



**UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE QUIXADÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE**

ANTONIO KAIOS ELIAS PORTELA

**UM ESTUDO EXPLORATÓRIO SOBRE O USO DE LLMS COMO RECURSO PARA O
APRENDIZADO DE FUNDAMENTOS DE PROGRAMAÇÃO**

QUIXADÁ

2025

ANTONIO KAO ELIAS PORTELA

UM ESTUDO EXPLORATÓRIO SOBRE O USO DE LLMS COMO RECURSO PARA O
APRENDIZADO DE FUNDAMENTOS DE PROGRAMAÇÃO

Trabalho de Conclusão de Curso de Graduação
em Engenharia de Software do Campus de
Quixadá da Universidade Federal do Ceará,
como requisito parcial à obtenção do grau de
em Engenharia de Software.

Orientador: Prof. Dr. Jeferson Kenedy
Morais Vieira.

QUIXADÁ

2025

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

P877e Portela, Antonio Kaio Elias.

Um estudo exploratório sobre o uso dos llms como recurso para o aprendizado de fundamentos de programação / Antonio Kaio Elias Portela. – 2025.

72 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Engenharia de Software, Quixadá, 2025.

Orientação: Prof. Dr. Jeferson Kenedy Morais Vieira.

1. Ensino de programação. 2. fundamentos de programação. 3. entrevistas com professores. 4. large language models. I. Título.

CDD 005.1

ANTONIO KAO ELIAS PORTELA

UM ESTUDO EXPLORATÓRIO SOBRE O USO DE LLMS COMO RECURSO PARA O
APRENDIZADO DE FUNDAMENTOS DE PROGRAMAÇÃO

Trabalho de Conclusão de Curso de Graduação
em Engenharia de Software do Campus de
Quixadá da Universidade Federal do Ceará,
como requisito parcial à obtenção do grau de
em Engenharia de Software.

Aprovada em: 26/02/2025.

BANCA EXAMINADORA

Prof. Dr. Jeferson Kenedy Morais
Vieira (Orientador)
Universidade Federal do Ceará (UFC)

Profa. Dra. Maria Viviane de Menezes
Universidade Federal do Ceará (UFC)

Prof. Dr. Jefferson de Carvalho Silva
Universidade Federal do Ceará (UFC)

À minha família, por ser minha base inabalável em todos os momentos. A minha irmã por sempre apoias meus sonhos e a meus pais, que sempre se dedicaram incansavelmente à minha educação e me ensinaram, com amor e sacrifício, o valor do conhecimento. Foi graças ao apoio e incentivo de vocês que hoje alcanço esta conquista.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por me guiar ao longo dessa trajetória, sempre me fortalecendo e me ajudando a superar os desafios, sem permitir que eu fraquejasse.

A Universidade Federal do Ceará por me acolher durante todos esses anos, demonstrando constantemente que o poder do conhecimento é ilimitado e capaz de crescer cada vez mais.

Ao Prof. Dr. Jeferson Kenedy Moraes Vieira, pela orientação dedicada e pelos ensinamentos que me guiaram ao longo de toda a jornada.

Aos professores participantes da banca examinadora, Prof. Dr. Maria Viviane de Menezes e Prof. Dr. Jefferson de Carvalho Silva, pelo tempo dispensado, pelas valiosas contribuições e sugestões que enriqueceram este trabalho.

Aos professores entrevistados, por generosamente compartilharem seus conhecimentos e experiências nas entrevistas realizadas.

Aos alunos que participaram da pesquisa, por dedicarem seu tempo e oferecerem feedbacks tão importantes para o desenvolvimento deste estudo.

A cada professor que fez parte da minha trajetória acadêmica, meu mais sincero agradecimento pela contribuição para minha formação e crescimento.

"Seja um padrão de qualidade. As pessoas não
estão acostumadas a um ambiente onde o
melhor é o esperado." (Steve Jobs)

RESUMO

Este trabalho investiga o impacto dos Large Language Models (LLMs), no aprendizado de desenvolvimento de software, com foco nas disciplinas introdutórias de Fundamentos de Programação (FUP) e Introdução à Programação de Computadores (IPD) na UFC – Campus Quixadá. A pesquisa adota uma abordagem mista, combinando um estudo exploratório sobre a taxa de aprovação dos alunos entre 2015 e 2024, um survey aplicado a discentes e entrevistas com professores que lecionaram essas disciplinas. Os resultados indicam uma tendência de aumento nas taxas de aprovação a partir de 2021, coincidindo com a popularização dos LLMs, e sugerem que essas ferramentas são amplamente utilizadas, especialmente por estudantes com pouca experiência. No entanto, o impacto positivo depende da forma de uso: alunos que interagem criticamente com as respostas dos modelos relatam maior aprendizado, enquanto aqueles que adotam uma abordagem passiva tendem a apresentar dificuldades em consolidar conhecimento. As entrevistas com professores reforçam essa percepção, apontando que, embora os LLMs possam facilitar a compreensão de conceitos complexos, muitos alunos desenvolvem uma dependência excessiva das ferramentas, o que pode prejudicar sua capacidade de resolver problemas de forma autônoma. Os docentes destacam a importância de um uso orientado e consciente dos LLMs para que essas tecnologias atuem como suporte ao aprendizado, sem substituir o raciocínio crítico e a prática de programação. Conclui-se que os LLMs podem ser recursos importante no ensino de programação, desde que integrados a estratégias pedagógicas que incentivem a autonomia e o pensamento crítico dos alunos.

Palavras-chave: Large Language Models; Ensino de Programação; Fundamentos de Programação; Entrevistas com Professores.

ABSTRACT

This study investigates the impact of Large Language Models (LLMs), such as ChatGPT and GitHub Copilot, on programming learning, focusing on the introductory courses of Fundamentals of Programming (FUP) and Introduction to Computer Programming (IPD) at UFC – Campus Quixadá. The research adopts a mixed-methods approach, combining an exploratory study on student approval rates from 2015 to 2024, a survey conducted with students, and interviews with professors teaching these subjects. The results indicate an increasing trend in approval rates since 2021, coinciding with the rise of LLMs, and suggest that these tools are widely used, especially by students with little programming experience. However, their positive impact depends on usage: students who critically engage with LLM-generated responses report greater learning benefits, whereas those who use them passively struggle to consolidate knowledge. Interviews with professors reinforce this perception, highlighting that while LLMs can aid in understanding complex concepts, many students develop excessive reliance on these tools, which may hinder their ability to solve problems independently. Faculty members emphasize the importance of guided and conscious use of LLMs so that these technologies support learning without replacing critical thinking and hands-on programming practice. The study concludes that LLMs can be valuable resources in programming education, provided they are integrated into pedagogical strategies that promote student autonomy and critical thinking.

Keywords: Large Language Models; Programming Education; Fundamentals of Programming; Active Learning, Professor Interviews.

LISTA DE FIGURAS

Figura 1 – Etapas de como o ChatGPT funciona	19
Figura 2 – Metodologias utilizadas na pesquisa	30
Figura 3 – Etapas do processo de realização das entrevistas	31
Figura 4 – Etapas do processo de realização do survey	32
Figura 5 – Taxa de sucesso do curso de Ciência da Computação na disciplina de FUP .	35
Figura 6 – Taxa de sucesso do curso Desing Digital na disciplinas de IPD	36
Figura 7 – Taxa de sucesso do curso de Engenharia da Computação na disciplina de FUP	37
Figura 8 – Taxa de sucesso do curso de Engenharia de Software na disciplina de FUP .	38
Figura 9 – Taxa de sucesso do curso de Redes de Computadores na disciplina de FUP .	39
Figura 10 – Taxa de sucesso do curso de Sistema de Informação na disciplina de FUP .	40
Figura 11 – Média Geral da taxa de sucesso das disciplinas FUP e IPD de todos os cursos	41
Figura 12 – Respostas do survey relacionadas ao curso de graduação.	48
Figura 13 – Experiência dos alunos com programação	49
Figura 14 – Rendimento final nas disciplinas de FUP/IPD em 2024.1	50
Figura 15 – Impacto dos LLMs como forma de auxílio na superação de desafios no aprendizado das disciplinas de FUP/IPD	51
Figura 16 – Impacto dos LLMs no desempenho dos alunos nas disciplinas de FUP/IPD	52
Figura 17 – Ferramenta de LLM como apoio à disciplina de FUP/IPD em 2024.1.	53
Figura 18 – Qualidade das soluções fornecidas pelos LLMs para problemas de programação em FUP/IPD	54

LISTA DE QUADROS

Quadro 1 – Quadro Comparativo	29
Quadro 2 – Perfil dos professores	43
Quadro 3 – Quadro Expositivo	47

SUMÁRIO

1	INTRODUÇÃO	13
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	Inteligência Artificial	15
2.2	Modelo de Linguagem de Grande Escala	16
2.2.1	<i>ChatGPT</i>	18
2.2.2	<i>Copilot</i>	19
2.3	Aprendizagem de Programação	20
2.4	Survey	22
3	TRABALHOS RELACIONADOS	24
3.1	<i>Enhancing Computer Programming Education using ChatGPT- A Mini Review</i>	24
3.2	<i>Generative AI for Programming Education: Benchmarking ChatGPT, GPT-4, and Human Tutors</i>	25
3.3	<i>Augmented Intelligence in Programming Learning: Examining Student Views on The Use Of ChatGPT For Programming Learning</i>	26
3.4	Análise das respostas do ChatGPT em relação ao conteúdo de programação para iniciantes	27
3.5	Análise Comparativa Entre os Trabalhos	29
4	METODOLOGIA	30
4.1	Etapas da metodologia	30
4.1.1	<i>Estudo Exploratório</i>	30
4.1.2	<i>Entrevistas</i>	31
4.1.3	<i>Survey</i>	31
5	RESULTADOS	34
5.1	Resultado dos estudos exploratórios sobre a taxa de sucesso entre os anos de 2015 - 2024 na disciplina de Fundamento de Programação	34
5.2	Resultado das entrevistas feita com os professores que ministraram FUP/IPD em 2024.1	42
5.2.1	<i>Perfil dos docentes</i>	42
5.2.2	<i>Políticas de Uso</i>	44

5.2.3	<i>Impacto no Aprendizado</i>	44
5.2.4	<i>Dificuldades Mais Frequentes</i>	45
5.2.5	<i>Metodologias de Avaliação</i>	46
5.2.6	<i>Perspectivas Futuras</i>	46
5.3	Resultados do survey com os alunos que ingressaram no campus da UFC Quixadá em 2024.1	47
5.4	Análise dos Resultados	56
6	CONCLUSÕES E TRABALHOS FUTUROS	58
6.1	Trabalhos Futuros	59
6.2	Limitações do Estudo	60
	REFERÊNCIAS	61
	APÊNDICE A –PERGUNTAS DO SURVEY SOBRE O USO DO CHATGPT	
	COMO RECURSO PARA O APRENDIZADO DE	
	DESENVOLVIMENTO DE SOFTWARE	65
A.1	Informações Demográficas e Acadêmicas	65
A.2	Uso e Aplicações dos LLMs	67
	APÊNDICE B –ROTEIRO PARA ENTREVISTA COM DOCENTES	
	QUE MINISTRARAM FUP/IPD EM 2024.1 SOBRE	
	O USO DE LLMS NO APRENDIZADO DO DESEN-	
	VOLVIMENTO DE SOFTWARE	69
B.1	Perfil do Docente	69
B.2	Percepção sobre uso de LLMS	69
B.3	Políticas e práticas em sala de aula	70
B.4	Impacto nos Resultados Acadêmicos	70
B.5	Experiências e Opiniões Pessoais	71
B.6	Sugestões e Recomendação de Uso	71

1 INTRODUÇÃO

Com o avanço da tecnologia, a Inteligência Artificial (IA) está se consolidando como uma importante aliada nas atividades humanas, especialmente no desenvolvimento de software. O uso de IA tem revolucionado a forma como problemas são resolvidos, permitindo a automatização de tarefas repetitivas e oferecendo soluções criativas para desafios complexos (Lee, 2019). Entre os modelos de IA mais comuns nesse contexto estão os Modelos de Linguagem de Grande Escala (LLM), sistemas avançados de aprendizado de máquina projetados para compreender e gerar textos em linguagem natural. Exemplos comuns desse tipo de ferramenta são ChatGPT, desenvolvido pela OpenAI, e o GitHub Copilot, criado em parceria entre a OpenAI e a Microsoft (Rodrigues; Rodrigues, 2023). Essas ferramentas são projetadas para realizar tarefas avançadas relacionadas ao processamento de linguagem natural (PLN), incluindo geração de texto, resposta a perguntas, tradução e no caso de ferramentas como o Copilot, geração e sugestão de código diretamente em ambientes de desenvolvimento (Puryear; Sprint, 2022).

Ferramentas como essas estão mudando o cenário do desenvolvimento de software, ampliando as capacidades de desenvolvedores iniciantes e experientes (Gibin *et al.*, 2023). O ChatGPT, lançado pela OpenAI em 2022, utiliza a arquitetura transformacional do modelo *Generative Pre-trained Transformer*, que é capaz de compreender e gerar textos, consolidando-se como uma ferramenta para desenvolver tarefas de geração de código, depuração e explicação de conceitos complexos (Sarrión, 2023). Já o GitHub Copilot integrada a *Integrated Development Environment* (IDE), sugere e gera trechos de código em tempo real, usando o modelo Codex, especializado em compreensão e geração de código de programação, treinado em repositórios públicos (Rahmaniar, 2024). No entanto, embora ofereçam benefícios como aumento de produtividade e aprendizado acelerado, essas ferramentas também apresentam limitações (Vogel *et al.*, 2025). Estudos indicam que respostas geradas por LLMs podem ser imprecisas ou confusas em situações mais complexas, o que pode gerar dificuldades na compreensão e insegurança para os usuários (Lima, 2023).

Devido a sua versatilidade, ferramentas de LLMs tem ganhado destaque no campo educacional e profissional, sendo utilizadas por desenvolvedores para resolver problemas de programação, automatizar tarefas e acelerar processos de aprendizado (Welsby; Cheung, 2023) (Cozman *et al.*, 2021). Entretanto, a literatura também aponta incertezas quanto ao impacto dessas ferramentas no aprendizado de programação. Por exemplo, o estudo de Yilmaz e Yilmaz (2023), realizado com estudantes de graduação em Tecnologia da Informação, destacou vantagens como

respostas rápidas e correção de erros comuns. Contudo, os alunos também relataram dificuldades associadas a respostas incompletas ou incorretas, além de uma sensação de dependência, o que pode comprometer a autonomia no processo de aprendizado.

Diante desse cenário, foi realizado um novo estudo para investigar as incertezas relacionadas ao impacto dos LLMs no desenvolvimento de software, especialmente no contexto do aprendizado do desenvolvimento de software. A temática ainda é pouco explorada na literatura, o que resulta em uma escassez de informações consistentes sobre a real eficácia dessas ferramentas. Ao final do estudo, espera-se obter conclusões sobre o papel dessas tecnologias de apoio no aprendizado do desenvolvimento de software, determinando se elas realmente contribuem para o processo educativo ou apresentam limitações significativas.

O estudo será conduzido com base em dados coletados de disciplinas introdutórias de programação oferecidas nos cursos da UFC - Campus Quixadá, incluindo Engenharia de Software, Engenharia da Computação, Sistemas de Informação, Ciência da Computação e Redes de Computadores. Esses cursos possuem, em sua grade curricular, a disciplina obrigatória de Fundamentos de Programação (FUP), ministrada no início da graduação. Além disso, destaca-se a disciplina específica de Introdução à Programação para Design (IPD) dos curso de Design Digital, integrando assim todas as disciplinas introdutórias de programação do campus.

O objetivo geral deste estudo foi compreender o impacto do uso de Modelos de Linguagem de Grande Escala (LLMs) como recurso de aprendizagem no ensino de programação, por meio da análise de dados empíricos de desempenho acadêmico e das percepções de alunos e professores de disciplinas introdutórias de programação nos cursos de graduação da UFC - Quixadá. Para isso, foram estabelecidos quatro objetivos específicos: analisar o desempenho acadêmico, considerando a taxa de sucesso, em disciplinas introdutórias de programação nos cursos de graduação da UFC - Quixadá no período de 2015 a 2024; investigar a percepção de alunos que cursaram essas disciplinas no último semestre sobre o uso de LLMs como ferramenta de aprendizado; examinar as percepções dos professores dessas disciplinas quanto ao impacto dos LLMs no ensino e aprendizado da programação; e, por fim, comparar os dados históricos de desempenho acadêmico com as percepções dos professores e alunos, buscando identificar possíveis tendências ou correlações no impacto dos LLMs na aprendizagem.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, serão apresentados os principais conceitos para o entendimento deste trabalho, abrangendo os temas: Inteligência Artificial, Modelo de Linguagem de Grande Escala e Aprendizagem de Programação.

2.1 Inteligência Artificial

A Inteligência Artificial (IA) pode ser definida como o campo da ciência da computação que busca desenvolver sistemas capazes de realizar tarefas que normalmente exigiriam inteligência humana, como raciocínio, aprendizado e tomada de decisão (Wang, 2019). Esses sistemas utilizam algoritmos avançados para analisar grandes volumes de dados, identificar padrões e adaptar-se conforme novas informações são introduzidas (Barbosa; Portes Luiza Alves Ferreira, 2023).

