



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**ARTHUR ALMEIDA ALVES**

**DESAFIOS NA COLABORAÇÃO ENTRE ENGENHEIROS DE  
SOFTWARE E CIENTISTAS DE DADOS EM PROJETOS DE MACHINE  
LEARNING**

**CAMPINA GRANDE - PB**

**2024**

**ARTHUR ALMEIDA ALVES**

**DESAFIOS NA COLABORAÇÃO ENTRE ENGENHEIROS DE  
SOFTWARE E CIENTISTAS DE DADOS EM PROJETOS DE MACHINE  
LEARNING**

**Trabalho de Conclusão de Curso  
apresentado ao Curso Bacharelado em  
Ciência da Computação do Centro de  
Engenharia Elétrica e Informática da  
Universidade Federal de Campina  
Grande, como requisito parcial para  
obtenção do título de Bacharel em  
Ciência da Computação.**

**Orientador: Franklin de Souza Ramalho**

**CAMPINA GRANDE - PB**

**2024**

**ARTHUR ALMEIDA ALVES**

**DESAFIOS NA COLABORAÇÃO ENTRE ENGENHEIROS DE  
SOFTWARE E CIENTISTAS DE DADOS EM PROJETOS DE MACHINE  
LEARNING**

**Trabalho de Conclusão de Curso  
apresentado ao Curso Bacharelado em  
Ciência da Computação do Centro de  
Engenharia Elétrica e Informática da  
Universidade Federal de Campina  
Grande, como requisito parcial para  
obtenção do título de Bacharel em  
Ciência da Computação.**

**BANCA EXAMINADORA:**

**Franklin de Souza Ramalho**

**Orientador – UASC/CEEI/UFCG**

**Hyggo Oliveira de Almeida**

**Examinador – UASC/CEEI/UFCG**

**Francisco Vilar Brasileiro**

**Professor da Disciplina TCC – UASC/CEEI/UFCG**

**Trabalho aprovado em: 15 de Maio de 2024.**

**CAMPINA GRANDE - PB**

## **RESUMO**

Recentes avanços em Machine Learning (ML) têm despertado considerável interesse na integração de capacidades de IA em software e serviços, tornando a colaboração entre cientistas de dados e engenheiros de software crucial, porém desafiadora. Este estudo investiga os desafios encontrados nessa colaboração em projetos de ML. Através de entrevistas com profissionais do campo, identificamos questões cruciais, como lacunas de conhecimento entre disciplinas, adaptações de práticas de Engenharia de Software (ES) para ML e dificuldades na avaliação de modelos. Destacamos a importância do envolvimento precoce de cientistas de dados na definição de requisitos do software, contribuindo para o desenvolvimento bem-sucedido de sistemas de ML. Este estudo oferece valiosos insights para equipes que enfrentam desafios similares na implementação de ML.

# **CHALLENGES IN COLLABORATION BETWEEN SOFTWARE ENGINEERS AND DATA SCIENTISTS IN MACHINE LEARNING PROJECTS**

## **ABSTRACT**

Recent advances in Machine Learning (ML) have sparked considerable interest in integrating AI capabilities into software and services, making collaboration between data scientists and software engineers crucial yet challenging. This study investigates the challenges encountered in this collaboration in ML projects. Through interviews with professionals in the field, we identify critical issues such as knowledge gaps between disciplines, adaptations of Software Engineering (SE) practices for ML, and difficulties in model evaluation. We highlight the importance of early involvement of data scientists in defining software requirements, contributing to the successful development of ML systems. This study provides valuable insights for teams facing similar challenges in ML implementation.

# Desafios na Colaboração entre Engenheiros de Software e Cientistas de Dados em Projetos de Machine Learning

Arthur Almeida Alves  
arthur.alves@ccc.ufcg.edu.br  
Universidade Federal de Campina Grande  
Campina Grande, Paraíba, Brasil

Franklin de Souza Ramalho  
franklin@computacao.ufcg.edu.br  
Universidade Federal de Campina Grande  
Campina Grande, Paraíba, Brasil

## Resumo

Recentes avanços em Machine Learning (ML) têm despertado considerável interesse na integração de capacidades de IA em software e serviços, tornando a colaboração entre cientistas de dados e engenheiros de software crucial, porém desafiadora. Este estudo investiga os desafios encontrados nessa colaboração em projetos de ML. Através de entrevistas com profissionais do campo, identificamos questões cruciais, como lacunas de conhecimento entre disciplinas, adaptações de práticas de Engenharia de Software (ES) para ML e dificuldades na avaliação de modelos. Destacamos a importância do envolvimento precoce de cientistas de dados na definição de requisitos do software, contribuindo para o desenvolvimento bem-sucedido de sistemas de ML. Este estudo oferece valiosos insights para equipes que enfrentam desafios similares na implementação de ML.

## Keywords

Machine Learning, Cientista de Dados, Engenheiro de Software, Desafios na Colaboração

## 1 INTRODUÇÃO

Inteligência artificial (IA) é um campo amplo que se refere ao uso de tecnologias para criar máquinas e computadores capazes de imitar funções cognitivas associadas à inteligência humana. Isso inclui capacidades como a compreensão e resposta à linguagem, análise de dados e recomendações. Embora seja considerada um sistema em si, a IA é na verdade um conjunto de tecnologias implementadas para permitir que máquinas ponderem, aprendam e ajam para resolver problemas complexos [3].

O Machine Learning, um subcampo da IA, evoluiu significativamente com o tempo. Ele nasceu da teoria de que computadores podem aprender com dados e reconhecer padrões sem serem explicitamente programados. A capacidade de se adaptar a novos dados é fundamental no processo iterativo de ML, permitindo que os modelos produzam decisões e resultados confiáveis. Embora não seja uma ciência nova, está ganhando um novo impulso [10].

As técnicas de ML permitem criar modelos que identificam padrões e fazem previsões a partir de dados, automatizando processos e melhorando a tomada de decisões [5]. Essas técnicas são aplicadas de várias maneiras, incluindo algoritmos supervisionados, não supervisionados e por reforço, para analisar grandes volumes de dados[9].

Engenharia de Software é a aplicação de princípios de engenharia para o desenvolvimento e gerenciamento de sistemas de software. Planejar, desenvolver e manter software é uma tarefa complexa, especialmente em empresas já consolidadas, que envolve uma série de atividades antes, durante e após o desenvolvimento de um projeto [4].

MLOps, abreviação para Machine Learning Operations, é uma abordagem específica para organizar e gerenciar a implantação de modelos de ML em produção. Em essência, o MLOps automatiza e padroniza a criação, manutenção e gerenciamento do ciclo de vida desses modelos, garantindo que permaneçam em boa qualidade e ofereçam sempre bom desempenho. Essa área de atuação conecta ML, engenharia de dados e DevOps (Development Operations), unificando o desenvolvimento e as operações no contexto de ML [1].

Este estudo visa investigar os desafios encontrados na colaboração entre cientistas de dados e engenheiros de software em projetos de ML. Reconhecendo a importância dessa cooperação para o desenvolvimento bem-sucedido de sistemas de ML, buscamos identificar e analisar as questões cruciais enfrentadas por essas equipes multidisciplinares. É crucial entender esses desafios para melhorar a cooperação entre as equipes e garantir o sucesso dos projetos de ML [8].

