. 2019] L., K. (2019). The blockchain scalability problem & the race for visa-like transaction speed. <https://towardsdatascience.com/the-blockchain-scalability-problem-the-race-for-visa-like-transaction-speed-5cce48f9d44>.
- [Lerner 2015] Lerner, S. D. (2015). Rsk. Technical report, Tech. Rep.
- [Maxwell et al. 2019] Maxwell, G., Poelstra, A., Seurin, Y., and Wuille, P. (2019). Simple schnorr multi-signatures with applications to bitcoin. *Designs, Codes and Cryptography*, 87(9):2139–2164.

- [Mearian 2019] Mearian, L. (2019). Sharding: What it is and why many blockchain protocols rely on it. <https://www.computerworld.com/article/3336187/sharding-what-it-is-and-why-so-many-blockchain-protocols-rely-on-it.html>.
- [Moura 2021] Moura, E. (2021). Cartesi rollups. <https://medium.com/cartesi/scalable-smart-contracts-on-ethereum-built-with-mainstream-software-stacks-8ad6f8f17997>. Accessed on 2022-02.
- [Muratov et al. 2018] Muratov, F., Lebedev, A., Iushkevich, N., Nasrulin, B., and Takemiya, M. (2018). Yac: Bft consensus algorithm for blockchain. *arXiv preprint arXiv:1809.00554*.
- [Nakamoto 2008] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. *Decentralized Business Review*, page 21260.
- [OmniLayer 2020] OmniLayer, C. (2020). Omnilayer documentation. <https://github.com/OmniLayer/spec/blob/master/OmniSpecification-v0.6.adoc>.
- [Pahlajani et al. 2019] Pahlajani, S., Kshirsagar, A., and Pachghare, V. (2019). Survey on private blockchain consensus algorithms. In *2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT)*, pages 1–6. IEEE.
- [Rebello et al. 2019] Rebello, G., Camilo, G., Silva, L., Souza, L., Guimarães, L., Alchieri, E., Greve, F., and Duarte, O. (2019). Correntes de blocos: Algoritmos de consenso e implementação na plataforma hyperledger fabric. *Sociedade Brasileira de Computação*.
- [Revoredo 2019] Revoredo, T. (2019). Blockchain: tudo que você precisa saber (potencial e realidade).
- [Shor 1994] Shor, P. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134.
- [Shor 1997] Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509.
- [Stach. et al. 2020] Stach., C., Giebler., C., Wagner., M., Weber., C., and Mitschang., B. (2020). Amnesia: A technical solution towards gdpr-compliant machine learning. In *Proceedings of the 6th International Conference on Information Systems Security and Privacy - ICISSP*, pages 21–32. INSTICC, SciTePress.
- [Szabo 1996] Szabo, N. (1996). Smart contracts: building blocks for digital markets. *EXTROPY: The Journal of Transhumanist Thought*, (16), 18(2):28.
- [Sá 2022] Sá, C. (2022). Escalabilidade para ethereum com rollups. <https://goblockchain.io/escalabilidade-para-ethereum-com-rollups/>. Accessed on 2022-05.
- [Teixeira and Nehab 2018] Teixeira, A. and Nehab, D. (2018). The core of cartesi. Whitepaper, Cartesi.

- [Tianchen et al. 2021] Tianchen, W., Miaoyan, X., Han, C., and Yuming, Y. (2021). An in-depth look at rollup's tech, application, and data. <https://medium.com/huobi-research/an-in-depth-look-at-rollups-tech-application-and-data-eb8f91bf369b>. Accessed on 2022-03.
- [Truong et al. 2020] Truong, N. B., Sun, K., Lee, G. M., and Guo, Y. (2020). Gdpr-compliant personal data management: A blockchain-based solution. *IEEE Transactions on Information Forensics and Security*, 15:1746–1761.
- [Wang et al. 2019] Wang, G., Shi, Z. J., Nixon, M., and Han, S. (2019). Sok: Sharding on blockchain. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, pages 41–61.
- [Wood 2014] Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 151.
- [Wu et al. 2021] Wu, K., Ma, Y., Huang, G., and Liu, X. (2021). A first look at blockchain-based decentralized applications. *Software: Practice and Experience*, 51(10):2033–2050.
- [Zhou et al. 2020a] Zhou, Q., Huang, H., Zheng, Z., and Bian, J. (2020a). Solutions to scalability of blockchain: A survey. *IEEE Access*, 8:16440–16455.
- [Zhou et al. 2020b] Zhou, Q., Huang, H., Zheng, Z., and Bian, J. (2020b). Solutions to scalability of blockchain: A survey. *IEEE Access*, 8:16440–16455.