O conceito de IA remonta a discussões filosóficas sobre a natureza da mente e da inteligência, mas ganhou forma concreta após a Segunda Guerra Mundial, com a formalização da computação e o surgimento dos primeiros programas inteligentes. Em 1950, Alan Turing propôs o Teste de Turing como um critério para avaliar a inteligência das máquinas, enquanto a Conferência de Dartmouth, em 1956, consolidou a IA como um campo de estudo independente (Russell; Norvig, 2016).

Desde então, a IA evoluiu significativamente, impulsionada pelo avanço do Aprendizado de Máquina (*Machine Learning*), que permite que sistemas aprendam a partir de dados sem a necessidade de programação explícita (Cozman *et al.*, 2021). Dentro desse campo, destacam-se o Aprendizado Supervisionado, no qual os algoritmos são treinados com dados rotulados; o Aprendizado Não Supervisionado, que identifica padrões sem rótulos pré-definidos; e o Aprendizado por Reforço, no qual agentes interagem com o ambiente e aprendem por meio de tentativa e erro (Zhou, 2021).

Uma subárea essencial do *Machine Learning* é o *Deep Learning*, que utiliza redes neurais artificiais para modelar representações hierárquicas de dados (Indurkhyia; Damerau, 2010). Essa abordagem tem sido fundamental para avanços em visão computacional, reconhecimento de voz e geração de linguagem natural (Taulli, 2020). No campo do Processamento de Linguagem Natural (PLN), a IA possibilita a comunicação entre humanos e máquinas por meio da linguagem escrita e falada (Erickson, 2021). Técnicas como Tokenização, Lematização e Stemming

ajudam a estruturar o texto para análise, enquanto métodos estatísticos, como TF-IDF (*Term Frequency-Inverse Document Frequency*) e Bag of Words (*BoW*), permitem que os sistemas extraiam significado dos dados textuais (Nieves *et al.*, 2021).

O PLN envolve seis etapas principais para a análise e compreensão de textos. Uma delas é o *TF-IDF* (*Term Frequency-Inverse Document Frequency*), que combina a Frequência de Termo e a Frequência Inversa nos Documentos. A Frequência de Termo mede quantas vezes uma palavra aparece em um documento em relação ao total de palavras no texto, enquanto a Frequência Inversa nos Documentos valoriza palavras raras, calculando a proporção do número total de documentos pelo número de documentos em que a palavra aparece. Outra técnica importante é o *Bag of Words* (*BoW*), que representa o documento como um vetor baseado na frequência de cada palavra presente no texto, sem considerar a ordem ou a estrutura gramatical. Essa abordagem transforma o texto em dados numéricos que podem ser utilizados por modelos de aprendizado de máquina (Caseli; Nunes, 2024).

A lematização também é uma etapa relevante, pois reduz as palavras às suas formas básicas ou de dicionário, como verbos no infinitivo ou substantivos no singular. Diferentemente do *stemming*, a lematização mantém as palavras em uma forma gramaticalmente correta e natural. Outra etapa importante é a tokenização, que consiste em dividir o texto em unidades menores chamadas tokens, que podem ser palavras, caracteres ou subpalavras. Esse processo define claramente as unidades mínimas de análise para as etapas subsequentes de processamento. A Remoção de *Stopwords* é crucial para eliminar palavras que não contribuem significativamente para o significado geral do texto, como artigos, preposições e conjunções. Isso permite focar nas palavras mais relevantes dentro do contexto. Por fim, o *stemming* comparado anteriormente, é utilizado para simplificar as palavras, reduzindo-as ao seu radical ou raiz. Dessa forma, diferentes formas de uma mesma palavra podem ser tratadas como equivalentes, facilitando a análise de textos em que variações têm significados semelhantes (Caseli; Nunes, 2024).

2.2 Modelo de Linguagem de Grande Escala

Os Modelos de Linguagem de Grande Escala (LLMs) são sistemas avançados de IA que utilizam redes neurais profundas para processar, entender e gerar textos em linguagem natural. Esses modelos são treinados em grandes quantidades de dados textuais, geralmente extraídos de fontes confiáveis como livros, artigos e outros conteúdos da internet. Através desse treinamento, os LLMs aprendem padrões linguísticos complexos, permitindo que desempenhem

tarefas relacionadas ao PLN (Moraes *et al.*, 2024).

Alguns LLMs, incluindo o ChatGPT, é construída com base na arquitetura *Transformer*, uma inovação nas redes neurais projetada para processar dados sequenciais, como texto, com eficiência e precisão. O *Transformer* introduziu o mecanismo de *self-attention*, que permite ao modelo identificar relações contextuais entre palavras e frases em uma sequência, independentemente da posição em que aparecem. Essa característica aumenta a capacidade do modelo de capturar o significado contextual e nuances linguísticas com maior precisão, contribuindo para sua eficácia no processamento de grandes volumes de dados (Soares, 2024).

O treinamento de um LLM é composto de várias etapas que começam com a coleta e pré-processamento de grandes volumes de dados textuais, convertendo essas informações em um formato numérico comprehensível pelo modelo. Durante o aprendizado não supervisionado, o modelo é exposto a sequências de palavras, aprendendo a prever a próxima palavra com base nas anteriores, o que o capacita a identificar padrões e a entender relações linguísticas. A medida que o modelo se ajusta, seus pesos internos são calibrados para aprimorar continuamente sua capacidade de prever com precisão. Posteriormente, ele pode passar por uma fase de aprendizado supervisionado, utilizando pares específicos de entrada e saída para refinar sua precisão em tarefas mais complexas (SILVA, 2023).

Para otimizar o desempenho, o modelo passa por um processo chamado de Aprendizado por Reforço com *feedback* humano, conhecido como *Reinforcement Learning from Human Feedback*. Nesta etapa, pessoas avaliam as respostas geradas pelo modelo e indicam quais são mais adequadas, permitindo que o sistema ajuste seus parâmetros para se alinhar a esses padrões. Esse refinamento com *feedback* humano melhora a qualidade e a adequação das respostas fornecidas. Durante e após cada etapa de treinamento, o modelo é testado com conjuntos de validação para garantir que ele possa generalizar adequadamente em diferentes contextos. Uma vez treinado e implementado, o modelo é atualizado periodicamente para se adaptar a novas tendências e atualizações na linguagem, mantendo a relevância e precisão de suas respostas (SILVA, 2023).

Tanto o GitHub Copilot quanto o ChatGPT exemplificam como os LLMs podem ser aplicados de maneira prática para resolver desafios específicos no desenvolvimento de software e no aprendizado de programação. O GitHub Copilot, baseado no modelo Codex, oferece suporte direto aos desenvolvedores, sugerindo trechos de código em tempo real. Já o ChatGPT, com sua arquitetura *Transformer*, vai além ao atuar como uma ferramenta versátil que auxilia não apenas

na geração de código, mas também na explicação de conceitos complexos, depuração de erros e até na simulação de discussões técnicas. Ambas as ferramentas mostram como os avanços em PLN, tem potencial para transformar profundamente a forma como desenvolvedores aprendem e trabalham, aumentando tanto a produtividade quanto a acessibilidade ao conhecimento técnico (Yetistiren *et al.*, 2022; Teel *et al.*, 2023).

2.2.1 ChatGPT

O ChatGPT é um modelo de IA baseado em aprendizado de máquina, que possui a capacidade de compreender linguagem natural, facilitando a interação e comunicação entre usuários e o sistema (Paredes *et al.*, 2023). Essa IA generativa entende padrões com base no comportamento do usuário e, através do PLN, interpreta as solicitações e fornece respostas adequadas (An *et al.*, 2023).

Construído com a arquitetura *Transformer*, introduzida em 2017, o ChatGPT trouxe avanços significativos no PLN, proporcionando um tratamento mais eficiente e sofisticado de textos (Praveen; Vajrobol, 2023). Desenvolvido pela OpenAI, ele é uma versão do modelo GPT-3, lançado em 2020, treinado com um grande conjuntos de dados textuais para gerar respostas coerentes em linguagem natural (Sharma *et al.*, 2024). Essa arquitetura de rede neural profunda permite ao ChatGPT simular uma conversa humana de forma fluente e relevante (Lima, 2023).

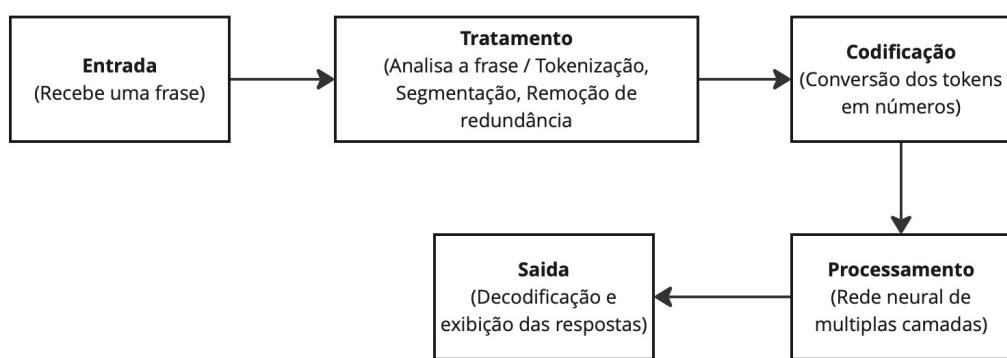
As camadas da arquitetura *Transformer* desempenham um papel central no funcionamento do ChatGPT. Elas são projetadas para processar dados sequenciais, como textos, capturando relações complexas entre palavras em uma sentença. Isso é realizado por meio de mecanismos de atenção, que permitem ao modelo focar em partes específicas do texto conforme necessário para gerar uma resposta contextualizada (Ram; Verma, 2023).

Durante o treinamento, o ChatGPT utiliza um grande conjunto de dados textuais para aprender padrões e relações entre palavras e sentenças (Sun *et al.*, 2023). Esse processo iterativo permite que o modelo refine sua capacidade de resposta continuamente, com ajustes específicos para tarefas como tradução de idiomas e criação de conteúdo (Kalla *et al.*, 2023).

A combinação das tecnologias de PLN e a arquitetura *Transformer* torna o ChatGPT uma ferramenta eficaz para interagir com usuários de forma natural. Ele não apenas comprehende e responde a diferentes formas de linguagem, mas também processa informações de maneira contextualizada, baseando-se em fontes amplas para garantir precisão e relevância nas respostas (Lima, 2023; Yilmaz; Yilmaz, 2023).

De acordo com Briganti (2024), o funcionamento do ChatGPT envolve cinco etapas principais, como mostra a Figura 1. Primeiro, ele recebe uma entrada textual, em seguida, o texto passa por tratamento inicial, onde é tokenizado, ou seja, dividido em pequenas unidades chamadas *tokens* (como palavras ou subpalavras) e segmentado em partes lógicas para facilitar a análise. Após isso, ocorre a codificação dos *tokens* em vetores numéricos, permitindo que o modelo processe as palavras matematicamente.

Figura 1 – Etapas de como o ChatGPT funciona



Fonte: Adaptada de (Briganti, 2024)

A fase de processamento utiliza a arquitetura *Transformer* para analisar o contexto global da entrada, considerando as relações entre os *tokens* e gerando uma resposta coerente. O modelo realiza cálculos complexos para compreender o significado da entrada e construir uma resposta contextualizada. Finalmente, ocorre a decodificação dos vetores em uma resposta em linguagem natural que é exibida ao usuário, completando o ciclo de interação. Assim, o ChatGPT responde de maneira relevante e contextual as solicitações, fornecendo *feedback* em tempo real (Briganti, 2024).

2.2.2 Copilot

O GitHub Copilot é uma ferramenta de IA desenvolvida pela GitHub em colaboração com a OpenAI, projetada para auxiliar programadores e desenvolvedores de software na escrita de código de forma mais rápida e eficiente. Ele age como um assistente de codificação, oferecendo sugestões de linhas, funções ou até mesmo blocos inteiros de código conforme o usuário digita, com base no contexto atual do projeto. O funcionamento do Copilot é impulsionado por modelos

de *Machine Learning* que foram treinados em uma grande quantidade de código disponível publicamente. Integrado a editores de código, através de extensões como o VS Code, ele analisa o que o usuário está tentando criar e propõe sugestões. Essas sugestões são baseadas no código já existente no projeto e no que está sendo escrito, permitindo que o Copilot ofereça soluções direcionadas (Copilot, 2023).

Ao escrever uma problemática, o Copilot sugere automaticamente a melhor saída para esse problema, gerando comentários explicativos e fornecendo documentação, auxiliando no aprendizado e na compreensão do código para desenvolvedores iniciantes e experientes. Entre as funções principais do Copilot estão a sugestão de código em tempo real, aprendizado contextual adaptado ao estilo do projeto em desenvolvimento, compatibilidade com várias linguagens de programação e suporte a prototipagem rápida, permitindo que ideias sejam testadas de maneira eficiente. O Copilot oferece ajuda na depuração de código, sugerindo soluções que seguem boas práticas e auxiliam na correção de erros, sendo uma boa ferramenta para o dia de desenvolvedores (Peng *et al.*, 2023).

2.3 Aprendizagem de Programação

A aprendizagem de programação no desenvolvimento de software é um processo prático e gradual em que os alunos se envolvem diretamente na escrita de códigos para resolver problemas reais. Esse envolvimento é essencial para o desenvolvimento de habilidades técnicas, capacitando-os a criar, entender e modificar software, competências fundamentais em várias áreas da indústria (Medeiros *et al.*, 2018). Durante esse aprendizado, os indivíduos desenvolvem o pensamento computacional, que envolve a capacidade de resolver problemas de maneira lógica e sistemática, decompondo tarefas complexas em etapas menores. Ao aprimorar essas habilidades, a programação ensina a identificar desafios, planejar soluções e implementá-las de forma eficiente (Menon *et al.*, 2013).

Além de prática, a programação é também uma ferramenta de expressão criativa, permitindo transformar ideias em realidade por meio do desenvolvimento de software, aplicativos, jogos e outras soluções tecnológicas. Os conceitos fundamentais, como variáveis, estruturas de controle e funções, são frequentemente os primeiros tópicos abordados. À medida que avançam, os alunos são expostos a conceitos mais complexos, como estruturas de dados, algoritmos, paradigmas de programação e práticas de codificação eficientes (Blackwell, 2002). Esse processo é apoiado por ferramentas e linguagens modernas, adaptadas tanto com as necessidades do

mercado quanto ao perfil dos alunos (Engholm, 2010).

Atualmente, há várias ferramentas que auxilia os estudantes no desenvolvimento de software. Entre elas, o GitHub facilita a colaboração em projetos de software, o controle de versão e o trabalho em equipe, sendo essencial para o desenvolvimento colaborativo. Um recurso importante é o GitHub Copilot, que usa inteligência artificial para sugerir código em tempo real, falado anteriormente, auxiliando na escrita de funções complexas e oferecendo soluções automáticas para problemas comuns, ajudando os alunos a manterem o fluxo de trabalho e aprimorarem suas habilidades (Copilot, 2023). Já o Visual Studio Code (VS Code) é um editor de código muito usado, com diversas extensões para várias linguagens, oferecendo recursos como autocompletar, depuração e integração com o GitHub (Murphy-Hill *et al.*, 2015).

Outras ferramentas, como o Replit, permitem que os alunos escrevam, executem e compartilhem código diretamente no navegador, eliminando a necessidade de configurar ambientes de desenvolvimento locais (Kovtaniuk, 2023). Para estudantes de desenvolvimento web, o CodePen proporciona visualização instantânea de HTML, CSS e JavaScript à medida que o código é editado, facilitando um aprendizado visual e prático (Abajyan, 2018). Já o Codecademy é uma plataforma que oferece um ambiente interativo com *feedback* imediato, permitindo que os alunos pratiquem e aprimorem suas habilidades (Hondro, 2024).

Ferramentas como Docker também são populares, facilitando a criação, teste e implantação de aplicações em contêineres, simplificando o ambiente de desenvolvimento e evitando problemas de configuração (Docker, 2020). Além disso, o ChatGPT emerge como um recurso no processo de aprendizado, oferecendo suporte adaptável e contínuo. Ele permite que os alunos obtenham soluções de código, explicações detalhadas e exemplos personalizados em tempo real, facilitando tanto a depuração quanto o entendimento de conceitos complexos (Hörnemalm, 2023).

A jornada de aprendizagem de programação é contínua, envolvendo tentativas e erros, e exige uma mentalidade de aprendizado constante. A programação é aplicável em diversas áreas, como matemática, física e economia, sendo uma ferramenta essencial para modelos, simulações e análises complexas. Além disso, ela promove a colaboração, com projetos desenvolvidos em equipe ou em comunidades online, favorecendo a troca de conhecimento entre programadores de diferentes níveis (Silva *et al.*, 2015; LOPES; GARCIA, 2002).