A metodologia adotada neste estudo envolveu a realização de entrevistas qualitativas com os participantes selecionados, seguidas pela transcrição e análise dos dados coletados. Além disso, utilizamos o software MAXQDA [7] para a análise dos resultados obtidos. O objetivo da avaliação foi obter *insights* significativos que possam contribuir para pesquisas futuras e para a formulação de recomendações práticas que abordem os desafios identificados. Essa abordagem metodológica foi implementada visando assegurar a solidez e a pertinência dos resultados alcançados, estabelecendo uma base consistente para as conclusões e discussões apresentadas ao longo deste estudo.

A sinergia entre as áreas de IA e ML já vem sendo explorada por alguns estudos. No estudo de Nahar et al. [8] identifica áreas críticas de desafios na colaboração entre essas equipes, destacando a importância da comunicação, documentação, engenharia e processo para o sucesso dos projetos de sistemas habilitados para ML. Ludemir [6] enfatiza a necessidade de modelos de ML explicáveis e a importância de conjuntos de dados representativos.

O estudo de Teixeira [11] investiga a caracterização da comunidade *open source* de MLOps e suas implicações na entrega contínua de sistemas de ML. As informações fornecidas sobre práticas e ferramentas disponíveis nos ajudam a compreender os desafios e oportunidades enfrentados por equipes que colaboram em projetos de ML, alinhando-se aos objetivos de nossa pesquisa.

## 1.1 Objetivos

Nesta subseção, apresentamos os objetivos deste estudo, delineando as metas específicas que orientaram nossa pesquisa. A colaboração entre cientistas de dados e engenheiros de software em projetos de ML é essencial para o sucesso de sistemas complexos. No entanto, essa colaboração pode ser prejudicada por uma série de desafios, desde lacunas de conhecimento até conflitos de interesse. Portanto, os objetivos deste estudo são:

- Identificar os principais desafios na colaboração entre cientistas de dados e engenheiros de software em projetos de ML.
- Investigar a natureza e a origem dos conflitos entre as disciplinas de ciência de dados e Engenharia de Software, visando compreender as razões por trás da falta de alinhamento e coordenação.
- Avaliar o impacto dos desafios identificados na eficiência, confiabilidade e adaptabilidade dos sistemas de ML desenvolvidos, analisando como esses desafios afetaram o ciclo de vida do desenvolvimento de software.
- Examinar estudos de caso ou exemplos práticos nos quais os desafios na colaboração entre cientistas de dados e engenheiros de software foram superados com sucesso, examinando as estratégias e abordagens utilizadas para resolver conflitos e promover uma colaboração mais eficaz.
- Gerar *insights* que possam contribuir para pesquisas futuras na proposição de recomendações práticas para lidar com os desafios identificados. Isso inclui a análise crítica dos resultados obtidos e a identificação de padrões ou lacunas que possam orientar estudos subsequentes.

## 1.2 Organização do Documento

Este documento está estruturado da seguinte forma: a Seção 2 apresenta a Fundamentação Teórica, abordando os conceitos fundamentais relacionados à colaboração entre cientistas de dados e engenheiros de software em projetos de ML. A Seção 3 apresenta os Trabalhos Relacionados, contextualizando este estudo dentro do corpo de pesquisa existente, destacando estudos semelhantes, abordagens metodológicas comparáveis e contribuições relevantes para o campo. A Metodologia utilizada é detalhada na Seção 4, incluindo a descrição dos procedimentos de entrevistas realizadas, transcrição e análise dos dados utilizando o software MAXQDA<sup>1</sup>. Resultados e Discussões sobre a análise são apresentados na Seção 5. As Conclusões do estudo são apresentadas na Seção 6, seguidas pelos Agradecimentos na Seção 7.

## 2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, será apresentada a fundamentação teórica que sustenta a pesquisa, destacando o método da teoria fundamentada como uma abordagem essencial para desenvolver análises teóricas em pesquisa científica. Será discutida a importância da imersão nos dados e da constante comparação entre eles para identificar padrões, relações e conceitos emergentes. Além disso, será apresentado o MAXQDA como uma ferramenta valiosa para apoiar o processo de análise,

fornecendo recursos essenciais, como codificação, organização hierárquica, análise flexível, visualização de dados e integração de fontes. Por meio da combinação do método da teoria fundamentada com o suporte oferecido pelo MAXQDA, esta pesquisa busca construir teorias empiricamente fundamentadas e contextualmente relevantes, contribuindo para o avanço do conhecimento em sua área de estudo.

### 2.1 Teoria Fundamentada

O método da teoria fundamentada oferece estratégias úteis para desenvolver análises teóricas para uma pesquisa científica. Esse método ajuda a gerar novos conceitos em sua disciplina e na literatura de pesquisa mais ampla. Esses conceitos podem ter aplicação direta em políticas e práticas profissionais em psicologia e além dela [2].

A abordagem da teoria fundamentada destaca a importância da imersão nos dados e da constante comparação entre eles para identificar padrões, relações e conceitos emergentes. Os pesquisadores são incentivados a adotar uma postura reflexiva e aberta durante todo o processo, permitindo que as teorias surjam organicamente a partir dos próprios dados coletados. A flexibilidade inerente ao método da teoria fundamentada permite que os pesquisadores respondam de maneira dinâmica às nuances e complexidades dos dados, resultando em teorias mais robustas e contextualmente relevantes. A utilização desse método aliada a ferramentas proporciona aos pesquisadores uma estrutura sólida para a análise qualitativa e a construção de teorias fundamentadas empiricamente [7].

### 2.2 MAXQDA como ferramenta auxiliar

O MAXQDA é uma ferramenta valiosa para apoiar a teoria fundamentada em várias etapas do processo de análise. Suas capacidades de codificação, organização hierárquica, análise flexível, visualização de dados e integração de fontes oferecem suporte significativo para os pesquisadores durante todo o processo [7].

#### 2.2.1 Facilidade de Codificação

O MAXQDA oferece uma variedade de opções de codificação, incluindo a capacidade de codificar por cores, destacando segmentos de texto como faria com um marcador em um livro físico. Isso simplifica o processo de identificar e categorizar temas emergentes nos seus dados.

#### 2.2.2 Organização Hierárquica

Com o MAXQDA, você pode organizar seus códigos em uma estrutura hierárquica, facilitando a visualização das relações entre diferentes conceitos e a criação de uma árvore de códigos.

#### 2.2.3 Flexibilidade na Análise

O MAXQDA permite uma análise flexível dos seus dados, adaptando-se às nuances e complexidades dos seus dados. Isso é essencial na teoria fundamentada, onde a ênfase está na imersão nos dados e na identificação de padrões emergentes.

<sup>1</sup>MAXQDA 2024. Recuperado de <https://www.maxqda.com/>

### 2.2.4 Visualização de Dados

A Codificação Criativa do MAXQDA permite criar diagramas visuais das suas teorias, ajudando a entender e comunicar os relacionamentos entre diferentes conceitos de uma forma clara e acessível.

### 2.2.5 Integração de Dados

Todas as suas análises no MAXQDA são integradas ao seu projeto, o que facilita a referência cruzada e o acompanhamento das fontes de dados ao longo do processo de análise.

Em resumo, o MAXQDA oferece uma variedade de recursos e ferramentas que tornam o processo de análise da teoria fundamentada mais eficiente e eficaz, permitindo que você se concentre na construção de teorias sólidas a partir dos seus dados.

## 2.3 Tipos de Codificação

Os tipos de codificação, como codificação in-vivo, aberta e seletiva, trabalham em conjunto para fornecer uma estrutura analítica robusta. A codificação in-vivo preserva a autenticidade dos dados, enquanto a codificação aberta identifica conceitos específicos. Posteriormente, a codificação seletiva organiza os códigos em categorias mais amplas, capturando a complexidade dos fenômenos estudados. As figuras foram elaboradas utilizando o software MAXQDA.

### 2.3.1 Codificação In-Vivo

Durante todo o processo de codificação, os pesquisadores podem optar por utilizar a codificação in-vivo, que envolve atribuir códigos diretamente com base nas palavras ou frases encontradas nos dados. Isso ajuda a manter os códigos ancorados no material original, preservando a autenticidade dos dados e facilitando a identificação de padrões e temas emergentes. A Figura 1 exemplifica um caso de codificação criativa realizada com o software MAXQDA, onde os códigos são diretamente atribuídos a trechos específicos dos dados, mantendo uma conexão direta com o conteúdo original.

### 2.3.2 Codificação Aberta

Ao contrário da codificação in-vivo, na codificação aberta, os pesquisadores identificam e nomeiam conceitos específicos encontrados nos dados, criando novos códigos para representar esses conceitos. Esses códigos são como etiquetas que capturam os diferentes elementos presentes nos dados, permitindo uma compreensão mais profunda do conteúdo. Enquanto a codificação in-vivo está mais focada em atribuir códigos diretamente com base nas palavras ou frases encontradas nos dados, a codificação aberta envolve uma abordagem mais ampla de identificação e nomeação de conceitos. A Figura 2 ilustra um exemplo de como os pesquisadores podem criar códigos, adicionar anotações e/ou comentários durante o processo de codificação aberta.

### 2.3.3 Codificação Seletiva

Posteriormente, os pesquisadores refinam os códigos criados durante a codificação aberta, agrupando códigos relacionados em

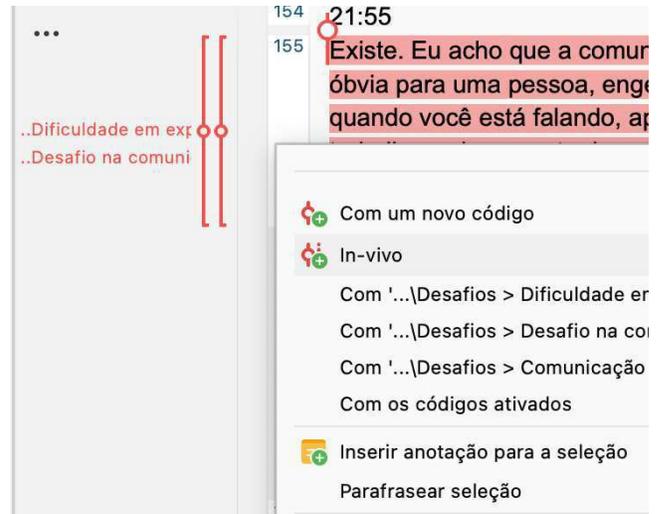


Figura 1: Exemplo de codificação In-Vivo

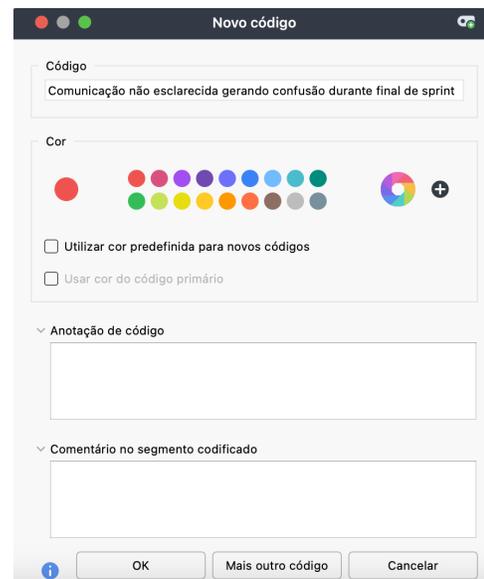


Figura 2: Exemplo de como criar o código, adicionar anotações e/ou comentário

categorias mais amplas e identificando conexões entre eles. Essa fase envolve uma análise mais aprofundada dos dados, onde os códigos são organizados hierarquicamente para capturar a complexidade dos fenômenos estudados. A Figura 3 exemplifica como a codificação pode ser feita, seguindo uma lista de codificação durante essa fase de codificação seletiva.

Esses tipos de codificação trabalham em conjunto para fornecer uma estrutura analítica robusta que permite aos pesquisadores explorar e interpretar os dados de maneira significativa, contribuindo para o desenvolvimento de teorias fundamentadas empiricamente.

Codificar a Lista de Codificações			
Come...	Grupo de documen...	Nome do documento	Código
○	Entrevistas	Transcrição_1	Desafios >...
○	Entrevistas	Transcrição_1	Desafios >...
○	Entrevistas	Transcrição_1	Desafios >...
○	Entrevistas	Transcrição_3	Desafios >...
○	Entrevistas	Transcrição_3	Desafios >...

Figura 3: Exemplo de como codificar seguindo uma Lista de Codificação

### 2.4 Customização do Sistema de Códigos

Durante as etapas posteriores de codificação, é essencial personalizar o sistema de códigos para refletir as nuances dos dados. O MAXQDA facilita isso permitindo a criação de hierarquias de códigos, organizando códigos em categorias pai e subcategorias, o que ajuda a identificar padrões e tendências. A Figura 4 mostra um exemplo de hierarquia entre os códigos, demonstrando como organizá-los de maneira hierárquica para uma análise mais estruturada.

Códigos	256
Formação Acadêmica	6
Experiência Profissional	8
Conflitos de Adaptação	1
Desafio na adaptação de novas tecnologias	3
Treinamento como solução para dificult...	1
Mentoria qualificada e treina...	1
Conflito de adaptação a um novo modelo	1
Opinião	
Opinião	
Opiniões	
Contexto de	

**Conflitos de Adaptação > Desafio na adaptação de novas tecnologias > Mentoria qualificada e treinamento como solução para dificuldade de adaptação à novas tecnologias**

Figura 4: Exemplo de hierarquia entre os códigos

### 2.5 Construção de Categorias com Codificação Criativa

Após a fase de codificação, é hora de construir categorias que irão fundamentar a sua teoria. O MAXQDA oferece a função de Codificação Criativa, que permite a livre movimentação e organização dos códigos em um espaço de trabalho. No exemplo mostrado na Figura 5, podemos observar uma codificação feita pela ferramenta de Codificação Criativa. Essa funcionalidade pode ser utilizada para explorar e organizar dados de maneira flexível, criando categorias significativas e alterando a representação visual dos códigos conforme necessário.

A Figura 6 apresenta um fluxograma criado para ilustrar diferentes abordagens para codificação em contextos técnicos ou educacionais.