As disciplinas de Fundamentos de Programação (FUP) e Introdução à Programação para Design (IPD) são oferecidas no primeiro semestre da graduação, sendo o primeiro contato

dos alunos com a programação. Elas introduzem conceitos como algoritmos, estruturas de controle, variáveis, tipos de dados, funções e estruturação de programas, utilizando linguagens como Python, C, Java e JavaScript. O objetivo é desenvolver o pensamento computacional e ensinar os princípios lógicos da programação, capacitando os alunos a resolver problemas computacionais e criar programas eficientes (Aguilar, 2008).

Na UFC – Campus Quixadá, a entrada de novos alunos ocorre apenas uma vez por ano, no primeiro semestre, por meio do Sistema de Seleção Unificada (SISU). A FUP e a IPD são inseridas nesse período, o que pode gerar dificuldades caso o aluno reprove, especialmente nos semestres pares. Isso ocorre porque, após a reprovação, o aluno acumula disciplinas e, ao retornar, encontra turmas mistas, com alunos de diferentes cursos, o que pode afetar sua adaptação. Nos semestres ímpares, os alunos tendem a estar mais motivados, pois estão iniciando o curso. Já nos semestres pares, o aumento da complexidade das disciplinas e o acúmulo de matérias podem impactar negativamente seu desempenho, além da adaptação ao ambiente universitário e à gestão do tempo, gerando uma diferença na taxa de sucesso (Silva *et al.*, 2015).

2.4 Survey

O survey é um método de coleta de dados utilizado para investigar opiniões, comportamentos, atitudes ou características de um grupo de indivíduos sobre um tema específico. Ele pode ser exploratório, descritivo ou explanatório, dependendo do objetivo da pesquisa. O survey exploratório tem como objetivo identificar conceitos iniciais sobre um tópico e determinar quais conceitos devem ser medidos. Nesse caso, ele busca mapear o que ainda não é bem entendido sobre o tema e ajudar a desenvolver uma base teórica para futuras investigações (Tavares *et al.*, 2020). O survey descritivo, por sua vez, busca identificar quais situações, eventos, atitudes ou opiniões aparecem em uma população. Ele é utilizado para descrever características ou tendências dentro de um grupo, mas não necessariamente para explicar causas ou relações. Já o survey explanatório tem como foco testar uma teoria e as relações causais, buscando entender as interações entre variáveis e verificar hipóteses (Scheuren, 2004).

O primeiro passo ao realizar um survey é definir claramente os objetivos da pesquisa, ou seja, determinar o que se deseja descobrir ou as informações que precisam ser obtidas. Esse planejamento inicial é fundamental para guiar o processo e garantir que as perguntas sejam focadas e relevantes (Kitchenham; Pfleeger, 2008). Em seguida, é necessário fazer o design do survey, que envolve a criação do questionário. Nessa etapa, decide-se o tipo de perguntas, o

formato das respostas e a estrutura geral do questionário. O design do survey também inclui a definição do público-alvo e a organização das perguntas de forma lógica e clara, de modo que os respondentes possam entender facilmente o que está sendo perguntado. Após definir o design, a próxima fase é o desenvolvimento do instrumento de pesquisa, que é o próprio questionário. As perguntas precisam ser formuladas de maneira objetiva, imparcial e clara, de forma a evitar ambiguidade e viés nas respostas. O questionário deve ser capaz de coletar dados relevantes de forma eficiente e precisa. Antes de aplicar o survey, é fundamental avaliar o instrumento por meio de um teste piloto, aplicando o questionário a um grupo pequeno para identificar possíveis falhas ou problemas, como perguntas mal formuladas ou partes confusas (Martinez, 2017).

A etapa de testes ajuda a fazer ajustes necessários para garantir que o survey será eficaz. Após a avaliação, a coleta de dados válidos se torna a próxima etapa, onde o survey é finalmente aplicado ao público-alvo. A coleta deve ser feita com cuidado para garantir que as respostas sejam sinceras e representativas, respeitando a confidencialidade e o consentimento dos participantes. Por fim, a análise dos dados coletados é a etapa final. Os dados são organizados e analisados, seja por meio de métodos estatísticos para dados quantitativos ou por análise de conteúdo para dados qualitativos. O objetivo da análise é identificar padrões, tendências e insights que respondam às perguntas da pesquisa e atinjam os objetivos definidos inicialmente (Cendón *et al.*, 2014).

3 TRABALHOS RELACIONADOS

Nesta seção, serão apresentados quatro trabalhos relacionados sobre uso do ChatGPT no contexto do aprendizado do desenvolvimento de software.

3.1 *Enhancing Computer Programming Education using ChatGPT- A Mini Review*

O estudo de Deriba *et al.* (2024) explora como o ChatGPT influencia a educação, com um foco especial no aprendizado de programação, buscando compreender de que maneira a ferramenta tem auxiliado tanto estudantes quanto educadores em cursos de computação. Para realizar a análise, foi conduzida uma revisão rápida de literatura nas bases Scopus e ACM, selecionando 13 artigos publicados em 2023. Esses artigos foram organizados em categorias que abordam aspectos como a precisão e confiabilidade do ChatGPT, seu impacto no ensino de programação, o desenvolvimento de habilidades colaborativas e as questões éticas envolvidas.

Os achados do estudo indicam que o ChatGPT desempenha um papel de apoio no aprendizado de programação de múltiplas formas. A ferramenta se destaca por simplificar códigos complexos, auxiliar iniciantes e promover a motivação e o engajamento dos estudantes. Adicionalmente, o ChatGPT contribui para aumentar a autoconfiança dos estudantes em programação, oferece suporte em turmas com uma alta razão de alunos por professor, facilita a prática de leitura e explicação de código, atua como "par programador" e ajuda a esclarecer conceitos avançados, como estruturas de dados e algoritmos. Esse conjunto de funções transforma o ChatGPT em um recurso valioso no ensino de computação, beneficiando alunos com diferentes níveis de habilidade.

A pesquisa conclui que o ChatGPT está se tornando cada vez mais relevante no ensino de programação, apontando que a ferramenta não apenas torna o aprendizado mais acessível e estimula o engajamento dos alunos, mas também oferece um suporte considerável aos educadores. No entanto, embora os benefícios sejam evidentes, o uso do ChatGPT na educação traz à tona questões éticas que precisam ser cuidadosamente abordadas, como a dependência excessiva de IA e possíveis vieses nos conteúdos gerados. Com base em seus impactos positivos, o estudo sugere a necessidade de investigações futuras sobre o papel do ChatGPT na educação para equilibrar suas vantagens com práticas éticas responsáveis.

A pesquisa de Deriba *et al.* (2024) apresenta pontos de convergência com este estudo, especialmente no que tange ao impacto do ChatGPT no aprendizado de programação. Ambos

os trabalhos compartilham o interesse em compreender como ferramentas baseadas em LLMs podem auxiliar estudantes e educadores no ensino de computação. Enquanto a pesquisa de Deriba *et al.* (2024) organiza as evidências de literatura em categorias como precisão, confiabilidade e impacto educacional, este estudo busca analisar percepções e dados empíricos, considerando tanto os alunos quanto os professores das disciplinas introdutórias de programação na UFC - Quixadá. Nossa investigação vai além ao explorar o desempenho acadêmico ao longo dos anos (2015-2024), permitindo a identificação de tendências e correlações mais amplas. Outra diferença relevante está na abordagem metodológica. Enquanto o estudo de Deriba *et al.* (2024) foca em uma revisão de literatura recente, este estudo combina análise histórica de desempenho acadêmico com dados coletados diretamente de alunos e professores, o que proporciona uma perspectiva prática e contextualizada ao uso de LLMs.

3.2 *Generative AI for Programming Education: Benchmarking ChatGPT, GPT-4, and Human Tutors*

O estudo conduzido por Phung *et al.* (2023) explora a utilização da IA, com foco em modelos como ChatGPT e GPT-4, no apoio a estudantes de programação, especialmente no fornecimento de dicas, feedbacks de avaliação e suporte para atividades de programação colaborativa. A pesquisa compara o desempenho desses modelos com o de tutores humanos. Os resultados indicam que o GPT-4 supera o ChatGPT em precisão, especialmente em cenários de geração de dicas e colaboração, atingindo uma taxa de acerto de 66% contra 18% do ChatGPT, ainda inferior aos 92% alcançados pelos tutores humanos.

Em termos de feedback avaliativo, o GPT-4 também mostra resultados superiores ao ChatGPT, embora ambos fiquem atrás dos tutores humanos em situações mais complexas. No contexto de programação em pares, o GPT-4 demonstrou maior capacidade para manter o contexto e corrigir programas de forma precisa, com 64% de precisão em comparação a 38% do ChatGPT, enquanto os tutores humanos alcançaram 82%.

Os achados sugerem que, embora o GPT-4 represente um avanço significativo, a IA ainda não atinge plenamente o nível de apoio oferecido por tutores humanos. A conclusão do estudo aponta que, apesar do progresso, ainda são necessárias melhorias na precisão e contextualização dos modelos de IA para que possam se tornar uma ferramenta confiável e eficaz no ensino de programação, destacando a importância do aprimoramento contínuo desses modelos.

Esse estudo adota uma abordagem mais ampla, investigando como diversas ferramentas de IA, incluindo o ChatGPT, estão sendo utilizadas por estudantes e professores no aprendizado e ensino de programação no contexto da UFC - Quixadá. Enquanto o estudo de Phung *et al.* (2023) analisa a precisão e eficácia dos modelos em tarefas específicas, esta pesquisa explora a percepção dos usuários sobre a frequência de uso de LLMs e o impacto dessas ferramentas na compreensão de conceitos de programação. Além disso, o estudo mencionado se concentra na comparação direta entre modelos de IA e tutores humanos, destacando a superioridade dos tutores em precisão e contexto, enquanto nosso trabalho busca entender o papel complementar dessas ferramentas no ambiente educacional, considerando suas potencialidades e limitações no suporte a turmas com diferentes níveis de complexidade e proporções.

3.3 Augmented Intelligence in Programming Learning: Examining Student Views on The Use Of ChatGPT For Programming Learning

O artigo de Yilmaz e Yilmaz (2023) tem como objetivo avaliar as percepções dos alunos sobre a utilização do ChatGPT no aprendizado de programação. Para isso, a pesquisa envolveu 41 estudantes de graduação dos cursos de Tecnologia da Informação de uma universidade pública, durante a disciplina de Programação Orientada a Objetos II, ao longo de 8 semanas. Os alunos foram incentivados a utilizar o ChatGPT para resolver atividades relacionadas ao curso.

A metodologia do estudo consistiu em monitorar o uso do ChatGPT pelos alunos durante o período da disciplina e avaliar os efeitos da ferramenta nas atividades de programação. Foram coletados dados sobre a eficácia da ferramenta, suas contribuições e limitações percebidas pelos alunos.

Os principais resultados indicaram que o ChatGPT trouxe benefícios significativos, como a capacidade de fornecer respostas rápidas e geralmente corretas, aprimorar as habilidades de raciocínio, facilitar a depuração de código e aumentar a autoconfiança dos alunos. No entanto, também foram identificadas limitações, incluindo a dependência excessiva da ferramenta, a incapacidade do ChatGPT de responder a algumas perguntas ou fornecer respostas incompletas, e o aumento da ansiedade profissional entre os alunos.

O estudo de Yilmaz e Yilmaz (2023) e a pesquisa em andamento compartilham o foco no uso de LLMs no ensino de programação, porém abordam o tema de maneira distinta. Enquanto o estudo de Yilmaz e Yilmaz (2023) se concentra na experiência direta de uso do ChatGPT por 41 alunos em atividades de programação, observando o impacto da ferramenta

durante a disciplina de Programação Orientada a Objetos II, nossa pesquisa adota uma abordagem mais ampla, visando compreender as percepções dos discentes sobre o impacto e a eficácia do uso do LLMs em seu processo de aprendizagem de programação no contexto acadêmico da UFC - Quixadá. O estudo de Yilmaz e Yilmaz (2023) coleta dados sobre a experiência prática dos alunos com o ChatGPT, identificando benefícios, como aumento da autoconfiança e melhoria nas habilidades de depuração de código, bem como limitações, como a dependência excessiva da ferramenta. Em contraste, nossa pesquisa busca analisar não apenas os efeitos práticos, mas também as percepções mais amplas sobre como as ferramentas estão sendo utilizadas. Ambos os estudos reconhecem o valor do ChatGPT como recurso de apoio ao aprendizado, mas enquanto um observa o uso em tempo real, o outro busca entender a avaliação dos discentes sobre essa utilização.

3.4 Análise das respostas do ChatGPT em relação ao conteúdo de programação para iniciantes

O estudo realizado por (Filho *et al.*, 2023) buscou examinar a qualidade das respostas geradas pelo ChatGPT em temas introdutórios de programação, com foco em sua utilidade para alunos iniciantes. A investigação incluiu dois tipos de testes: qualitativos, que analisaram tanto a clareza do código gerado quanto a precisão das explicações, considerando as interações feitas com o ChatGPT; e quantitativos, que examinaram as respostas sem interação adicional, abordando tópicos como uso de variáveis, operadores, estruturas de decisão e repetição, além de vetores e matrizes.

Os achados revelaram um desempenho geral superior a 80%, incluindo um acerto completo para tópicos básicos. No entanto, em conteúdos mais complexos, como manipulação de strings e vetores, o ChatGPT demonstrou algumas limitações, especialmente no que diz respeito à qualidade das respostas para esses conteúdos avançados. Em certas respostas, a ferramenta utilizou funções que podem ser inseguras ou avançadas demais para alunos iniciantes, requerendo, assim, conhecimentos prévios para que os estudantes possam identificar e corrigir os erros.

Em conclusão, o ChatGPT mostrou-se promissor como uma ferramenta auxiliar no aprendizado de programação. Contudo, suas respostas não são sempre totalmente precisas ou adequadas para estudantes iniciantes, o que destaca a necessidade de um conhecimento prévio para que o usuário possa avaliar criticamente e aproveitar melhor as respostas oferecidas pela

ferramenta.

O estudo de Filho *et al.* (2023) e a pesquisa em andamento tem objetivos semelhantes no que diz respeito ao uso do ChatGPT como ferramenta de apoio ao aprendizado de programação, mas abordam o tema de maneira distinta. O trabalho de Filho *et al.* (2023) foca na avaliação da qualidade das respostas geradas pelo ChatGPT em tópicos introdutórios de programação, analisando tanto aspectos qualitativos, como a clareza do código e a precisão das explicações, quanto quantitativos. Embora os resultados mostrem um bom desempenho da ferramenta em tópicos básicos, as limitações se tornam evidentes em conteúdos mais complexos, sugerindo que o ChatGPT pode não ser totalmente adequado para iniciantes sem o conhecimento prévio necessário para interpretar e corrigir suas respostas. Em contraste, este estudo foca em entender o impacto dos LLMs no processo de aprendizado de programação de forma mais abrangente, coletando percepções dos alunos sobre o uso das ferramentas no contexto acadêmico da UFC - Quixadá. Embora ambos os estudos reconheçam o potencial do ChatGPT como uma ferramenta útil para estudantes de programação, esta pesquisa explora não apenas a qualidade das respostas, mas também o impacto mais amplo das ferramentas de LLMs no desenvolvimento das habilidades dos alunos.

3.5 Análise Comparativa Entre os Trabalhos

O Quadro 1 permite vizualizar o comparativo entre os trabalhos relacionados.

Quadro 1 – Quadro Comparativo

Critérios	(Deriba <i>et al.</i> , 2024)	(Phung <i>et al.</i> , 2023)	(Yilmaz; Yilmaz, 2023)	(Filho <i>et al.</i> , 2023)	Este trabalho
Objetivo	Analisar como o ChatGPT influencia o ensino de programação.	Avaliar como o ChatGPT podem complementar ou substituir tutores humanos.	Avaliar a percepção dos alunos sobre a utilização do ChatGPT no aprendizado de programação.	Avaliar as respostas do ChatGPT sobre conteúdos básicos de programação.	Explorar o uso de LLMs como ferramenta no processo de aprendizagem de programação.
Método de Pesquisa	Revisão rápida de literatura.	Realização de testes em cenários de ensino de programação.	Pesquisa de campo com estudantes.	Testes qualitativos e quantitativos.	Pesquisa de campo com discentes e docentes sobre experiências e percepções.
Coleta de dados	Foram selecionados 13 artigos publicados em 2023, nas bases Scopus e ACM.	Respostas de GPT-4, ChatGPT e tutores humanos para problemas de programação.	Observação e questionários aplicados aos alunos.	Realização de testes básicos com a linguagem de programação.	Estudo Exploratório, Survey com discentes e Entrevista com docentes.
Análise dos dados	Os artigos foram categorizados por temas.	Comparação de desempenho usando métricas de correspondência.	Análise estatística e qualitativa das respostas dos alunos.	Avaliação da correção das respostas.	Análise qualitativa e quantitativa dos dados coletados.
População/Amostra	Os 13 artigos voltados ao uso do ChatGPT na educação em programação.	Dois tutores humanos avaliando tarefas de programação introdutória em Python.	41 estudantes de graduação em Tecnologia da Computação e Sistemas de Informação.	Exercícios de programação básica, especialmente direcionados a estudantes iniciantes.	Docentes e discentes dos cursos de tecnologia da informação do UFC - Campus Quixadá.