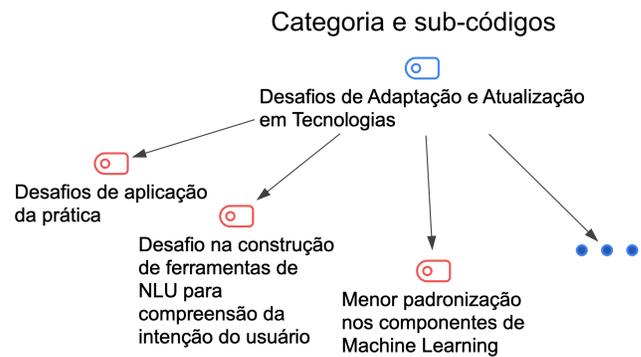


Figura 5: Exemplo de Codificação Criativa com MAXQDA

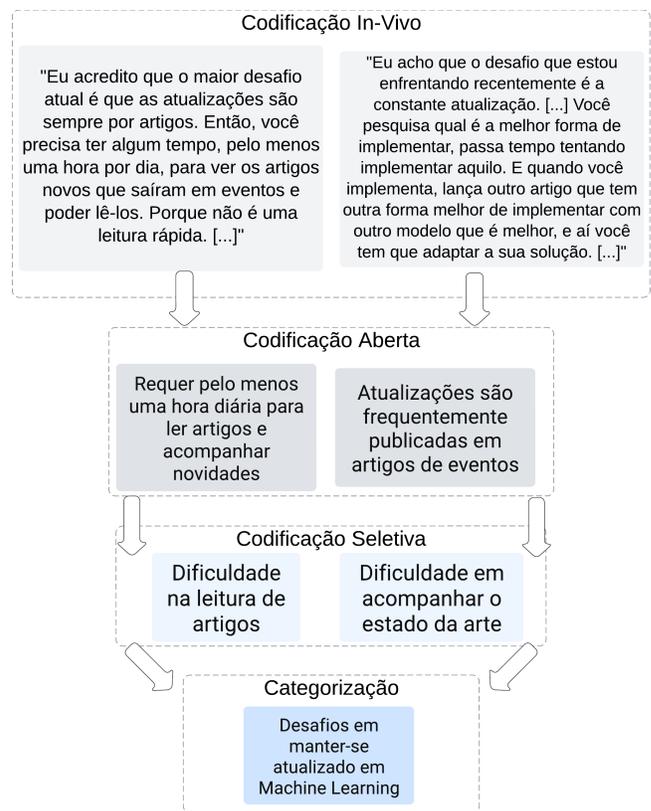


Figura 6: Fluxograma ilustrando diferentes abordagens de codificação

A Figura 6 é um fluxograma que representa visualmente as abordagens de codificação, destacando as seções de "Codificação In-Vivo", "Codificação Aberta", "Codificação Seletiva" e por último uma "Codificação Categórica". Na seção "Codificação In-Vivo", são discutidos desafios relacionados ao código não utilizado, enquanto na seção "Codificação Aberta", são apresentadas diferentes considerações, como a necessidade de menos recursos e a adaptação de soluções. A "Codificação Categórica" é feita já no Sistema de

Códigos, onde os códigos foram agrupados com base em suas semelhanças temáticas, facilitando a análise e compreensão dos desafios enfrentados nas diferentes áreas relacionadas à inteligência artificial e análise de dados. Essa organização permite uma visão mais clara e estruturada dos problemas e questões abordadas nas respostas. Com essa organização, a análise está mais pronta para avançar para os próximos passos, como a construção de teorias e a elaboração de modelos conceituais. Esta figura complementa a discussão sobre os desafios enfrentados ao introduzir novas tecnologias em ML, oferecendo uma perspectiva visual das abordagens de codificação mencionadas pelos entrevistados.

### 3 TRABALHOS RELACIONADOS

Nos trabalhos relacionados, exploramos as práticas emergentes na área de ML e suas implicações na colaboração entre cientistas de dados e engenheiros de software.

O artigo de Nahar et al. [8] aborda os desafios enfrentados na colaboração entre cientistas de dados e engenheiros de software em projetos de sistemas habilitados para ML. Os autores identificam quatro áreas críticas de desafios na colaboração, incluindo comunicação, documentação, engenharia e processo. Essas percepções são essenciais para compreender a dinâmica de equipe e os processos envolvidos na implementação bem-sucedida de sistemas de ML em organizações. A metodologia do estudo envolveu duas etapas principais: Na Etapa 1, os pesquisadores definiram o escopo e desenvolveram um guia de entrevista baseado na revisão da literatura sobre colaboração em sistemas habilitados para ML. Eles selecionaram 15 artigos relevantes e identificaram nove áreas iniciais de colaboração. Na Etapa 2, foram conduzidas entrevistas semiestruturadas com 45 participantes de 28 organizações, cada uma com duração de 30 a 60 minutos. Os participantes foram escolhidos intencionalmente para abranger diferentes papéis, tipos de empresas e países. O estudo visa analisar os desafios de colaboração em projetos de sistemas habilitados para ML, com foco em comunicação, documentação, engenharia e processo. Os pesquisadores identificaram áreas comuns de colaboração e forneceram recomendações para enfrentar esses desafios.

Ludemir [6] explica em seu estudo uma série de desafios enfrentados na área de Inteligência Artificial e ML, que têm implicações diretas na colaboração entre cientistas de dados e engenheiros de software em projetos de ML. Uma das questões discutidas é a necessidade de modelos de ML serem não apenas precisos, mas também explicáveis, justos, que preservem a privacidade, causais e robustos. Isso é relevante para o nosso estudo, pois destacamos a importância do envolvimento precoce de cientistas de dados na definição de requisitos do software, e modelos de ML explicáveis podem facilitar essa colaboração ao permitir que os *stakeholders* entendam as soluções propostas pelos modelos. Além disso, o autor explora a questão da quantidade e qualidade dos dados utilizados no treinamento de algoritmos de ML, destacando a importância de conjuntos de dados representativos e não enviesados. Essa discussão é relevante para nosso estudo, pois identificamos lacunas de conhecimento entre disciplinas e dificuldades na avaliação de modelos de ML, que podem ser exacerbadas por conjuntos de dados inadequados. Outro aspecto tratado por Ludemir durante seu artigo, é a necessidade de enfrentar problemas dinâmicos e contínuos de dados, o que destaca

a importância de técnicas de aprendizado em ambientes dinâmicos para resolver problemas complexos. Isso ressoa com nossas descobertas sobre a necessidade de adaptações de práticas de ES para ML e a importância de desenvolver sistemas de ML que possam lidar com mudanças e atualizações contínuas. Por fim, Ludemir discute o desafio de tornar a resolução de problemas com ML mais automática, destacando o potencial do Meta-Aprendizado para automatizar o processo de seleção de algoritmos e ajuste de hiperparâmetros. Esta discussão é relevante para nossa pesquisa, pois apontamos a importância de desenvolver sistemas de ML eficientes e automatizados.

Texeira [11] investiga a caracterização da comunidade *open source* de MLOps, identificando características e métricas que podem guiar práticas da ES na adoção de ferramentas de automação de MLOps. A pesquisa segue uma metodologia exploratória, minerando e analisando os principais projetos de MLOps para eleger uma forma de diagnosticar a maturidade de uma ferramenta *open source*. Os resultados alcançados incluem um conjunto de boas práticas, diretrizes e uma ferramenta que auxilia profissionais de MLOps na escolha e adoção de ferramentas de automação. Dessa forma, é viável perceber a relevância do estudo de Texeira para nossa pesquisa, pois aborda diretamente a questão da entrega contínua de sistemas de ML, uma das áreas de interesse do nosso trabalho. Além disso, ao analisar a caracterização de projetos de MLOps em comunidades *open source*, este estudo fornece informações valiosas sobre práticas e ferramentas disponíveis para a implementação e manutenção de sistemas de ML, o que pode influenciar diretamente nossas descobertas e recomendações. A compreensão das características e métricas que guiam a escolha de ferramentas de automação de MLOps também pode ajudar a elucidar os desafios e oportunidades enfrentados por equipes que colaboram em projetos de ML, alinhando-se com os objetivos do nosso estudo.