Fonte: Elaborado pelo autor.

4 METODOLOGIA

Esta seção apresenta os procedimentos metodológicos adotados neste estudo para alcançar os objetivos propostos, fundamentando em dados empíricos. A coleta de dados foi realizada por meio de três abordagens complementares: análise de dados históricos de aprovações em disciplinas FUP e IPD, entrevistas com professores que lecionaram no semestre 2024.1 essas respectivas disciplinas e aplicação de um *survey* com alunos do mesmo período. Essas estratégias foram escolhidas para oferecer uma visão abrangente sobre o impacto do uso de LLMs na aprendizagem de programação.

4.1 Etapas da metodologia

Figura 2 – Metodologias utilizadas na pesquisa



Fonte: Elaborado pelo autor.

4.1.1 Estudo Exploratório

O estudo exploratório incluiu a análise de dados históricos de aprovações em disciplinas fundamentais relacionadas ao desenvolvimento de software. Foram analisadas turmas de Fundamentos de Programação (FUP) dos cursos de Engenharia de Software, Engenharia da Computação, Ciência da Computação, Sistemas de Informação e Redes de Computadores, bem como a disciplina de Introdução à Programação para Design (IPD), ofertada exclusivamente ao curso de Design Digital. A disciplina IPD compartilha os mesmos preceitos de FUP, mas com uma carga horária reduzida.

Os dados foram coletados de forma anonimizada por meio do Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA), abrangendo o período de 2015 a 2024.1. A taxa de sucesso foi definida como a relação entre o número de estudantes aprovados e o total de matriculados em cada turma. Essa métrica permitiu avaliar o desempenho histórico dos alunos, fornecendo uma base quantitativa para investigar possíveis mudanças associadas à introdução de tecnologias como ferramentas de LLMs no processo de aprendizagem.

4.1.2 *Entrevistas*

Figura 3 – Etapas do processo de realização das entrevistas



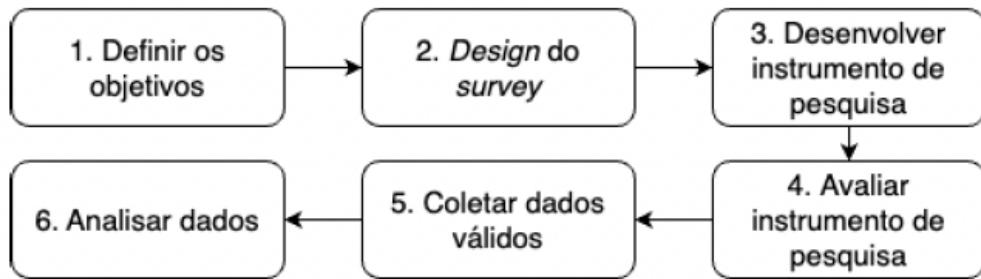
Fonte: Adaptada de (Lüdke; ANDRÉ, 1986)

O primeiro passo das entrevistas foi a formulação de um roteiro, que estabeleceu os objetivos e definiu as diretrizes para a coleta de dados. Esse roteiro organizou as etapas da pesquisa e orientou a construção do questionário para as entrevistas. Após a formulação do roteiro, foi realizada a etapa das entrevistas com os docentes da UFC - Quixadá que lecionaram a disciplina de FUP e IPD em 2024.1. As entrevistas, de caráter semi-estruturado, foram conduzidas conforme descrito no roteiro, que está disponível no Apêndice B, permitem explorar de forma aprofundada as experiências e opiniões dos docentes. Por fim, os dados obtidos nas entrevistas foram tabulados e submetidos a uma análise comparativa. Essa etapa tem como objetivo identificar padrões e tendências que ajudem a compreender como os LLMs estão influenciando a aprendizagem e o desempenho acadêmico dos estudantes.

4.1.3 *Survey*

A Figura 4 apresenta uma visão detalhada das etapas envolvidas no processo de realização do survey, desde a definição dos objetivos da pesquisa até a análise dos dados coletados. Cada etapa é importante para garantir a eficácia da pesquisa e a qualidade das informações obtidas.

Figura 4 – Etapas do processo de realização do survey



Fonte: (Kitchenham; Pfleeger, 2008; Vieira *et al.*, 2023)

O objetivo deste survey é compreender se os alunos utilizaram LLMs durante suas atividades acadêmicas e, em caso afirmativo, identificar quais ferramentas foram empregadas, buscando mapear o uso dessas tecnologias e entender quais são as mais populares e acessíveis no contexto educacional. Além disso, o estudo investiga como essas ferramentas foram aplicadas, analisando os contextos em que os LLMs se mostram mais úteis e como são integradas ao aprendizado. Também é avaliado as percepções sobre o impacto do uso dessas ferramentas em seu desempenho acadêmico, considerando tanto os benefícios quanto as possíveis limitações.

A metodologia baseia-se em um survey composto por um questionário estruturado, projetado para coletar dados de estudantes de maneira representativa. O survey adota um design transversal, seguindo o modelo proposto por Kitchenham e Pfleeger (Kitchenham; Pfleeger, 2008), em que os participantes respondem a questões sobre o uso de LLMs em um momento específico. Esse questionário foi administrado via Google Forms, permitindo fácil acesso e respostas autônomas pela internet. Foram capturados dados sobre a frequência e a forma de uso de LLMs, além das percepções dos discentes quanto à eficácia e às limitações dessas ferramentas no processo de aprendizado de programação.

O desenvolvimento do instrumento de pesquisa foi elaboração de um questionário completo, disponível no apêndice A para consulta. As perguntas foram planejadas para captar as experiências dos discentes no uso de LLMs em atividades acadêmicas ligadas ao desenvolvimento de software. Está incluso questões sobre o nível de familiaridade com diferentes linguagens de programação, a complexidade dos problemas enfrentados e a contribuição dos LLMs na resolução desses problemas. Antes da aplicação, o questionário foi submetido a um pré-teste com um pequeno grupo de discentes. Esse pré-teste permitiu verificar se as perguntas eram claras e abrangiam todas as áreas relevantes ao aprendizado de programação, garantindo a validade e a confiabilidade do instrumento.

A coleta de dados foi realizada com discentes dos cursos de Engenharia de Software, Engenharia da Computação, Sistemas de Informação, Ciência da Computação, Design Digital e Redes de Computadores do Campus Quixadá da UFC. A escolha desses alunos foi devido a relação direta com disciplinas de programação. A análise dos dados coletados ocorreu em etapas. Inicialmente, foi feita uma análise descritiva para identificar padrões de uso das LLMs entre os diferentes grupos de discentes. Foram considerados critérios como o nível de familiaridade com linguagens de programação e a frequência de uso das ferramentas. Com isso, tornou-se possível avaliar como essas tecnologias estão sendo integradas ao processo de aprendizado, além de analizar o impacto na eficiência e na qualidade do aprendizado dos estudantes.

5 RESULTADOS

Nesta seção, são apresentados os resultados obtidos ao longo do estudo. A primeira seção aborda um estudo exploratório sobre a taxa de sucesso dos alunos aprovados na disciplina de FUP e IPD no período de 2015 a 2024. A segunda seção expõe os resultados de entrevistas realizadas com docentes da UFC-Quixadá que ministraram a disciplina de FUP e IPD no período de 2024.1. Por fim, a terceira seção apresenta os resultados de um survey realizado com discentes da UFC-Quixadá, focado no uso de LLMs para o aprendizado no desenvolvimento de software.

5.1 Resultado dos estudos exploratórios sobre a taxa de sucesso entre os anos de 2015 - 2024 na disciplina de Fundamento de Programação

Este estudo foi conduzido com base em informações anonimizadas extraídas do Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA). A análise abrangeu o período de 2015.1 a 2024.1, permitindo a comparação entre diferentes fases do uso de LLMs no aprendizado de desenvolvimento de software. O período de 2015.1 a 2019.2 foi considerado como linha de base, uma vez que, de acordo com Yao *et al.* (2024), os primeiros avanços significativos dos LLMs ocorreram a partir de 2022, especialmente com o surgimento do ChatGPT. Contudo, o ano de 2022 foi marcado pela pandemia, o que torna a análise da taxa de sucesso nesse período difícil de comparar diretamente com os demais. O contexto de ensino remoto e as condições adversas enfrentadas pelos alunos, como a falta de suporte adequado, tornam esse período atípico para análise. Por esse motivo, a linha de base para a análise, sem a interferência de LLMs, foi estabelecida entre 2015.1 e 2019.2.

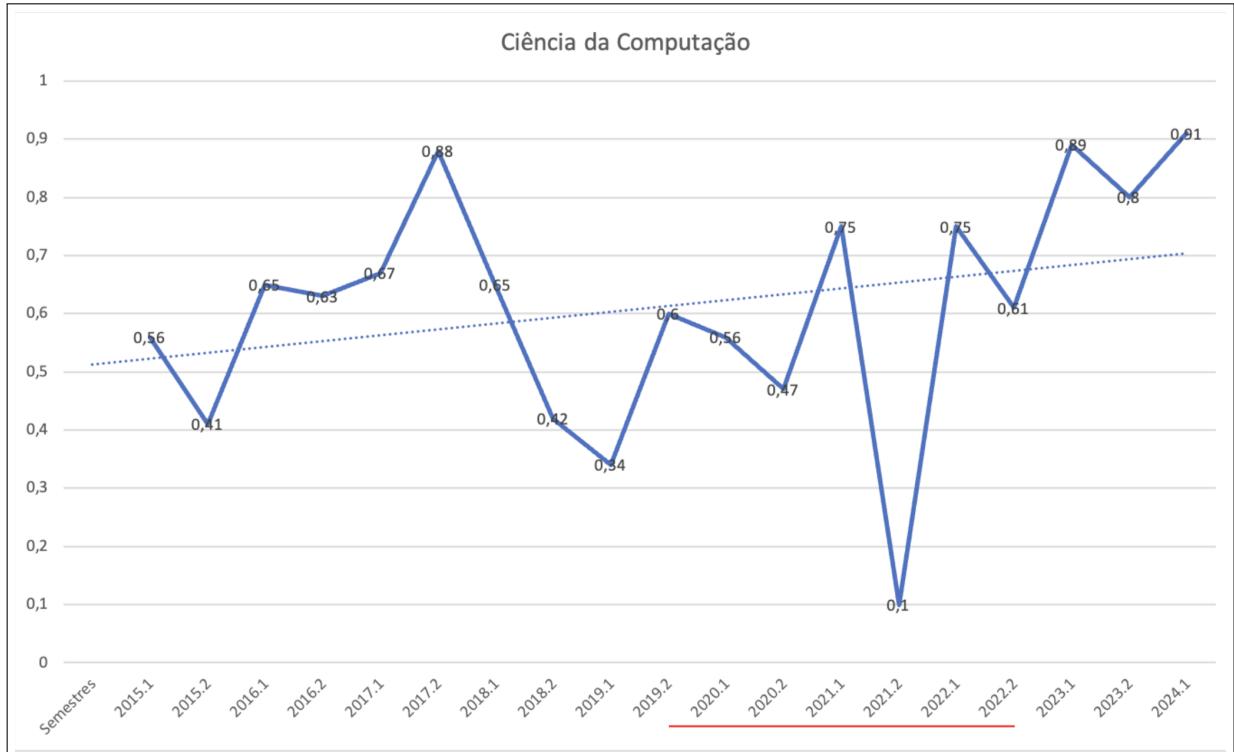
Dessa forma, esse intervalo representa uma fase sem a influência significativa dessas ferramentas. O período de pandemia, e consequentemente de ensino remoto, na UFC - Quixadá, ocorreu entre 2020.1 até 2022.2, marcada nos gráficos com uma linha vermelha, enquanto as taxas de sucesso que podem ter influência dos LLMs se estendeu de 2023.1 a 2024.1. Esse último período é caracterizado pelo aumento da popularidade dos LLMs no mercado e pela sua integração mais ativa no desenvolvimento de software. A comparação entre esses dois períodos permite identificar possíveis tendências ou correlações sobre o impacto dessas tecnologias no processo de aprendizagem.

Essa abordagem possibilitou identificar possíveis taxas de crescimento ou declínio nos índices de aprendizagem ao longo do tempo, considerando a taxa de sucesso na disciplina

de FUP/IPD nos cursos de Redes de Computadores, Engenharia de Software, Engenharia de Computação, Sistemas de Informação, Ciência da Computação e Design Digital. Para o cálculo dessa taxa, foi considerado o número de aprovações na disciplina dividido pelo total de matrículas, incluindo estudantes reprovados por nota, por falta e os suprimidos, mas excluindo aqueles com matrícula cancelada. Esse método permitiu determinar o percentual de sucesso dos alunos ao longo do período analisado.

Na Figura 5 é apresentado um gráfico com informações do curso de Ciência da Computação que apresenta a taxa de sucesso na disciplina de FUP, mostrando uma linha de tendência crescente ao longo do tempo. Até 2020.2, observamos oscilações naturais devido à reincidência de reprovações.

Figura 5 – Taxa de sucesso do curso de Ciência da Computação na disciplina de FUP



Fonte: elaborada pelo autor.

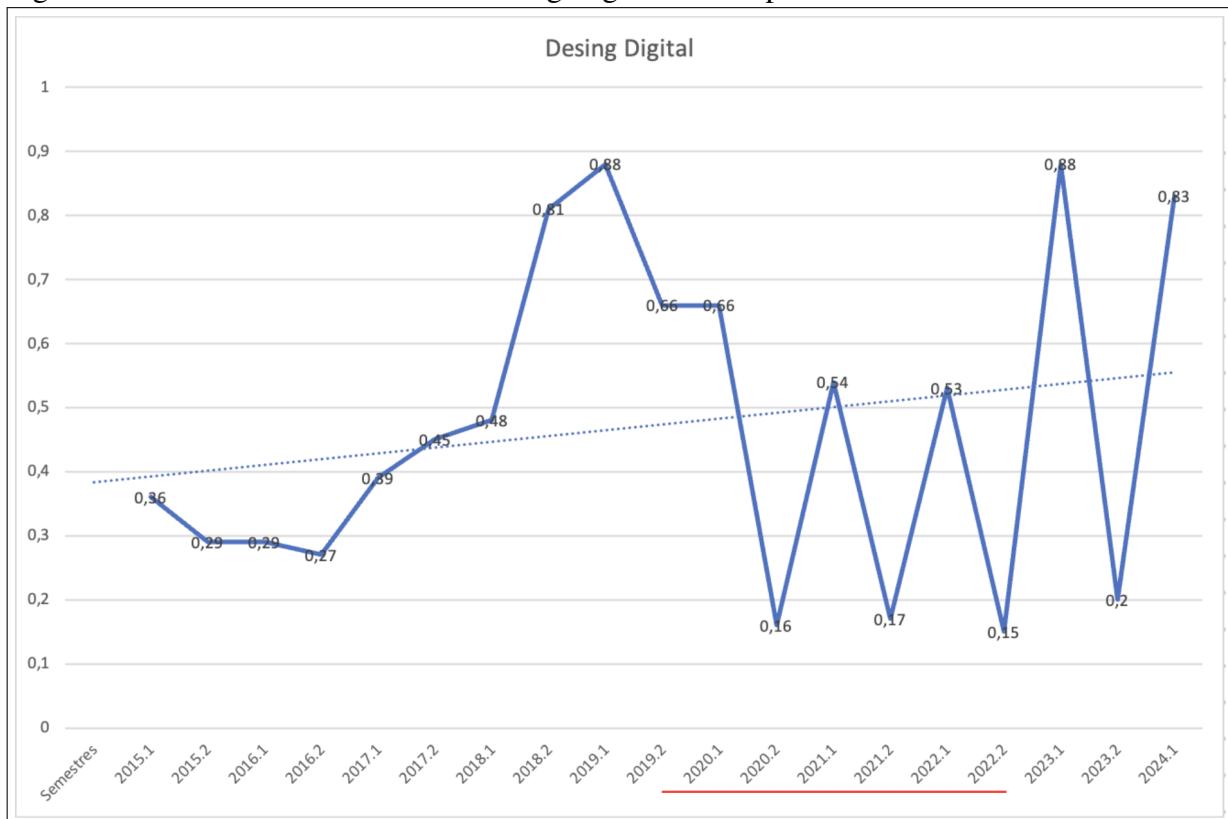
No semestre par, as turmas geralmente são compostas por alunos de diferentes cursos, pois aqueles que foram aprovados em FUP no semestre ímpar seguem para outras disciplinas, enquanto os reprovados formam novas turmas mistas, reunindo estudantes de dos outros cursos de graduação do campus que estão na mesma situação. Além disso, como o semestre par não é o período regular para cursar FUP, os alunos que tentam refazê-la já estão no segundo semestre do curso, conciliando essa disciplina com outras mais avançadas. Esse acúmulo de conteúdos pode

dificultar o aprendizado e impactar negativamente a taxa de sucesso.

Em 2021.2, possui uma queda acentuada no índice, possivelmente relacionada ao período da pandemia, quando o ensino remoto foi adotado. Esse cenário trouxe desafios significativos mediante a pandemia. A partir desse período, até 2024.1, observamos um crescimento expressivo nos índices.

Na Figura 6, observa-se um gráfico da taxa de sucesso do curso de Design Digital, que apresenta uma tendência crescente ao longo dos anos. Entre 2015.1 e 2018.1, as variações na taxa de sucesso foram mínimas. A partir de 2018.1, começaram a ocorrer oscilações mais significativas, com um aumento expressivo nos índices. Entre 2020.1 e 2022.2, houve uma queda nos índices, tanto nos semestres ímpar quanto nos semestres pares. A partir de 2023.1 teve novos picos em seus índices, chegando a 2024.1 com uma taxa de 0,83.

Figura 6 – Taxa de sucesso do curso Desing Digital na disciplinas de IPD

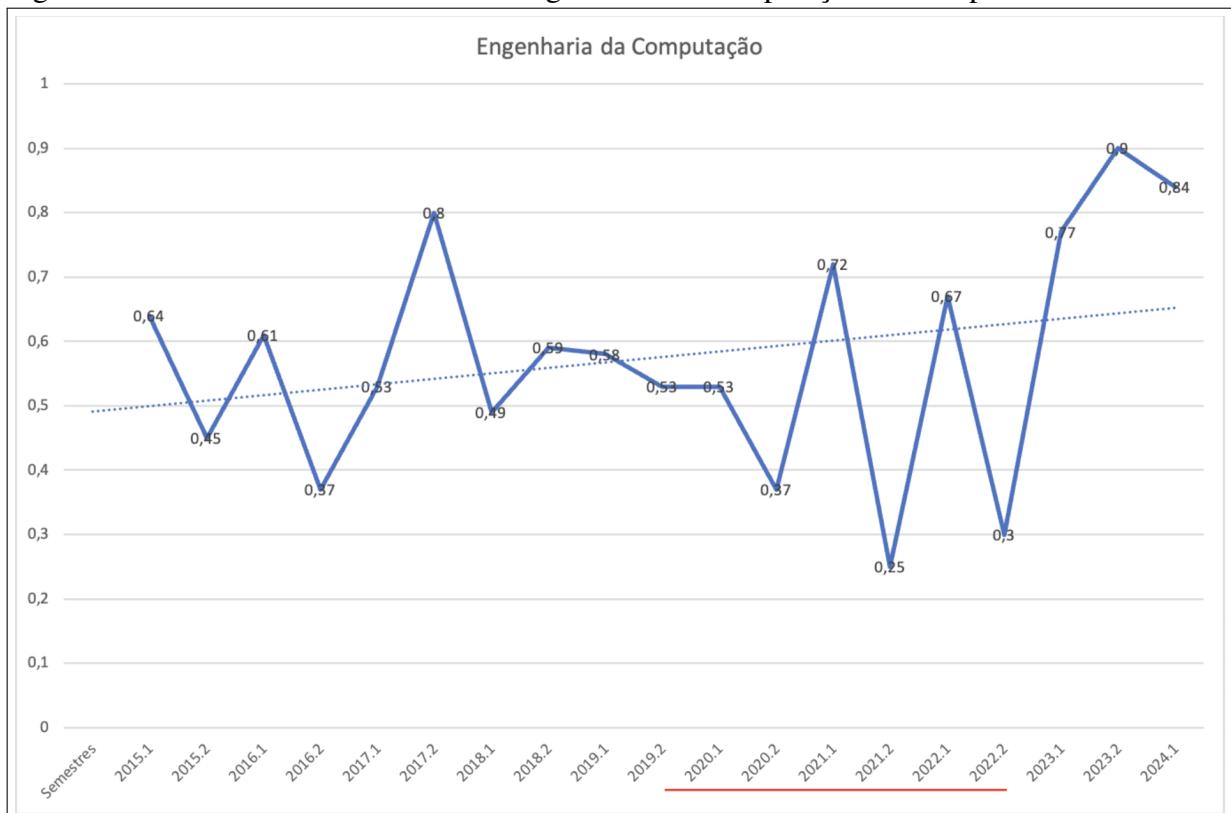


Fonte: elaborada pelo autor.

A Figura 7 apresenta o gráfico da taxa de sucesso do curso de Engenharia da Computação, evidenciando uma tendência geral de crescimento ao longo dos anos, apesar de pequenas oscilações. Entre 2015.1 e 2018.1, observa-se um período de instabilidade. Já entre 2018.2 e 2020.2, a taxa apresentou uma queda em relação aos anos anteriores. A partir de 2021.1

até 2024.1, os índices voltaram a se estabilizar em patamares semelhantes aos períodos anteriores, com variações equivalentes. O maior aumento registrado ocorreu em 2023.2, quando a taxa atingiu 0,9.

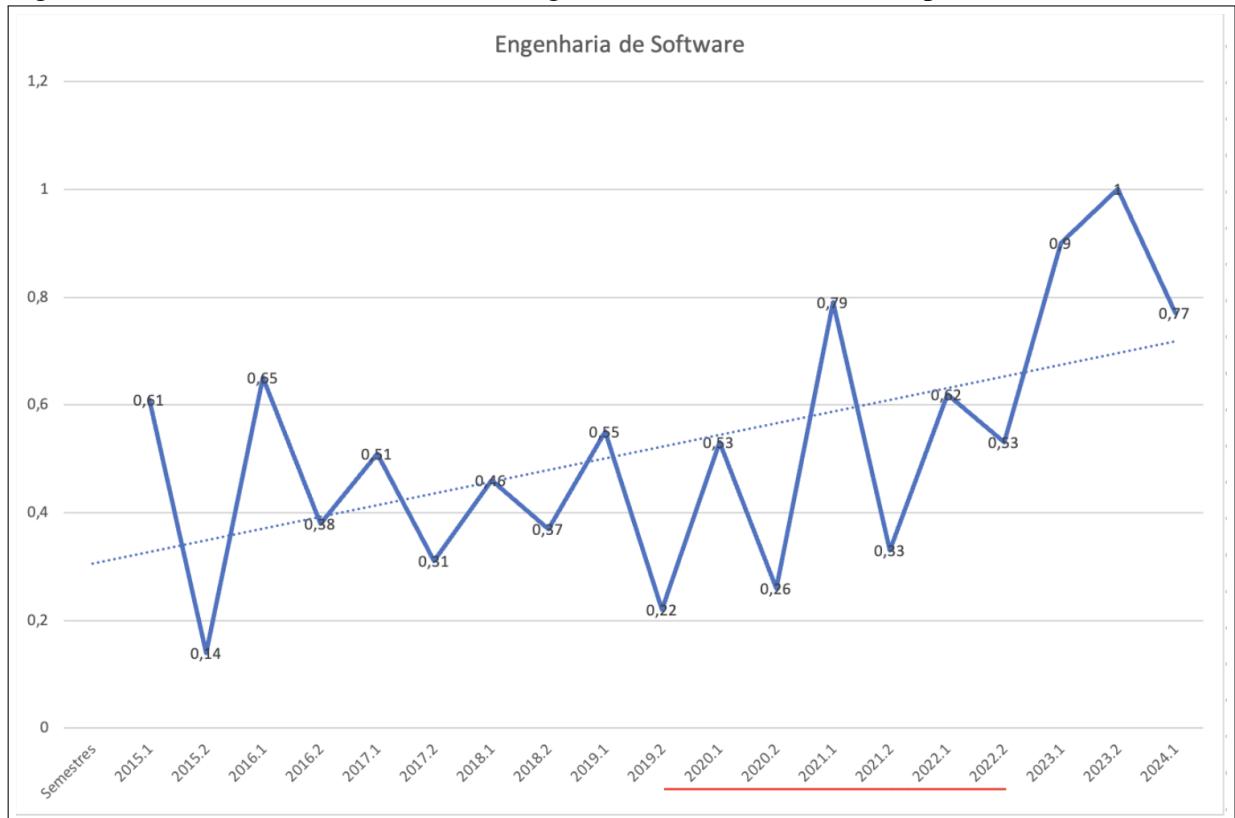
Figura 7 – Taxa de sucesso do curso de Engenharia da Computação na disciplina de FUP



Fonte: elaborada pelo autor.

A Figura 8 apresenta o gráfico da taxa de sucesso do curso de Engenharia de Software, destacando uma tendência geral de crescimento ao longo dos anos, apesar de pequenas oscilações. Observa-se que, em 2021.1, a taxa atingiu seu maior valor até então, chegando a 0,79. A partir de 2022.2, o crescimento se intensificou, culminando em 2023.1 com uma taxa de sucesso de 0,9.

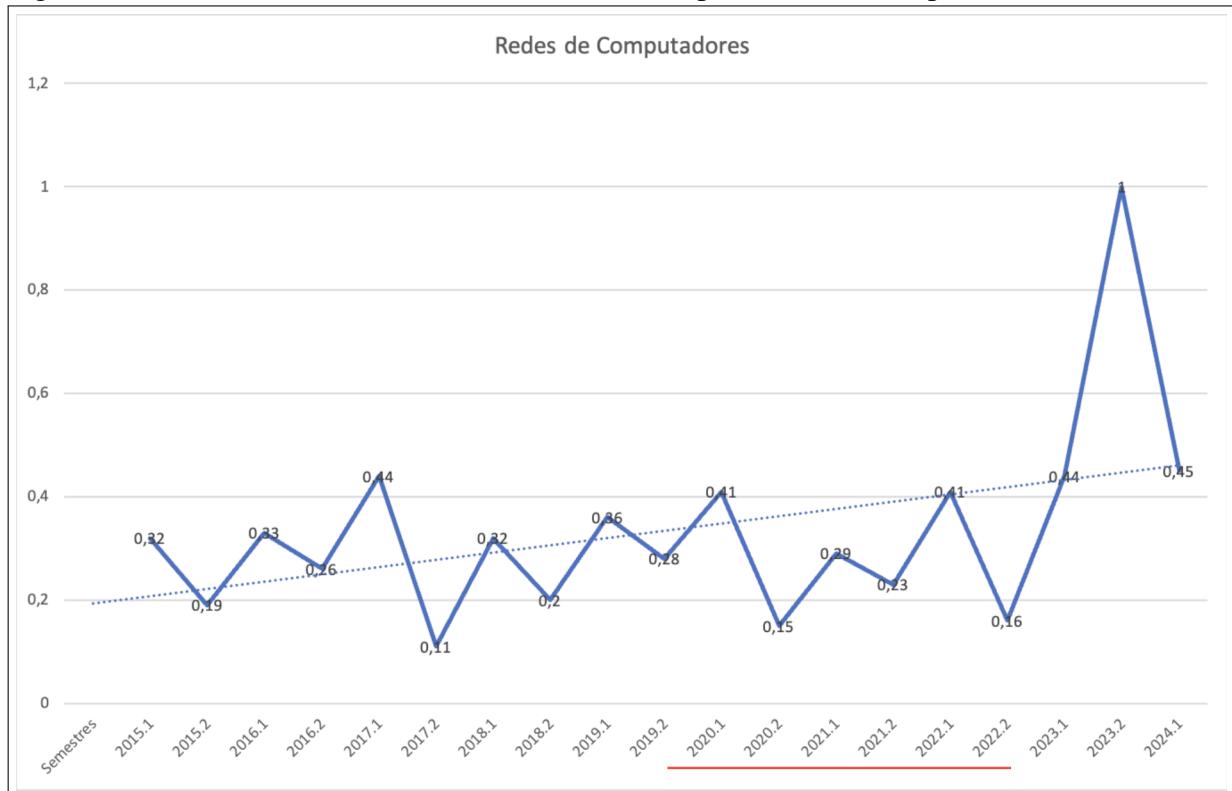
Figura 8 – Taxa de sucesso do curso de Engenharia de Software na disciplina de FUP



Fonte: elaborada pelo autor.

A Figura 9 apresenta o gráfico da taxa de sucesso do curso de Redes de Computadores. O gráfico exibe uma tendência crescente ao longo dos anos, com taxas de sucesso relativamente estáveis e sem grandes variações. A partir de 2022.2 os índices cresceram consideravelmente, chegando, em 2023.2, a uma taxa de sucesso igual a 1, ou seja, sem reprovações na disciplina de FUP.

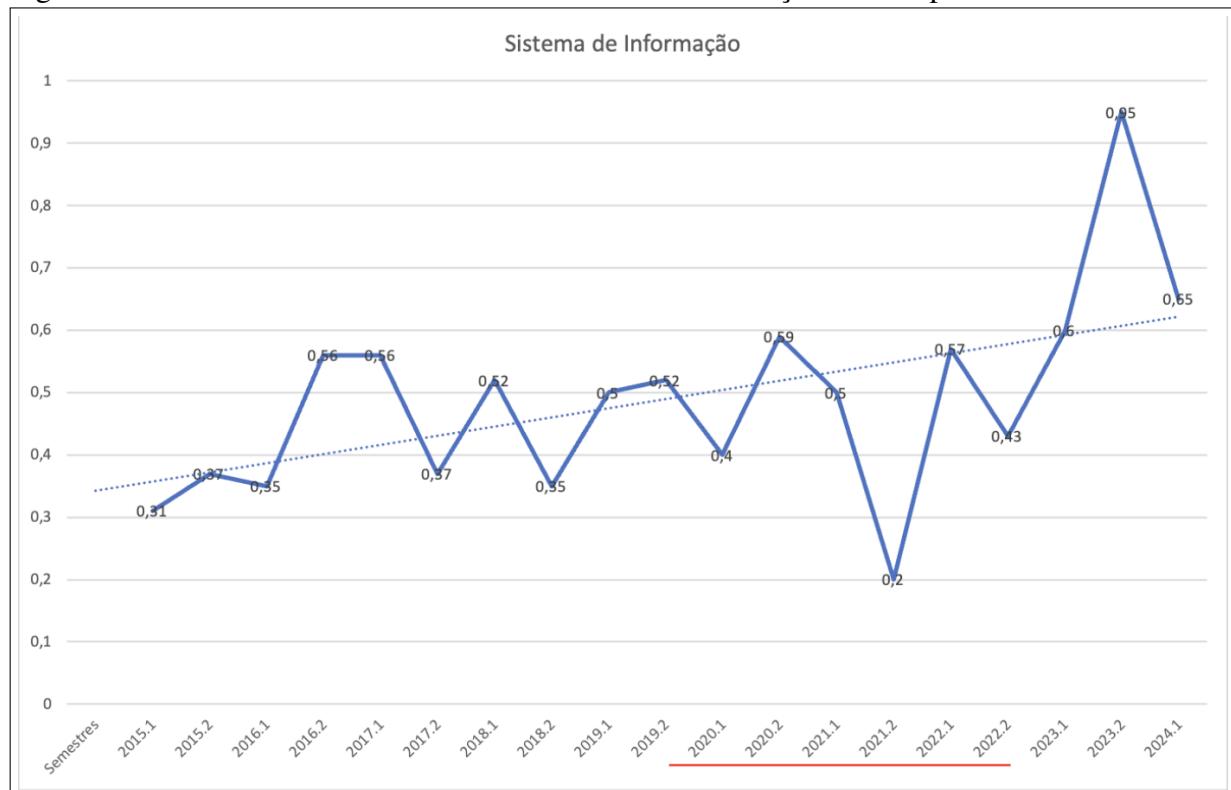
Figura 9 – Taxa de sucesso do curso de Redes de Computadores na disciplina de FUP



Fonte: elaborada pelo autor.

A Figura 10 apresenta o gráfico da taxa de sucesso do curso de Sistemas de Informação. O gráfico mostra uma tendência crescente ao longo dos anos, com taxas relativamente estáveis entre 2015.1 e 2021.1, apresentando apenas pequenas variações entre semestres pares e ímpares. Em 2021.2, observa-se uma queda acentuada na taxa de sucesso. A partir de 2022.2 as taxas passaram a crescer de forma consistente, com um aumento expressivo em 2023.2, registrando o maior índice do período. Esse crescimento em um semestre par é particularmente relevante, pois historicamente as taxas de sucesso nesses semestres costumam ser inferiores às dos semestres ímpares.