A colaboração entre cientistas de dados e engenheiros de software em projetos de ML é fundamental para o sucesso e a qualidade dos resultados. Neste contexto, nossa pesquisa se destaca ao aprofundar essa integração e fornecer *insights* específicos para superar os desafios inerentes a essas áreas distintas. Um diferencial de nosso trabalho é o uso da ferramenta MAXQDA para análise qualitativa dos dados. Ao empregar técnicas avançadas de análise de dados qualitativos, como codificação, categorização e análise comparativa, pudemos extrair informações valiosas das entrevistas. A integração da MAXQDA em nossa metodologia permitiu uma análise aprofundada das percepções e experiências dos participantes, fornecendo uma base sólida para nossas conclusões e recomendações. Ao preencher lacunas específicas na literatura existente e oferecer uma abordagem mais prática e direcionada para lidar com os desafios na colaboração entre engenheiros de software e cientistas de dados em projetos de ML, nossa pesquisa visa impactar positivamente a qualidade e o sucesso dessas colaborações interdisciplinares. Diferenciando-se de estudos anteriores, nossa pesquisa concentra-se em identificar estratégias práticas e recomendações concretas para melhorar a eficiência e eficácia das colaborações em projetos de ML. Enquanto muitos estudos oferecem uma visão ampla dos desafios e soluções potenciais, nosso trabalho busca oferecer abordagens tangíveis para lidar com esses desafios na prática.

## 4 METODOLOGIA

Nesta seção, delineamos os procedimentos adotados para conduzir nossa pesquisa qualitativa por meio de entrevistas. Abordamos a elaboração do roteiro das entrevistas, a seleção dos participantes, a condução das entrevistas e a transcrição dos dados coletados. Detalhamos também a metodologia de análise qualitativa dos dados, reservando a discussão dos resultados para a seção de Resultados e Discussões.

### 4.1 Entrevistas

Para garantir que as entrevistas abordassem os objetivos de pesquisa de forma abrangente, foi elaborado um roteiro contendo perguntas específicas sobre o perfil dos participantes, práticas de Engenharia de Software no desenvolvimento de componentes de ML, contexto para o uso de ML e os desafios enfrentados por equipes multidisciplinares.

#### 4.1.1 Levantamento do referencial teórico para o contexto das perguntas da entrevista

Antes de iniciar as entrevistas, realizamos um extenso levantamento do referencial teórico relacionado aos desafios na colaboração entre engenheiros de software e cientistas de dados em projetos de ML.

#### 4.1.2 Elaboração do Roteiro de Entrevistas

O roteiro de entrevistas foi elaborado com a contribuição de Sheila Paiva, mestranda em Ciência da Computação pela UFCG. Sheila participou ativamente na definição das perguntas e na estruturação do roteiro. Sua colaboração foi fundamental para garantir a abrangência e relevância das questões abordadas. Enquanto nessa pesquisa a ênfase foi na investigação dos desafios enfrentados durante o desenvolvimento de software, Sheila concentrou-se na exploração das práticas de Engenharia de Software. Essa abordagem diversificada enriqueceu significativamente o conteúdo do roteiro, permitindo uma análise abrangente dos temas relevantes para a pesquisa. Como a ênfase deste trabalho foi em desafios, as perguntas também serão relacionadas a eles.

#### 4.1.3 Convite dos Participantes

Os participantes foram convidados para as entrevistas por meio de convites por e-mail enviados para grupos de comunidades de tecnologia e através do canal da coordenação de Ciência da Computação da Universidade Federal de Campina Grande, para alunos ativos e inativos. Essa abordagem permitiu alcançar participantes que podem garantir uma amostra diversificada e representativa, além de aproveitar redes estabelecidas e facilitar o processo de comunicação. Isso contribuiu para a qualidade e relevância dos resultados da pesquisa.

#### 4.1.4 Elaboração de entrevista piloto

Antes do início das entrevistas, foi realizada uma entrevista piloto com um dos participantes por se enquadrar melhor no perfil estabelecido previamente. O objetivo foi identificar problemas potenciais no roteiro de entrevista e realizar ajustes necessários para garantir

a eficácia da coleta de dados. Essa entrevista teve aproximadamente 45 minutos de duração e posteriormente também foi usada para transcrição e análise.

#### 4.1.5 Realização das Entrevistas

As entrevistas foram conduzidas de forma semi-estruturada, permitindo a flexibilidade para explorar diferentes aspectos conforme surgissem durante a conversa. As oito entrevistas tiveram uma duração média de 35 minutos cada uma e foram gravadas para posterior análise.

### 4.2 Perguntas

Nesta seção, exploramos diversos aspectos relacionados aos desafios e práticas de Engenharia de Software em projetos de ML. Os participantes foram convidados a compartilhar suas experiências e perspectivas sobre diferentes temas, desde os desafios enfrentados na Engenharia de Software até as práticas úteis no desenvolvimento de componentes de ML. Cada subseção aborda uma área específica de interesse, como o contexto de uso do ML, os desafios durante a integração e a necessidade de manter-se atualizado com as novas tecnologias em ML. Ao final das entrevistas, os participantes foram convidados a fornecer *feedback* adicional ou compartilhar pontos que desejavam acrescentar, permitindo uma reflexão final sobre os desafios e aprendizados em projetos de ML.

#### 4.2.1 Desafios em Engenharia de Software

Os participantes foram questionados sobre os desafios enfrentados na área de Engenharia de Software durante o desenvolvimento de software, especialmente com componentes de Machine Learning. As perguntas direcionadas foram:

- Você pode compartilhar exemplos de desafios enfrentados na Engenharia de Software durante o desenvolvimento de software com componentes de ML?
- Se sim, como você superou esses desafios?
- Quais foram os principais aprendizados que você obteve ao superar esses desafios?
- Como esses aprendizados influenciaram suas práticas de Engenharia de Software?
- Houve alguma modificação no processo de desenvolvimento após superar esses desafios?

#### 4.2.2 Desafios Durante a Integração

Aqui, os entrevistados foram questionados sobre os desafios enfrentados ao integrar componentes de ML em projetos de software já existentes. As perguntas específicas foram:

- Quais desafios você enfrentou ao integrar componentes de ML em projetos de software já existentes?
- Como você superou esses desafios?
- Quais foram os principais aprendizados que você obteve ao superar esses desafios?
- Como esses aprendizados influenciaram suas práticas de Engenharia de Software?

Tabela 1: Perfis dos Entrevistados

ID	Gênero	Formação Acadêmica	Tempo como Desenvolvedor (Meses)	Ocupação Atual
P1	Masculino	Graduando em C. da Computação	36	Engenheiro de Software
P2	Feminino	Graduando em C. da Computação	24	Desenvolvedora
P3	Masculino	Mestre em C. da Computação	23	Desenvolvedor
P4	Masculino	Formado em C. da Computação	44	Product Owner
P5	Feminino	Graduando em C. da Computação	33	Pesquisadora
P6	Masculino	Graduando em C. da Computação	38	Pesquisador
P7	Masculino	Mestre em C. da Computação	168	Gerente
P8	Masculino	Mestre em C. da Computação	144	Gerente / Doutorando

#### 4.2.3 Desafios em Manter-se Atualizado

Os entrevistados foram convidados a compartilhar suas experiências em lidar com a introdução de novas tecnologias em ML e os desafios associados a isso. As perguntas direcionadas foram:

- Como você enfrentou desafios em manter-se atualizado em ML? Foi difícil acompanhar as novas tecnologias?
- Como você lidou com a introdução de novas tecnologias em ML? Isso já causou problemas de adaptação?
- O que você acha que poderia ser feito para contribuir com a superação desses desafios?