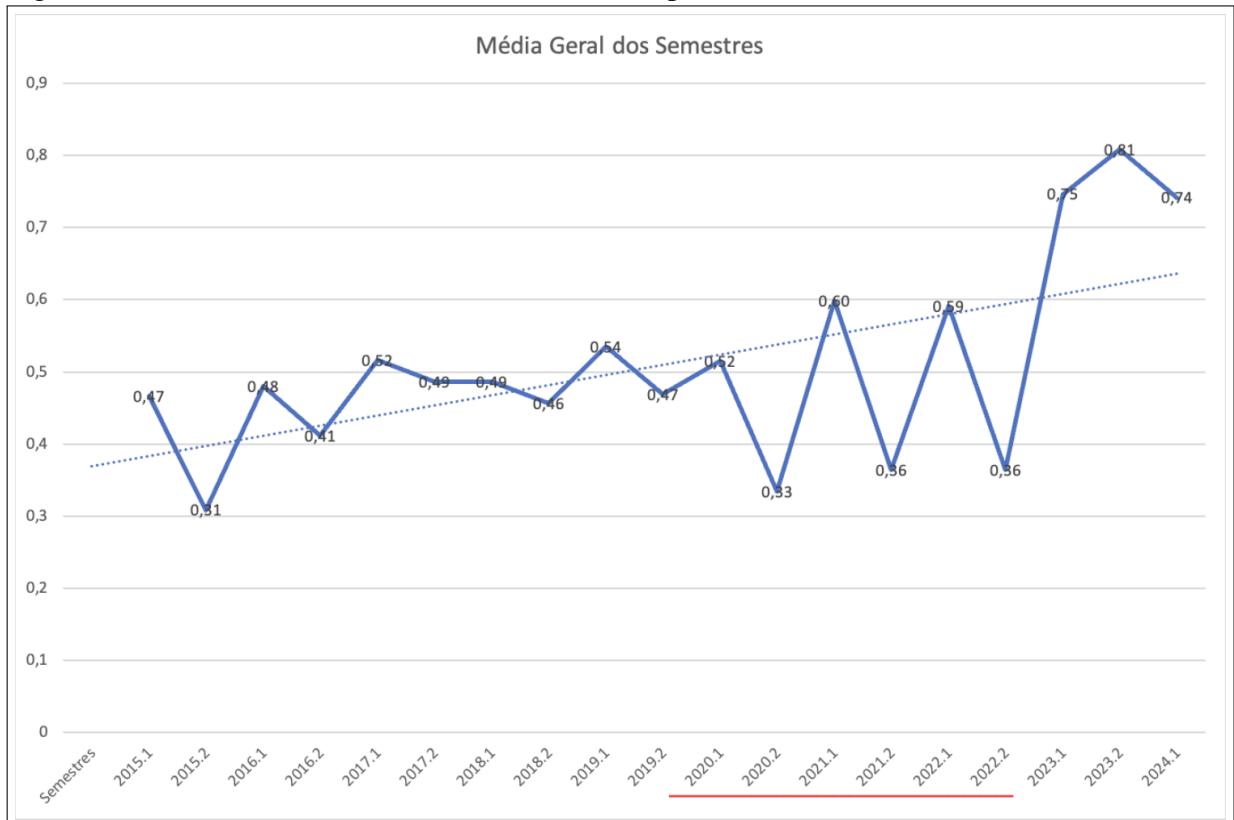
Figura 10 – Taxa de sucesso do curso de Sistema de Informação na disciplina de FUP



Fonte: elaborada pelo autor.

A Figura 11 apresenta a média geral da taxa de sucesso das disciplinas de FUP e IPD em todos os cursos de graduação. A análise dessa figura revela o impacto significativo da pandemia da COVID-19 nos índices de sucesso, especialmente entre os semestres 2020.1 e 2022.2. Em 2020.2, observa-se uma queda acentuada na média, atingindo 0,33, possivelmente devido às dificuldades enfrentadas na transição para o ensino remoto, como também pelo fato de ser um semestre par. No entanto, já em 2021.1, houve uma recuperação expressiva, com a média subindo para 0,60, sugerindo uma adaptação dos alunos e professores ao novo formato.

Figura 11 – Média Geral da taxa de sucesso das disciplinas FUP e IPD de todos os cursos



Fonte: elaborada pelo autor.

No período pós-pandemia, de 2023.1 a 2024.1, percebe-se uma recuperação consistente, com crescimento contínuo e picos em 2023.2 (0,81) e 2024.1 (0,74). Esses resultados sugerem uma estabilização no processo de aprendizagem e um possível uso dos LLMs para o aprendizado do desenvolvimento de software.

Outro aspecto relevante é a variação entre os semestres ímpares e pares. Como os cursos analisados possuem ingresso anual de novos alunos, os semestres ímpares concentram, em sua maioria, estudantes cursando as disciplinas pela primeira vez. Já nos semestres pares, as turmas são compostas predominantemente por alunos que reprovaron anteriormente e estão refazendo as disciplinas. A oferta dessas turmas em semestres pares busca reduzir o represamento de alunos, permitindo a retomada do percurso acadêmico sem que seja necessário aguardar um ano inteiro para cursar novamente a disciplina.

Com o objetivo de aprofundar a análise, foram realizados estudos complementares. Após finalizar o estudo exploratório, foram conduzidas entrevistas com professores que ministraram as disciplinas de FUP/IPD em 2024.1, buscando compreender como os LLMs eram utilizados durante as aulas. Paralelamente, foi aplicado um survey aos alunos que cursaram FUP/IPD no mesmo período, para avaliar como essas ferramentas auxiliavam no ensino e sob a

ótica dos alunos, se contribuíam para o aprendizado. Esses estudos adicionais foram essenciais para preencher as lacunas deixadas pelos dados quantitativos do SIGAA e oferecer uma compreensão mais abrangente sobre os fatores que podem ter influenciado o aumento da taxa de sucesso nas disciplinas de programação.

5.2 Resultado das entrevistas feita com os professores que ministraram FUP/IPD em 2024.1

As entrevistas com os professores proporcionaram uma visão sobre o uso de LLMs no ensino de programação. Para este estudo, foram entrevistados quatro dos seis docentes que ministraram as disciplinas de FUP/IPD no semestre 2024.1. A escolha desse recorte temporal deve-se à disponibilidade dos professores para participar das entrevistas, bem como à realização simultânea do survey com alunos que cursaram a disciplina de FUP/IPD no período de 2024.1, permitindo uma análise mais alinhada com o contexto. Além disso, esse período facilita a identificação das turmas e contribui para a qualidade das respostas dos alunos, pois, tendo cursado a disciplina recentemente, eles conseguem fornecer relatos mais detalhados e precisos devido à maior clareza na recordação de suas experiências.

Todos os participantes que aceitaram o convite possuíam experiência, tendo ministrado as disciplinas de FUP/IPD mais de dez vezes. Essa percepções dos professores em sala de é importante, já que os participantes possuem experiência sobre o impacto dessas ferramentas ao longo dos anos. As entrevistas foram conduzidas remotamente via Google Meet, com duração média de 40 minutos cada, e foram gravadas com a devida autorização dos participantes. Após a realização das entrevistas, os vídeos foram convertidos em áudio por meio de uma ferramenta nativa do sistema operacional iOS. Em seguida, a transcrição foi gerada utilizando a plataforma de inteligência artificial Restream.io. Ao final do processo, todas as transcrições passaram por uma revisão para garantir sua precisão e fidelidade ao conteúdo original. Esses textos transcritos serviram como base para a análise detalhada das falas dos professores.

5.2.1 Perfil dos docentes

Nesta subseção, é apresentado um quadro com o perfil de cada professor, destacando como eles interpretam o uso das ferramentas de LLMs no aprendizado do desenvolvimento de software.

Quadro 2 – Perfil dos professores

Entrevistados	Características
Professor 1	<ul style="list-style-type: none"> - Já ministrou a disciplina de FUP cerca de 12 vezes e IPD uma vez. - Os LLMs estão trazendo muitos benefícios porque eles estão gerando coisas que conseguiríamos gerar, só que de maneira muito mais rápido. - Acredita que as LLMs são ferramentas poderosas, mas que precisam ser utilizadas com discernimento. Ainda não usei esses recurso em sala de aula, mas os alunos podem usar.
Professor 2	<ul style="list-style-type: none"> - Já ministrou entre 12 a 15 turmas dessa disciplina, muitas vezes em mais de uma turma por semestre. - Demonstra um entendimento sobre LLMs, tanto no desenvolvimento quanto na identificação de textos gerados por essas ferramentas. - Orienta os alunos a utilizarem LLMs de maneira construtiva, recomendando que os empreguem para diagnosticar erros e compreender conceitos, em vez de simplesmente solicitar respostas prontas.
Professor 3	<ul style="list-style-type: none"> - Já ministrou a disciplina de FUP entre 15 a 20 vezes. - Utiliza LLMs desde os primeiros momentos da disciplina, ensinando os alunos a usá-los conscientemente. - Incentiva o uso do ChatGPT para pesquisa e compreensão de conceitos fundamentais de programação.
Professor 4	<ul style="list-style-type: none"> - Já ministrou a disciplina de FUP 7 vezes. - Não proíbe o uso de LLMs, desde que seja feito de forma consciente. - Recomenda o uso dos LLMs, por ser um facilitador de respostas, evidenciando a forma correta de se usar.

Fonte: Elaborado pelo autor.

5.2.2 *Políticas de Uso*

Os professores adotaram diferentes abordagens em relação ao uso de LLMs em sala de aula. O Professor 3 permitiu o uso dessas ferramentas tanto em sala quanto em atividades, desde que seu uso fosse devidamente documentado. Ele destacou que "os alunos poderiam utilizar a ferramenta, mas precisavam registrar como ela foi empregada e, posteriormente, refazer as atividades sem o suporte para consolidar o aprendizado". Essa abordagem flexível visava encorajar os estudantes a desenvolverem sua autonomia, permitindo o uso de LLMs como um recurso adicional, e não como um substituto para a prática ativa de programação.

Durante avaliações, contudo, as restrições eram mais rigorosas. Os Professores 1 e 4 optaram por aplicar provas no papel e monitorar os alunos de perto para evitar o uso inadequado das ferramentas. Foi mencionado que questões mais desafiadoras e baseadas em raciocínio lógico foram incluídas para testar a real compreensão dos conceitos pelos alunos. As políticas de restrição são acompanhadas de orientações claras sobre ética e plágio, reforçando que o objetivo principal das avaliações é refletir o aprendizado real, não apenas a capacidade de utilizar ferramentas externas.

5.2.3 *Impacto no Aprendizado*

Os LLMs foram descritos pelo Professor 2 como facilitadores do aprendizado autônomo, oferecendo suporte em momentos em que o aluno estuda sozinho. O Professor 3 relata que ao usar LLMs para esclarecer dúvidas ou corrigir erros, os alunos conseguem avançar em atividades que antes pareciam ser difíceis de resolver. "Essas ferramentas oferecem respostas claras e imediatas, ajudando o aluno a progredir em seu ritmo e, muitas vezes, a entender tópicos complexos de forma mais prática", afirmou o Professor 3.

Por outro lado, todos os professores entrevistados também apontaram riscos, como a dependência excessiva dos LLMs. Muitos alunos utilizam as ferramentas para resolver problemas sem se esforçar para compreender os conceitos envolvidos. Como comentou o Professor 3, "Os alunos precisam aprender a discernir se a resposta fornecida é correta ou não, mas muitos ainda não possuem essa habilidade". Essa falta de discernimento pode levar a adoção de respostas incorretas e dificultar a consolidação do aprendizado, especialmente em disciplinas que exigem raciocínio lógico e autonomia.

Em relação ao impacto dos LLMs nas taxas de aprovação e reprovação o Professor 3

argumenta que os LLMs não desestabilizam a estrutura da disciplina, pois são tratados como ferramentas de apoio dentro do próprio processo de ensino. Ele enfatiza que os alunos são orientados a declarar o uso dessas ferramentas e que o aprendizado envolve repetição e revisão, com um sistema de progressão estruturado. Assim, seu ponto de vista sugere que o uso dos LLMs, por si só, não altera significativamente os índices de aprovação, pois a metodologia já prevê formas de avaliar a compreensão real dos alunos. Por outro lado, o Professor 1 acredita que houve um aumento na taxa de aprovação e que os LLMs tem alguma contribuição nesse sentido. Ele argumenta que essas ferramentas fornecem um suporte adicional para os alunos, especialmente para aqueles que estudam sozinhos, ao oferecer respostas e validação de conhecimento. No entanto, ele não quantifica esse impacto e reconhece que outros fatores podem estar envolvidos.

Já os Professor 2 e 4 adota uma abordagem mais cautelosa e baseada em dados. Ele reconhece que podem ter havido uma alteração nas taxas, mas não acredita que tenha sido significativa. Para eles, a melhor maneira de responder a essa pergunta seria comparando os períodos antes e depois da popularização dos LLMs. De modo geral, as respostas indicam que os professores reconhecem o impacto dos LLMs na forma como os alunos estudam e resolvem exercícios, mas não há um consenso sobre o quanto isso tem alterado as taxas de aprovação e reaprovação. Enquanto um professor percebe uma possível contribuição positiva, outro defende que o efeito é mínimo.

5.2.4 Dificuldades Mais Frequentes

Os tópicos mais desafiadores identificados pelo Professor 1 incluem laços de repetição, vetores, matrizes e interpretação de enunciados. Esses conteúdos frequentemente levam os alunos a buscar ajuda nos LLMs. O Professor 1 explicou que “É comum os alunos consultarem a ferramenta para identificar erros em loops ou entender como manipular matrizes, que são tópicos geralmente mais complexos para iniciantes”. As ferramentas servem como um apoio importante para superar essas dificuldades iniciais.

Além disso, os LLMs são usados para reformular enunciados ou fornecer explicações mais detalhadas sobre problemas complexos. No entanto, foi observado pelo Professor 3 que muitos estudantes ignoram etapas essenciais do processo de resolução, como planejar a solução antes de codificar. Essa abordagem impulsiva pode comprometer a aprendizagem e reforçar a dependência das ferramentas, ao invés de incentivar a construção de habilidades críticas.

5.2.5 Metodologias de Avaliação

Os professores 3 e 4 implementaram metodologias para monitorar o uso de LLMs e avaliar o aprendizado dos alunos de forma justa. Uma das estratégias foi utilizar plataformas como GitHub para acompanhar o progresso individual e identificar padrões de uso inadequado. “O histórico de interação dos alunos com as atividades nos dá pistas sobre como eles chegaram a solução e se recorreram a ferramentas externas”, explicou o Professor 4. Outro método consistiu em pedir que os alunos refizessem atividades previamente realizadas com LLMs, incentivando a internalização dos conceitos.

Nas avaliações formais, as restrições foram mais severas. Provas impressas ou monitoramento em sala de aula para provas usando computadores foram adotados para evitar o uso de LLMs durante as provas. Essas estratégias não apenas reduzem o plágio, mas também asseguram que os resultados representem o aprendizado real dos alunos, sem a influência direta das ferramentas externas, destaca o Professor 4.

5.2.6 Perspectivas Futuras

Os LLMs tem um grande potencial para transformar o ensino de programação, desde que sejam integrados de forma consciente ao processo de aprendizado, ressalta o Professor 1. Os Professores 3 e 4 enfatizam a importância de os alunos compreenderem como utilizar essas ferramentas para consolidar o aprendizado. “O uso dos LLMs deve ser acompanhado de maturidade e senso crítico, para que complementem o ensino em vez de substituírem o esforço individual”, observa o Professor 3.

Uma das prioridades é ensinar os alunos a usar os LLMs para fortalecer o pensamento computacional. Isso envolve orientá-los sobre como formular perguntas de forma eficaz, avaliar criticamente as respostas e integrar os resultados ao processo de aprendizado. O sucesso dessa abordagem depende tanto da orientação dos professores quanto da conscientização dos alunos, promovendo um ambiente em que as ferramentas tecnológicas sejam utilizadas de maneira responsável e produtiva para aprimorar o aprendizado, esclarece o Professor 4.

O Quadro 3 categoriza aspectos relacionados ao uso de LLMs no ensino, destacando pontos relevantes e exemplos. Ele aborda temas como benefícios e limitações, políticas de uso, impacto no aprendizado, dificuldades frequentes, metodologias de avaliação e perspectivas futuras. Além disso, apresenta exemplos que ilustram cada categoria, como a facilitação da com-

preensão, a dependência excessiva, o uso consciente e estratégias avaliativas. Essa organização sintetiza informações essenciais para a análise do impacto dos LLMs .

Quadro 3 – Quadro Expositivo

Categoria	Aspectos Relevantes	Exemplos
Uso de LLMs no Ensino.	Benefícios e limitações identificados pelos professores.	Facilitar compreensão; Dependência excessiva.
Políticas de Uso	Estratégias e regras para permitir ou restringir o uso de LLMs.	Uso consciente incentivado; Proibição em provas.
Impacto no Aprendizado	Efeitos das LLMs na compreensão e desempenho dos alunos.	Suporte imediato; Risco de respostas erradas
Dificuldades mais Frequentes	Conteúdos que levam os alunos buscar ajuda com os LLMs.	Vetores, Matrizes, Laços de Repetição.
Metodologias de Avaliação	Práticas de Avaliação para evitar plágio e promover aprendizado.	Uso do GitHub; Provas no Papel, Moodle.
Perspectivas Futuras	Recomendações para integrar LLMs de forma produtiva no ensino.	Desenvolver orientações claras; Melhorias conscientização sobre o uso.

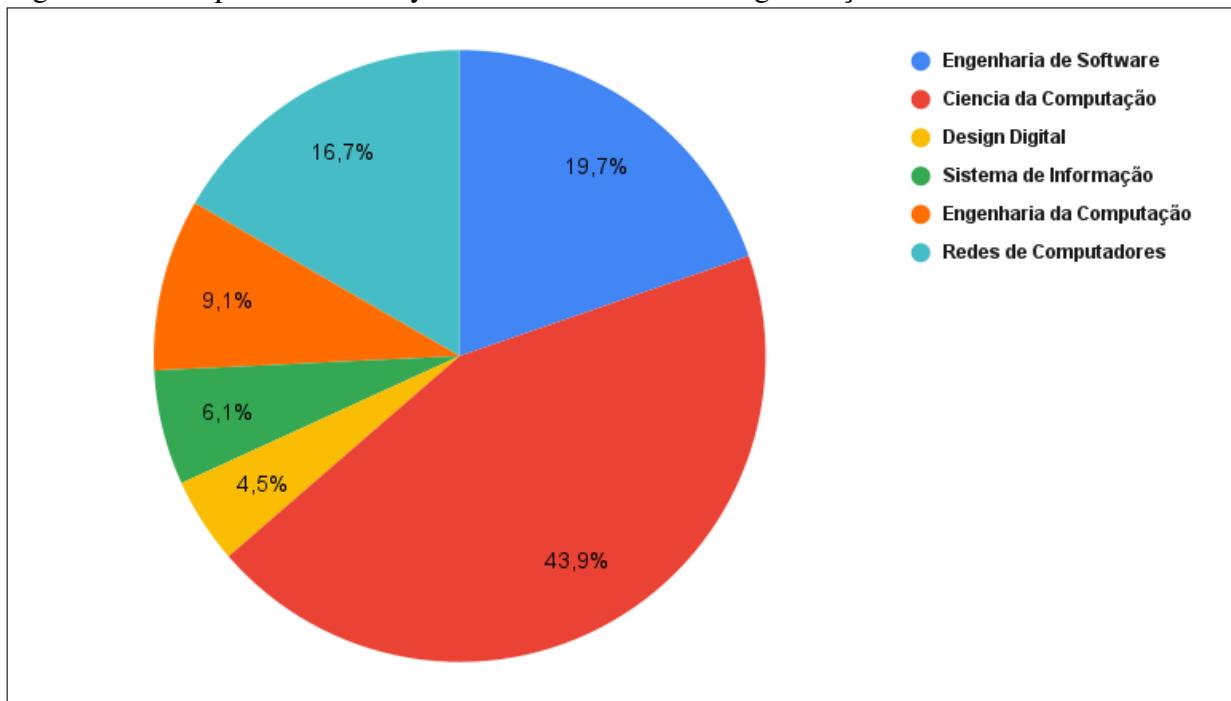
Fonte: Elaborado pelo autor.

5.3 Resultados do survey com os alunos que ingressaram no campus da UFC Quixadá em 2024.1

A pesquisa contou com 68 participantes, o que representa aproximadamente 22,6% de uma população média de 300 alunos dos seis cursos de graduação do campus da UFC Quixadá. O survey foi realizado com alunos que cursaram a disciplina de FUP/IPD no período de 2024.1, proporcionando uma análise mais alinhada ao contexto atual. Esse período também facilita a identificação das turmas, além de contribuir para a qualidade das respostas, uma vez que, ao terem cursado a disciplina recentemente, os alunos conseguem fornecer relatos mais detalhados e precisos, devido à maior clareza na recordação de suas experiências.

A Figura 12 apresenta um gráfico com os dados do survey, destacando a distribuição dos respondentes de acordo com seus cursos de graduação. Observa-se que, entre os seis cursos analisados, Ciência da Computação foi o que registrou o maior número de respostas.

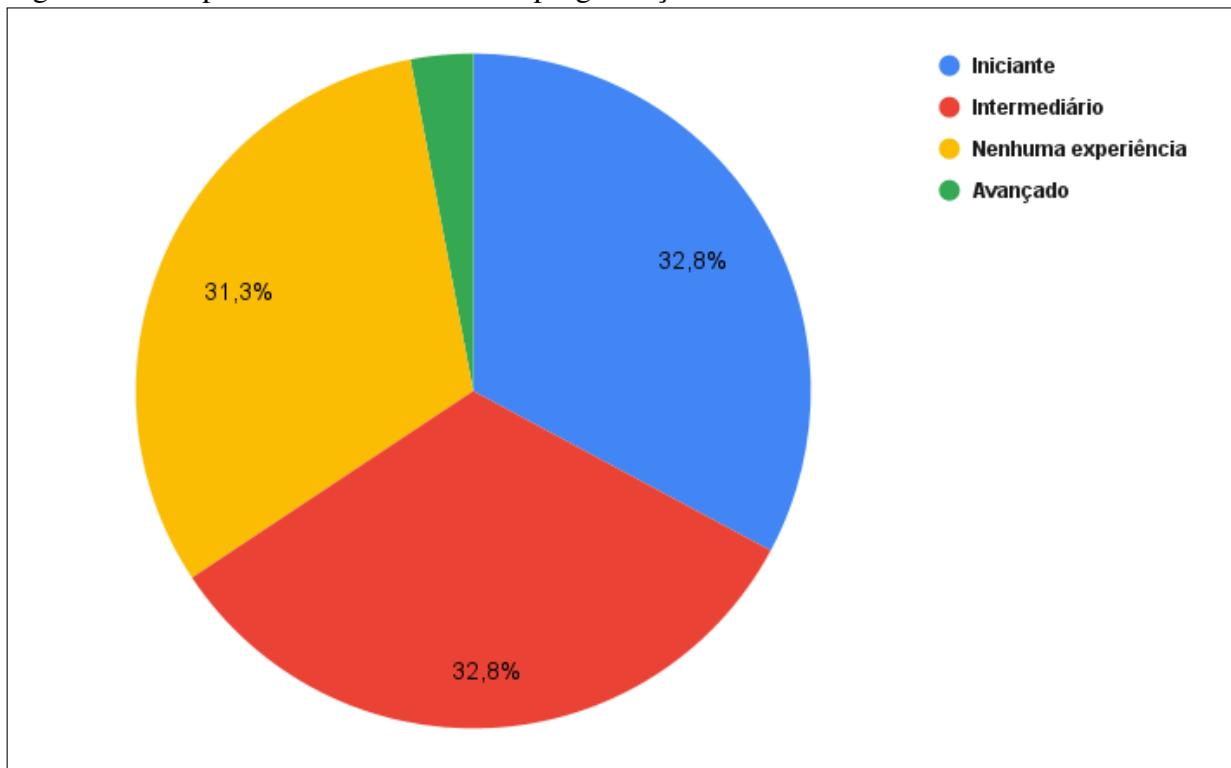
Figura 12 – Respostas do survey relacionadas ao curso de graduação.



Fonte: elaborada pelo autor.

A Figura 13 apresenta um gráfico que representa as experiências dos alunos ingressantes no campus em 2024.1, os quais cursaram a disciplina de FUP/IPD no mesmo período. Nota-se que as turmas são bastante diversificadas, abrangendo diferentes níveis de experiência em programação.

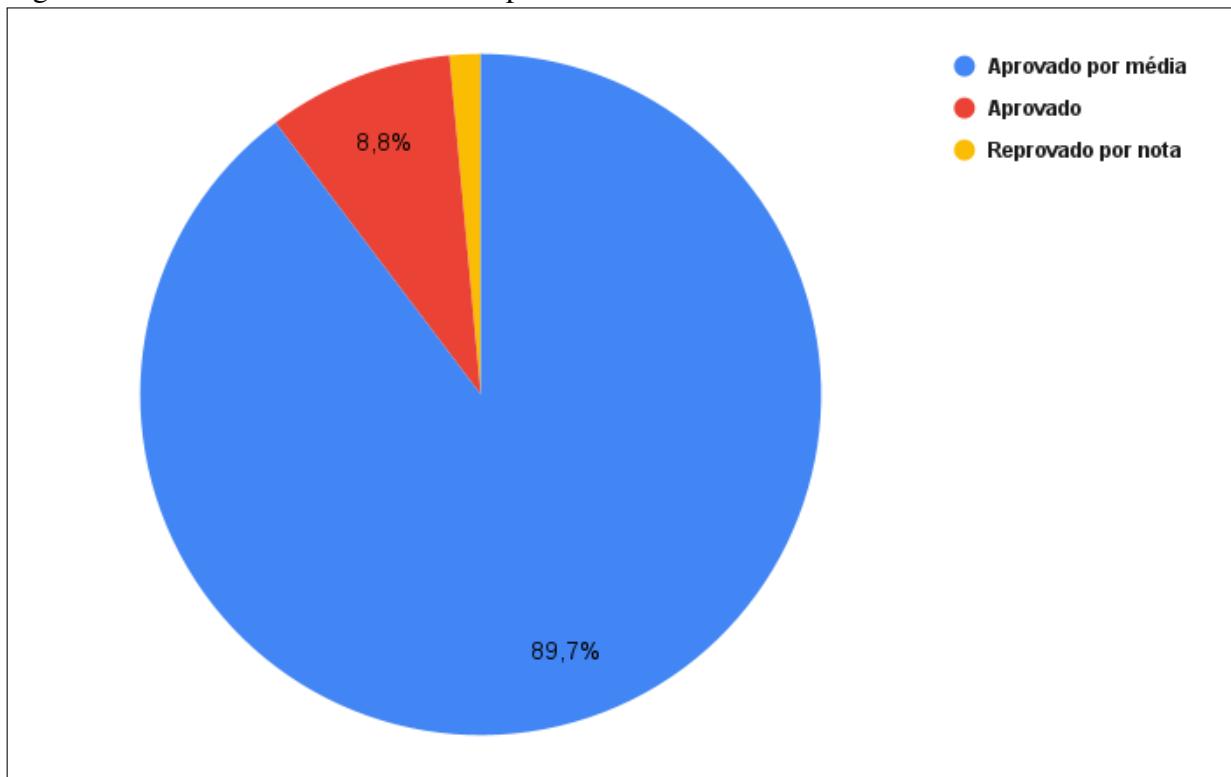
Figura 13 – Experiência dos alunos com programação



Fonte: elaborada pelo autor.

A Figura 14 apresenta um gráfico com o rendimento final da disciplina de FUP/IPD no semestre 2024.1. Observa-se um número elevado de aprovados, o que ao comparar com dados de períodos anteriores, parece inconsistente com o histórico da disciplina. Inicialmente, a coleta de respostas por meio do survey enviado via e-mail não obteve o retorno esperado. Para contornar essa limitação, adotei uma abordagem mais direta, identificando os professores responsáveis pela disciplina de Programação Orientada a Objetos (POO) e, a partir disso, localizando as turmas de alunos ingressantes em 2024.1 que atualmente cursam POO. Como FUP/IPD é um pré-requisito para essa disciplina, conclui-se que esses alunos foram aprovados em FUP/IPD, o que explica os dados apresentados no gráfico.

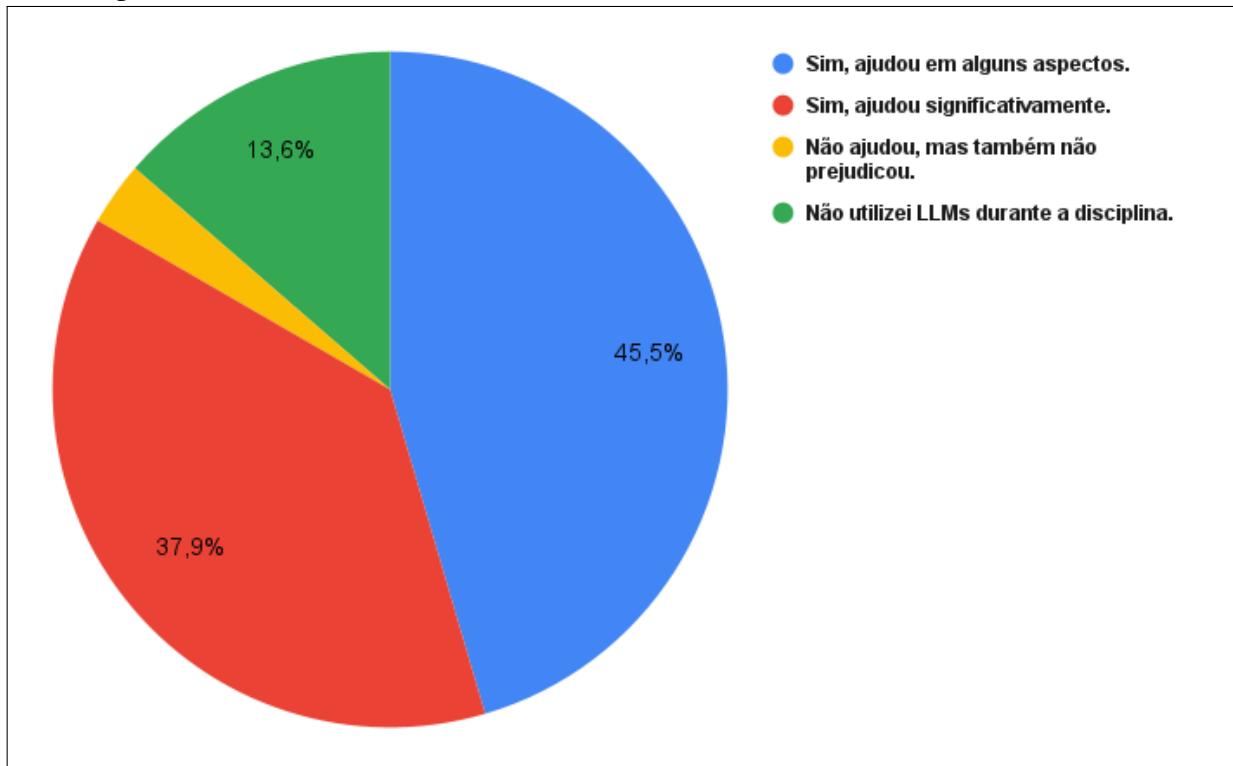
Figura 14 – Rendimento final nas disciplinas de FUP/IPD em 2024.1



Fonte: elaborada pelo autor.

O gráfico da Figura 15 apresenta as respostas sobre o quanto os LLMs auxiliam na superação de desafios no aprendizado das disciplinas de FUP/IPD. Observa-se que a opção mais selecionada foi "Sim, ajudou em alguns aspectos", o que sugere que, embora os LLMs tenham contribuído para o aprendizado, seu suporte não é absoluto. Os dados indicam que os alunos frequentemente precisam conferir e validar as respostas fornecidas pelas ferramentas de IA, demonstrando uma dependência parcial, mas não total, desses modelos no processo de aprendizagem.

Figura 15 – Impacto dos LLMs como forma de auxílio na superação de desafios no aprendizado das disciplinas de FUP/IPD

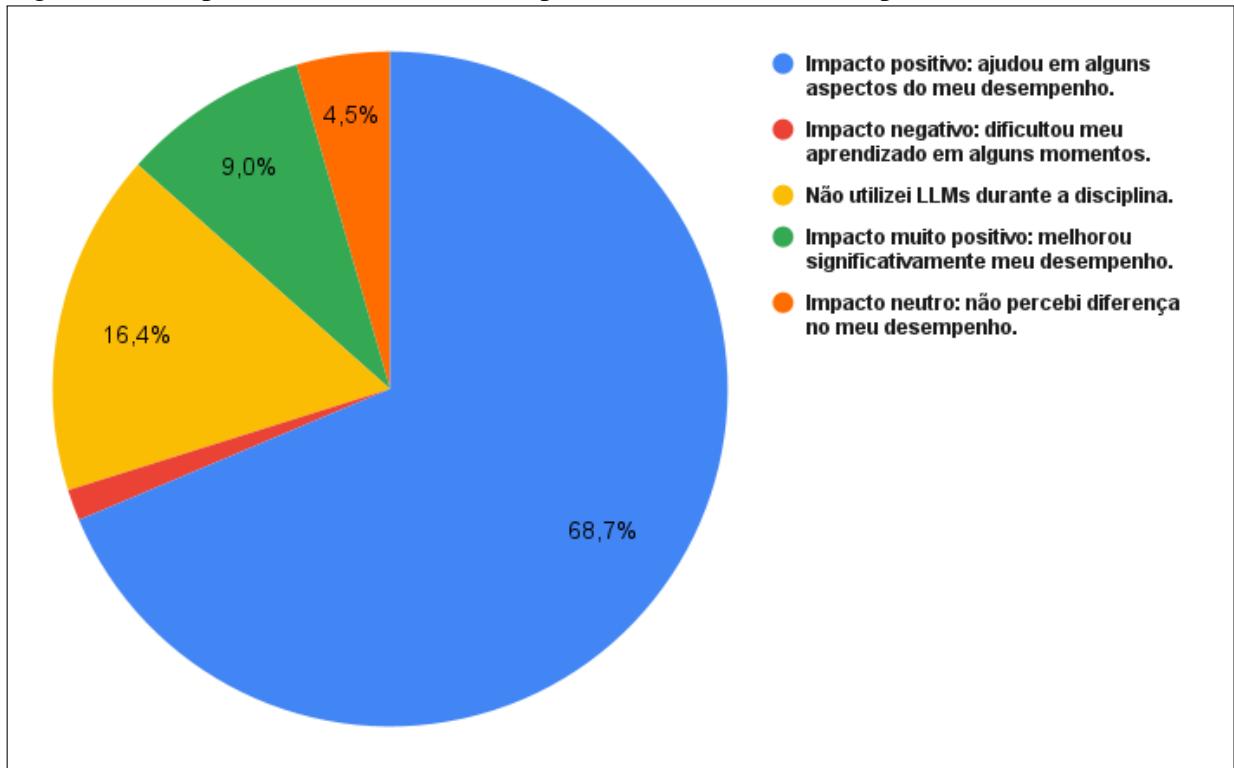


Fonte: elaborada pelo autor.