### 4.3 Transcrição e adaptação dos dados para análise qualitativa

Após cada entrevista, os dados foram transcritos e adaptados para análise qualitativa. Utilizou-se a API do Fireflies<sup>2</sup>, um serviço de transcrição automática, para transcrever automaticamente as entrevistas. O *bot* do Fireflies foi integrado ao Google Meet<sup>3</sup> para realizar a transcrição em tempo real durante as entrevistas, permitindo uma rápida obtenção dos dados transcritos para posterior análise qualitativa. Para isso, o vídeo da entrevista foi revisado novamente, enquanto se lia a frase transcrita simultaneamente. Essa abordagem permitiu identificar possíveis erros ou imprecisões na transcrição automática, garantindo que o conteúdo fosse transposto com precisão para o texto escrito. Essa revisão cuidadosa e a correlação entre áudio e texto foram essenciais para garantir que os dados transcritos refletissem com precisão as informações fornecidas durante as entrevistas. Assim, foi possível realizar uma análise eficaz e confiável dos dados coletados, contribuindo para a qualidade e validade dos resultados obtidos.

### 4.4 Análise dos Dados Utilizando MAXQDA e Teoria Fundamentada

Os dados coletados foram submetidos a uma análise qualitativa utilizando o software MAXQDA 2024. A codificação do texto permitiu a identificação de padrões e tendências emergentes, contribuindo para a compreensão dos desafios na colaboração entre engenheiros de software e cientistas de dados em projetos de ML.

<sup>2</sup>Fireflies. Recuperado de <https://www.fireflies.ai/>

<sup>3</sup>Google Meet. Recuperado de <https://meet.google.com/>

### 4.5 Codificação

As transcrições foram importadas para o MAXQDA. Para cada transcrição, foi feita uma codificação In-Vivo e Aberta, selecionando palavras ou frases diretamente nos dados, identificando pontos cruciais que possam ser posteriormente classificados e categorizados.

**Personalização do Sistema de Códigos:** O Sistema de Códigos do MAXQDA ajudou a organizar uma estrutura hierárquica dos códigos, facilitando que tais códigos passassem por um refinamento cíclico, onde cada vez que eram codificada uma nova transcrição, era revisitados os códigos, verificado se já se se encaixava.

**Construção de Categorias com Codificação Criativa:** Com o modo de Codificação Criativa, a hierarquia de códigos foram visualmente expostas para melhorar a capacidade de encontrar padrões emergentes.

Ao final da codificação, foi possível ter categorias significativas e então fazer uma análise mais profunda dos códigos. Foi aplicado o método da Teoria Fundamentada na análise dos dados, permitindo uma compreensão mais profunda dos fenômenos observados.

## 5 RESULTADOS E DISCUSSÃO

Nesta seção, apresentamos os principais resultados derivados da análise dos desafios identificados na colaboração entre cientistas de dados e engenheiros de software em projetos de ML, em cada entrevistas foi identificado pelo menos um dos desafios a seguir. Além disso, discutimos cada resultado em relação aos achados dos estudos relacionados e à relevância para o campo.

### 5.1 Perfil dos Entrevistados

Para fornecer uma visão detalhada do perfil dos participantes das entrevistas, apresentamos os principais aspectos demográficos e profissionais na Tabela 1. Os participantes foram selecionados de forma a abranger uma variedade de perspectivas e experiências relevantes para os objetivos da pesquisa.

Como ilustrado na Tabela 1, os participantes variam em termos de gênero, formação acadêmica, tempo de experiência como desenvolvedores e ocupações atuais. Essa diversidade enriqueceu as discussões e análises realizadas durante as entrevistas, proporcionando uma compreensão mais abrangente dos desafios e práticas relacionadas ao desenvolvimento de componentes de Machine Learning.

## 5.2 Categorias Significativas

Esta seção apresenta as categorias significativas identificadas durante a análise dos desafios enfrentados na colaboração entre cientistas de dados e engenheiros de software em projetos de ML. No total, foram identificados 231 códigos relacionados a Desafios, e cada um foi classificado em pelo menos uma categoria. Cada categoria destaca padrões e tendências observados, oferecendo informações importantes para compreender as complexidades envolvidas nesse processo colaborativo e orientar estratégias para superar tais desafios. As categorias abordadas incluem Desafios em Engenharia de Software e Análise de Dados, Desafios de Adaptação e Atualização em Tecnologias, Desafios em Processos e Documentação, e Desafios na Integração. A seguir, cada categoria é detalhada com seus respectivos desafios identificados.

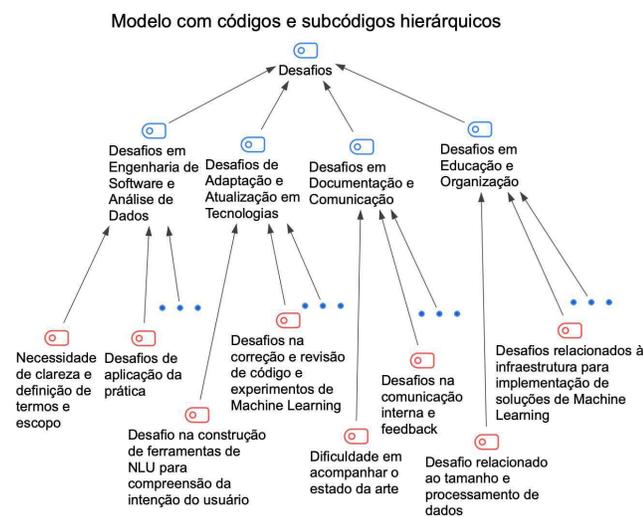


Figura 7: Um fluxograma ilustrando as categorias significativas identificadas.

A Figura 7 ilustra as categorias identificadas durante a análise dos desafios na colaboração entre cientistas de dados e engenheiros de software em projetos de ML. Cada categoria é representada por um ramo distinto no fluxograma, e as setas indicam a relação entre elas. Este fluxograma foi gerado pelo MAXQDA e serve como uma representação visual das categorias discutidas nesta seção e ajuda a contextualizar os desafios enfrentados pelas equipes multidisciplinares.

### 5.2.1 Categoria 1: Desafios em Engenharia de Software e Análise de Dados

Identificamos diversos desafios relacionados à Engenharia de Software, análise de dados e adoção de tecnologias. Os principais padrões e tendências observados incluem:

- **Complexidade dos Projetos de Dados:** Desafios decorrentes da negligência de práticas de Engenharia de Software,

dificuldade em compreender a qualidade dos dados e demonstrar a relevância das análises.

- **Desafios em ML:** Questões como a falta de padronização nos componentes de ML, desafios financeiros na pesquisa em IA e a necessidade de especialização em diferentes vertentes de ML e ciência de dados.
- **Integração e Tempo:** Problemas relacionados à integração, tempo de resposta e alocação de recursos, destacando a importância do tempo para adaptação.

Os padrões acima indicam a complexidade envolvida na Engenharia de Software e análise de dados, destacando áreas críticas que precisam ser endereçadas para melhorar a eficiência e eficácia das operações.

### 5.2.2 Categoria 2: Desafios de Adaptação e Atualização em Tecnologias

Observamos desafios específicos relacionados à adaptação e atualização em tecnologias. Os principais padrões e tendências identificados são:

- **Adoção de Novas Tecnologias:** Dificuldades na construção de ferramentas de Processamento de Linguagem Natural (PLN) e desafios na transição de ferramentas para produção.
- **Defasagem na Comunicação:** Desafios na comunicação entre diferentes áreas de especialização, na representação de todas as atividades do modelo e dificuldades em garantir cobertura adequada dos testes automáticos.
- **Pressão por Resultados Rápidos:** A pressão por resultados rápidos e a necessidade de incorporar soluções de IA no dia a dia dos clientes.

Os padrões acima refletem os obstáculos enfrentados pelas equipes ao tentar adotar e integrar novas tecnologias, destacando a importância de uma comunicação eficaz e da gestão adequada das expectativas dos *stakeholders*.

### 5.2.3 Categoria 3: Desafios em Processos e Documentação

Os participantes ressaltaram desafios relacionados aos processos e à documentação. Os principais padrões e tendências observados incluem:

- **Requisitos e Processamento de dados:** Dificuldades relacionadas ao processo de elicitação de requisitos para ML, cálculo da similaridade entre casos na base de dados e objetivos de processamento de dados.
- **Maturidade e Documentação:** Questões como maturidade variável de *analytics* nas empresas e falta de documentação adequada para futuro treinamentos.
- **Processos e Big Data:** Questões associadas ao processamento de grandes volumes de dados, entendimento conjunto entre desenvolvedores, engenheiros de software e cientistas de dados.

Os desafios acima destacam a necessidade de abordagens colaborativas e adaptativas para lidar com as complexidades dos processos e da documentação em projetos de ML.

#### 5.2.4 Categoria 4: Desafios na Integração

Por fim, identificamos desafios relacionados integração em projetos que usam componentes de ML. Os principais padrões e tendências identificados incluem:

- **Falta de equipe dedicada de MLOps:** A ausência de uma equipe dedicada dificulta a gestão da infraestrutura e a implementação de práticas eficazes de MLOps.
- **Dependência de infraestrutura e equipe especializada:** A dependência de infraestrutura específica e de uma equipe altamente especializada pode dificultar a adoção e implementação de novas soluções.
- **Conflitos na Integração:** Desafios na integração das soluções de IA, na representação de todas as atividades do modelo e dificuldades em garantir cobertura adequada dos testes automáticos.

Os desafios acima ressaltam a importância de uma comunicação eficaz, coordenação adequada e investimento contínuo em atualizações e capacitação para enfrentar os desafios de integração.

### 5.3 Comunicação e Integração

Um dos desafios mais recorrentes identificados na análise foi a falta de comunicação eficaz entre as equipes de cientistas de dados e engenheiros de software. Dentre os oito participantes, seis expressaram dificuldades em entender as necessidades e expectativas uns dos outros, resultando em atrasos e falta de alinhamento nos projetos de ML, contribuindo para problemas como Integração.

Um exemplo dito pelo participante P4 reforça esse problema na Integração, destacando: "...Existe uma complexidade maior da ferramenta em R com notebooks, por isso enfrentamos dificuldades em migrar a ferramenta para a *cloud* da empresa...".

Outro conflito na Integração e Comunicação foi mencionado pelo participante P5, afirmando que: "... a gente teve que fazer justamente essa modificação do banco de dados, modificar como a gente estava armazenando as informações para poder fazer esses cálculos online. Aí, de novo, esse desafio surgiu justamente porque foi pedido uma nova coisa que a gente não estava esperando antes, e aí a gente teve que modificar nosso banco de dados para atender a esse pedido.."

Para a Falta de Comunicação, o participante P2 opina que: "... eu acho que a comunicação, às vezes, não é tão clara, porque uma coisa parece óbvia para uma pessoa, engenheiro de Machine Learning, por exemplo, ou cientista de dados, e quando você está falando, apresentando os resultados da semana, por exemplo, a pessoa que trabalha mais na parte de Engenharia de Software pode simplesmente não estar entendendo nada do que você está falando.."

Essa dificuldade técnica resalta os desafios enfrentados pela equipe de Engenharia de Software ao tentar integrar ferramentas desenvolvidas por cientistas de dados em ambientes de produção. Essa observação corrobora com os achados de Nahar et al. [8], que também destacaram a importância da comunicação eficaz para o sucesso de projetos de ML, juntamente com a integração dos componentes, identificando desafios semelhantes na colaboração entre equipes multidisciplinares.

### 5.4 Lacunas de Conhecimento e Habilidades

A análise revelou lacunas de conhecimento e habilidades entre as equipes, impactando diretamente na eficiência e qualidade dos sistemas de ML desenvolvidos. Quatro participantes mencionaram a "Falta de experiência em integração" e a "Dificuldade em acompanhar o estado da arte" como desafios enfrentados.

Por exemplo, o participante P2 afirmou: "... temos dificuldades em acompanhar as últimas técnicas e ferramentas de ML, o que nos deixa defasados em relação ao estado da arte...".

Essas lacunas demonstram a importância de investir em treinamento contínuo para manter a equipe atualizada e preparada para enfrentar os desafios técnicos do desenvolvimento de sistemas de ML. Essas descobertas ecoam as discussões de Ludemir [6] sobre os desafios na área de Inteligência Artificial e ML, destacando a importância de se manter atualizado com as últimas técnicas e ferramentas de ML para garantir a eficácia no desenvolvimento de sistemas complexos.

### 5.5 Documentação e Adaptação

Outro ponto crítico identificado foi a falta de documentação adequada dos dados e processos de integração. A ausência de documentação foi evidenciada por quatro participantes onde ressaltaram que por causa disso prejudicou a compreensão e colaboração entre as equipes, dificultando a reprodutibilidade e adaptabilidade dos sistemas de ML. Problemas de adaptação foram alertados por todos os participantes, seja adaptação do participante na introdução a uma nova tecnologia, como também adaptação de novos modelos de ML em seus sistemas.

O participante P5 explicou: "... a documentação técnica incompleta nos impede de entender completamente o funcionamento das classes e modelos, dificultando a manutenção e adaptação do sistema..".

Já o participante P6 evidenciou que: "... você fazer o clone daquele repositório e executar aquele modelo, mas falta, como eu posso dizer, um pouco mais descrição do que aquela classe está fazendo, o que está acontecendo naquele modelo naquele momento. Se você pode alterar aquele parâmetro, valores que poderiam ser colocados, carece pouco de documentação, apesar de ter muitos artigos científicos, mas o artigo não traz esse tipo de documentação mais tecnológica.."

Ainda sobre reprodutibilidade e adaptabilidade dos sistemas de ML, o participante P4 afirma que para isso é preciso: "... então, é convencimento. E a pessoa que estou convencendo é uma pessoa que tá sob muita pressão de cima pra fazer as coisas acontecerem. E eu já vi relatos também, essa empresa que eu tô, ela é listada na Bolsa de Valores, capital aberto. Então, eu ouvi falar também que toda empresa que tem capital aberto, elas prezam menos por qualidade. A qualidade é lá embaixo, o negócio é entrega.."

A falta de documentação resalta a importância de adotar práticas e métodos mais adequados aos desafios enfrentados e priorizar a capacitação específica necessária para superar desafios técnicos. Essas observações refletem os achados de Teixeira [11], que enfatizam a importância da documentação adequada e das práticas de Engenharia de Software na comunidade de MLOps, proporcionando *insights* relevantes para abordar os desafios de documentação identificados nesta pesquisa.

## 6 CONCLUSÕES

Na seção de resultados é destacada a complexidade da colaboração entre cientistas de dados e engenheiros de software em projetos de ML e ressaltada a importância de uma abordagem integrada e estratégica para enfrentar os desafios e alcançar o sucesso nos projetos. Essas descobertas contribuem para a compreensão dos fatores que influenciam a eficiência, confiabilidade e adaptabilidade dos sistemas de ML, fornecendo *insights* valiosos para pesquisas futuras e recomendações práticas para aprimorar a colaboração entre as equipes.

A colaboração entre cientistas de dados e engenheiros de software em projetos de ML é essencial para o desenvolvimento bem-sucedido de sistemas complexos. Neste estudo, investigamos os desafios encontrados nessa colaboração, identificando questões cruciais através de entrevistas com profissionais do campo. Emergiram temas como lacunas de conhecimento entre disciplinas, adaptações de práticas de Engenharia de Software para ML, conflitos na integração, comunicação e dificuldades na avaliação de modelos. Destacamos a importância do envolvimento precoce de cientistas de dados na definição de requisitos do software, contribuindo para o alinhamento entre as equipes e para a eficiência do desenvolvimento de sistemas de ML. Além disso, reconhecemos a relevância de abordagens metodológicas como a Teoria Fundamentada, aliada ao uso da ferramenta MAXQDA, para a análise qualitativa dos dados coletados. Por meio dessa pesquisa, oferecemos valiosos *insights* para equipes que enfrentam desafios similares na implementação de ML, destacando a necessidade de estratégias colaborativas e adaptativas para superar as barreiras na colaboração entre cientistas de dados e engenheiros de software. Esperamos que nossas descobertas contribuam para o avanço do campo, promovendo uma colaboração mais eficaz e impactando positivamente a qualidade e o sucesso dos projetos de Machine Learning.

### 6.1 Trabalhos Futuros

Considerando os resultados e discussões apresentados nesta pesquisa, diversas oportunidades surgem para investigações futuras que possam aprofundar o entendimento e abordar lacunas identificadas. Algumas áreas promissoras para trabalhos futuros incluem:

#### 6.1.1 Desenvolvimento de Estratégias de Comunicação

Explorar abordagens mais eficazes para melhorar a comunicação entre cientistas de dados e engenheiros de software, como o uso de ferramentas colaborativas, metodologias ágeis adaptadas para projetos de ML e treinamento em habilidades de comunicação interdisciplinar.

#### 6.1.2 Automação de Processos de Integração e Testes

Investigar soluções automatizadas para a integração de componentes de ML em sistemas de produção, incluindo a implementação de pipelines de CI/CD específicos para ML e o desenvolvimento de ferramentas para teste contínuo de modelos.

#### 6.1.3 Melhoria da Documentação e Reprodutibilidade

Desenvolver diretrizes e boas práticas para a documentação de modelos de ML, incluindo a criação de *templates* de documentação e o uso de ferramentas para capturar informações essenciais sobre o desenvolvimento, treinamento e avaliação de modelos.

#### 6.1.4 Treinamento e Desenvolvimento Profissional

Investir em programas de treinamento e desenvolvimento profissional para equipes multidisciplinares, abordando lacunas de conhecimento identificadas e promovendo a atualização contínua em técnicas, ferramentas e práticas relevantes para projetos de ML.

#### 6.1.5 Avaliação de Impacto e Comunicação em ML

Expandir a pesquisa para incluir considerações éticas e sociais na colaboração entre cientistas de dados e engenheiros de software, investigando métodos para avaliar o impacto social e comunicativo de sistemas de ML e desenvolver *frameworks* para garantir a equidade e transparência em projetos de ML.

Essas áreas representam apenas algumas das muitas direções possíveis para trabalhos futuros na interseção entre ciência de dados, Engenharia de Software e ML. Ao abordar essas questões, pesquisadores e profissionais podem contribuir para avançar o estado da arte em colaboração interdisciplinar e impulsionar a inovação em projetos de ML.

## 7 AGRADECIMENTOS

Gostaria de iniciar meus agradecimentos expressando minha profunda gratidão a Deus por todas as conquistas alcançadas e por Seu constante cuidado em minha vida.

Agradeço também aos meus pais, Marcos Alfredo e MarluCIA de Almeida, por sempre acreditarem em mim e por nunca faltarem amor, atenção e cuidado, sendo minha fonte constante de inspiração e apoio. Um agradecimento eterno aos meus avós, Antônia Adelia e Alfredo Alves, cujo amor e sabedoria iluminaram meu caminho e moldaram minha jornada de vida.

Não posso deixar de mencionar meu orientador, Franklin Ramalho, pela paciência, atenção e valiosas contribuições não apenas para este trabalho, mas também para o meu desenvolvimento pessoal e acadêmico. Um agradecimento especial ao professor Tiago Massoni, cujas ideias contribuíram significativamente para a melhoria desta pesquisa e que também foi parte integrante do projeto como um todo.

À minha amiga Sheila Paiva, minha sincera gratidão por sua colaboração na parte das entrevistas e análises, que enriqueceram os resultados deste trabalho. Por fim, expresso meu profundo agradecimento a todos os meus amigos e familiares por todo apoio, conforto e encorajamento nas horas precisas. Seus gestos de amizade e carinho foram fundamentais para minha jornada acadêmica.

Muito obrigado a todos!

## Referências

- [1] BLOG SOMOSTERA. O que é mlops e como funcionam operações de machine learning. <https://blog.somostera.com/data-science/mlops>, s.d. Acessado em 6 de maio de 2024.

- [2] CHARMAZ, K., AND THORNBURG, R. The pursuit of quality in grounded theory. *Qualitative Research in Psychology* 18, 3 (2021), 305–327.
- [3] GOOGLE CLOUD. Inteligência artificial vs. aprendizado de máquina, s.d.
- [4] JUNCCKES, G. D., AND MORGADO, P. Gerência de riscos em desenvolvimento de software, s.d.
- [5] LOBO, L. H. Uso das técnicas de machine learning na gestão de riscos.
- [6] LUDERMIR, T. B. Inteligência artificial e aprendizado de máquina: estado atual e tendências. *Estudos Avançados* 35, 101 (Jan 2021), 85–94.
- [7] MAXQDA. Code smarter not harder – advanced rounds of coding with maxqda’s smart coding tool. <https://www.maxqda.com/blogpost/smart-coding-tool>, 2021. Acesso em: 26, abril, 2024.
- [8] NAHAR, N., ZHOU, S., LEWIS, G., AND KÄSTNER, C. Collaboration challenges in building ml-enabled systems: Communication, documentation, engineering, and process. In *Proceedings of the 44th international conference on software engineering (2022)*, pp. 413–425.
- [9] SAMYROAD. Exemplos de inteligência artificial: explorando o universo da ia, suas ferramentas e exemplos de uso, junho 9 2023.
- [10] SAS. Machine learning: What it is and why it matters, s.d.
- [11] TEIXEIRA, F. Ferramentas de entrega contínua de sistemas de machine learning em comunidades open source: como caracterizar projetos de ml ops. Trabalho de conclusão de curso, Universidade de Brasília, Brasília, 2022. Trabalho de Conclusão de Curso (Bacharelado em Engenharia de Software).