Os dados analisados mostram diferenças significativas na forma como os LLMs são utilizados, variando de acordo com o curso e o nível de experiência dos alunos. Essa variação indica que o aproveitamento dessas ferramentas está diretamente relacionado ao contexto acadêmico e ao conhecimento prévio de cada estudante. Em particular, 17,6% dos participantes apresentaram padrões distintos de uso, reforçando a influência desses fatores na adoção e na confiança nas respostas geradas pelos modelos de IA. Esse dado foi a partir do cruzamento entre a variável que indica o nível de experiência em programação dos alunos e a variável que mede a frequência e o tipo de uso dessas ferramentas no aprendizado.

O gráfico da Figura 16 ilustra o impacto das ferramentas de LLMs nas disciplinas de FUP/IPD. As respostas foram divergentes, porém um destaque significativo foi a opção "Impacto positivo: ajudou em alguns aspectos do meu desempenho", indicando que muitos alunos perceberam benefícios no uso dessas ferramentas para o aprendizado em desenvolvimento de software. Além disso, uma parcela dos participantes selecionou a opção "Não utilizei LLMs durante a disciplina", sugerindo que, enquanto alguns alunos incorporaram essas ferramentas em sua rotina acadêmica, outros optam por não utilizá-las.

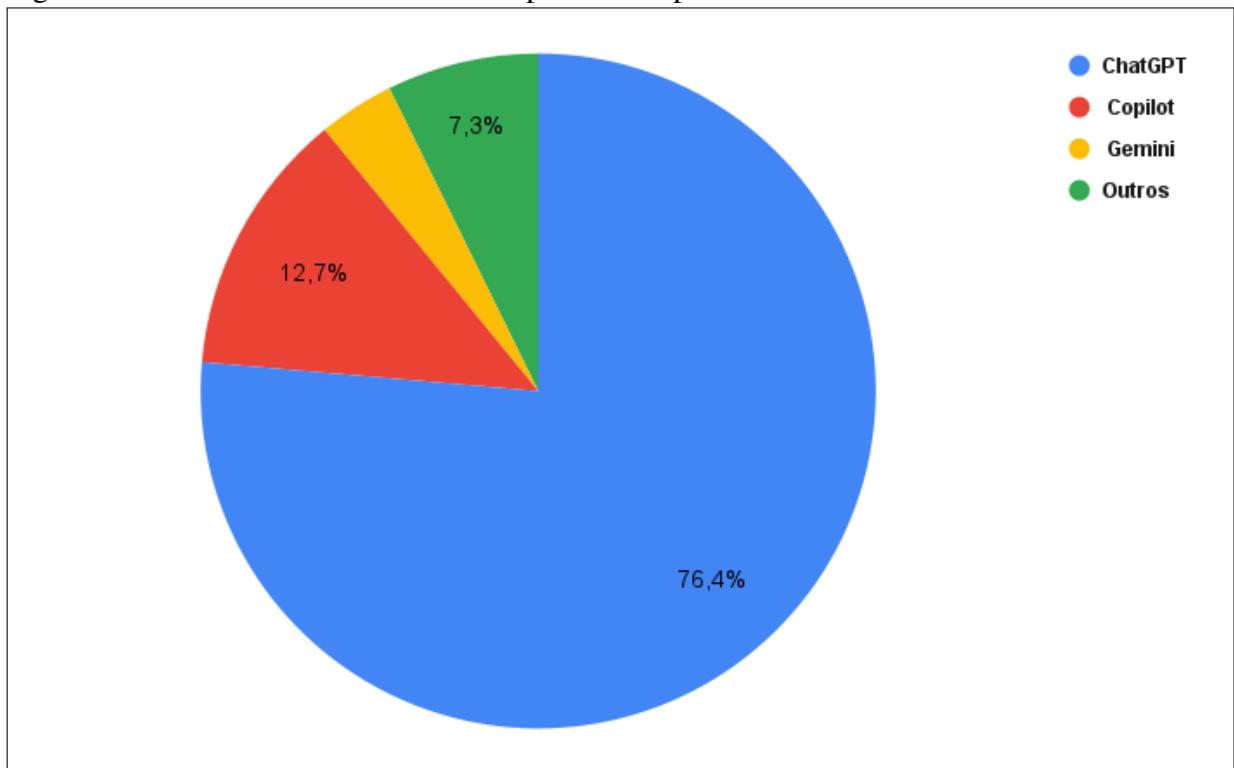
Figura 16 – Impacto dos LLMs no desempenho dos alunos nas disciplinas de FUP/IPD



Fonte: elaborada pelo autor.

O gráfico da Figura 17 apresenta a popularidade das ferramentas de LLMs entre os alunos, destacando quais são as mais utilizadas no contexto acadêmico. Observa-se que o ChatGPT foi, de longe, a ferramenta mais mencionada, evidenciando sua predominância em relação às demais opções disponíveis. Esse resultado pode ser atribuído a diversos fatores, incluindo sua acessibilidade, interface intuitiva e capacidade de compreensão e geração de código, tornando-se uma opção prática para estudantes que buscam auxílio no aprendizado de programação.

Figura 17 – Ferramenta de LLM como apoio à disciplina de FUP/IPD em 2024.1.

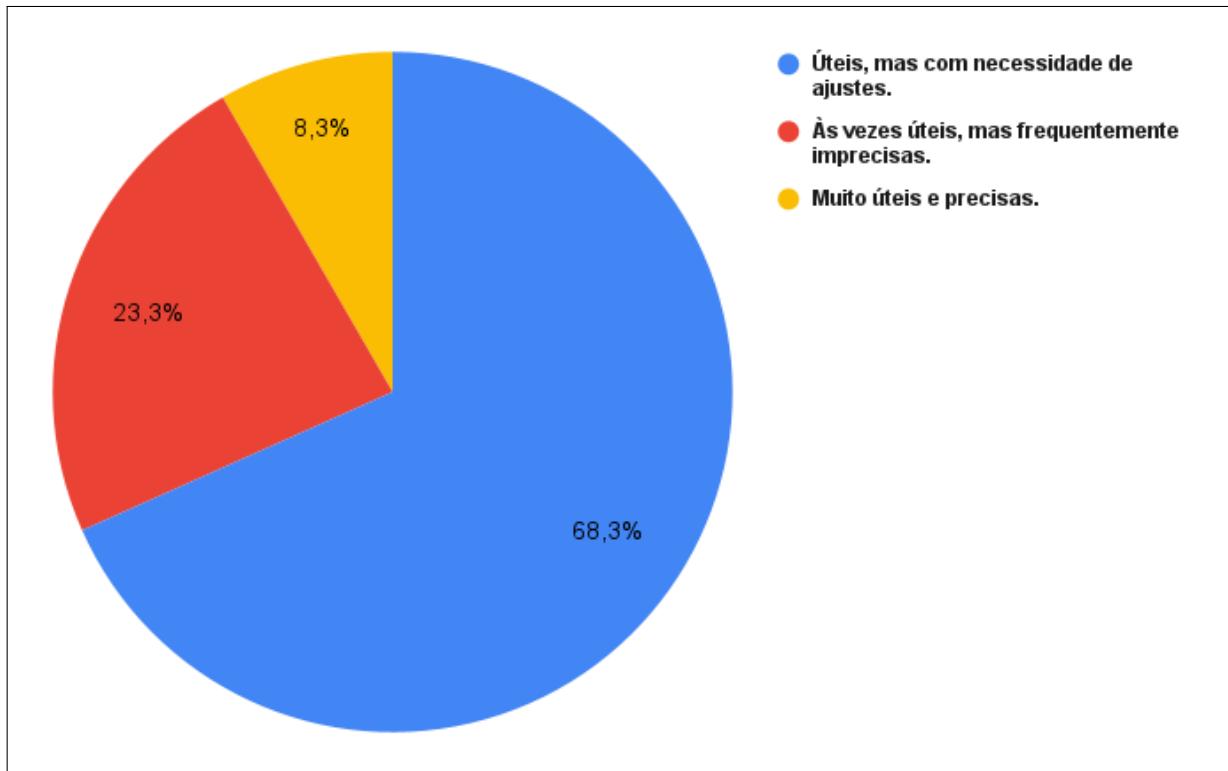


Fonte: elaborada pelo autor.

Além do ChatGPT, outras ferramentas também foram citadas, ainda que com menor frequência, indicando que há um certo nível de diversificação no uso de LLMs pelos alunos. No entanto, a ampla adoção do ChatGPT sugere que sua popularidade pode estar relacionada tanto à sua eficiência na resolução de problemas quanto à sua disseminação no meio acadêmico como uma ferramenta de suporte ao aprendizado.

O gráfico da Figura 18 apresenta as respostas obtidas no survey sobre a percepção dos alunos quanto a qualidade das respostas fornecidas pelos LLMs para solucionar problemas relacionados ao aprendizado de programação. Os resultados indicam que a opção mais selecionada foi "Útil, mas com a necessidade de ajustes", evidenciando que, embora essas ferramentas sejam vistas como auxiliares, ainda tem limitações em sua precisão e confiabilidade.

Figura 18 – Qualidade das soluções fornecidas pelos LLMs para problemas de programação em FUP/IPD



Fonte: elaborada pelo autor.

Essa predominância sugere que os estudantes reconhecem o potencial dos LLMs para oferecer suporte na resolução de problemas, mas entendem que as respostas nem sempre podem ser utilizadas diretamente sem revisão ou correções. Esse comportamento pode estar relacionado a fatores como a possibilidade de respostas imprecisas, a necessidade de adaptar as soluções ao contexto específico do problema e até mesmo a tendência dos modelos de IA de gerar códigos funcionais.

Além disso, observou-se que a confiança na qualidade das respostas fornecidas pelos LLMs teve um impacto no aprendizado. Alunos que consideraram as respostas mais precisas e úteis relataram um impacto positivo maior em seu desempenho acadêmico, com 61,7% dos participantes expressando essa percepção. Essa afirmação foi fundamentada a partir do cruzamento de dados entre a percepção da precisão das respostas fornecidas pelos LLMs e a avaliação do impacto no aprendizado. A análise revelou uma correlação entre a confiança na qualidade das respostas e a percepção de benefício acadêmico, indicando que estudantes que consideraram as respostas mais precisas e úteis relataram um impacto positivo mais significativo em seu desempenho. Esse achado reforça a ideia de que o uso crítico e estratégico dessas ferramentas pode facilitar o processo de aprendizagem, auxiliando na superação de dificuldades

e no aprimoramento das habilidades de programação.

Os resultados desta pesquisa mostram que a utilização de LLMs, em especial o ChatGPT, tem sido uma prática adotada pelos estudantes do campus da UFC Quixadá no aprendizado de programação das disciplinas de FUP/IPD no semestre de 2024.1. A predominância do uso dessas ferramentas para acessar exemplos de código, resolver problemas de codificação e compreender conceitos teóricos destaca sua importância como suporte no aprendizado. No entanto, a pesquisa feita indica desafios importantes.

Enquanto estudantes mais experientes demonstraram maior capacidade de extrair benefícios das ferramentas de LLMs, iniciantes podem encontrar dificuldades em distinguir respostas corretas de informações imprecisas, o que trás a tona a importância do pensamento crítico na interação com essas tecnologias. Esse cruzamento foi obtido a partir da análise comparativa entre o nível de experiência dos alunos e a percepção deles sobre a confiabilidade e precisão das respostas fornecidas pelos LLMs. Os dados do survey incluíram o nível de confiança nas respostas geradas e a necessidade de validação com outras fontes. Ao segmentar os participantes entre iniciantes e experientes, observou-se que alunos com maior familiaridade em programação demonstraram um uso mais criterioso dos LLMs, validando as respostas com outras fontes e utilizando as ferramentas de maneira estratégica. Já os iniciantes, que relataram menor experiência prévia, mostraram maior tendência a aceitar as respostas sem questionamento, o que pode levá-los a dificuldades na identificação de informações imprecisas.

Diante disso, no contexto da UFC, Campus Quixadá, conclui-se que os LLMs são recursos importantes para o ensino de programação, especialmente quando utilizados de forma crítica e complementada por métodos tradicionais de aprendizado. A combinação dessas ferramentas com abordagens pedagógicas convencionais pode potencializar o processo de ensino, proporcionando um equilíbrio entre inovação tecnológica e o desenvolvimento de habilidades fundamentais no aprendizado de programação. A integração dessas duas estratégias pode tornar as práticas das disciplinas de FUP/IPD mais eficazes, pois, embora a ferramenta forneça explicações ou até mesmo respostas diretas, seu verdadeiro valor está em servir como um ponto de partida para que os alunos compreendam o raciocínio por trás das soluções. Dessa forma, o aprendizado se torna mais sólido e significativo. Além disso, a orientação adequada dos professores e o desenvolvimento da capacidade crítica dos alunos são importantes para melhorar os benefícios dessas ferramentas.

5.4 Análise dos Resultados

A análise dos resultados obtidos a partir do estudo exploratório, do survey e das entrevistas com professores permite uma compreensão mais ampla do impacto dos LLMs no aprendizado de programação. O cruzamento desses dados evidencia tanto o papel dessas ferramentas no apoio aos estudantes quanto os desafios percebidos pelos professores em relação ao seu uso.

Os professores entrevistados relataram uma crescente dependência dos alunos em relação aos LLMs, especialmente entre aqueles com menos experiência. Segundo eles, tem uma tendência de os estudantes utilizarem essas ferramentas como uma solução imediata para problemas, sem necessariamente compreenderem a lógica por trás das respostas. Essa percepção se confirma nos dados do survey, que indicam que a maioria dos alunos considera as respostas dos LLMs úteis, mas com necessidade de ajustes. No entanto, um dado relevante é que aqueles que verificam e ajustam as respostas fornecidas relataram um impacto mais positivo em seu aprendizado, reforçando a hipótese de que o uso crítico dessas ferramentas pode contribuir para um melhor desempenho acadêmico.

A relação entre o nível de experiência do aluno e o uso dos LLMs também foi uma questão observada. Os dados indicam que estudantes iniciantes confiam mais nas respostas dessas ferramentas e as utilizam como apoio principal para a resolução de exercícios. Em contrapartida, alunos mais experientes tendem a recorrer aos LLMs como um suporte complementar, o que pode indicar um amadurecimento no uso da tecnologia ao longo do tempo. Os professores reforçaram essa ideia ao mencionarem que os alunos que dependem excessivamente dos LLMs demonstram dificuldades quando precisam resolver problemas sem esse suporte.

Além disso, percebeu-se que a confiança na qualidade das respostas fornecidas pelos LLMs influenciou diretamente a percepção de benefício no aprendizado, uma vez que estudantes que consideraram as respostas mais precisas também relataram um impacto positivo mais significativo, representando 61,7% dos respondentes.

O cruzamento dos dados permite uma análise mais aprofundada da hipótese levantada no estudo exploratório, que sugere que a tendência de crescimento na taxa de aprovação nas disciplinas de FUP/IPD entre os anos de 2015.1 e 2024.1 pode estar relacionada ao uso de LLMs. Os resultados obtidos indicam que essas ferramentas são utilizadas, sendo percebidas como um suporte relevante para dificuldades iniciais na programação.

Os professores entrevistados, no entanto, destacaram que o uso dos LLMs pode

gerar dependência excessiva, especialmente entre os alunos que não possuem base sólida em lógica de programação. Esse ponto se alinha com os dados do survey, nos quais a maioria dos estudantes classificou as respostas dos LLMs como “úteis, mas com necessidade de ajustes”, o que sugere que a ferramenta auxilia, mas não substitui totalmente a compreensão dos conteúdos. Essa relação entre o uso crítico dos LLMs e o impacto positivo no aprendizado também aparece nos resultados: alunos que verificam e ajustam as respostas obtidas tendem a apresentar um melhor desempenho acadêmico.

Outro fator relevante é a relação entre a experiência do estudante e a forma como ele utiliza os LLMs. Enquanto os iniciantes tendem a confiar mais nos resultados gerados e os utilizam de maneira mais direta, os alunos com mais bagagem costumam recorrer às ferramentas para complementar seu aprendizado, validando respostas e refinando seu entendimento dos conceitos. Essa diferença de abordagem pode indicar que, conforme os alunos avançam no curso e ganham mais domínio sobre a programação, o uso dos LLMs se torna mais estratégico e menos dependente.

Dessa forma, a análise dos resultados reforça a hipótese do estudo exploratório ao sugerir que o aumento na taxa de sucesso nas disciplinas de FUP/IPD ao longo dos anos pode estar, em parte, relacionado a popularização dos LLMs como suporte ao aprendizado. Contudo, a dependência excessiva dessas ferramentas por alunos iniciantes levanta a necessidade de estratégias pedagógicas que incentivem um uso mais crítico e consciente, garantindo que os LLMs sejam empregados como um complemento à aprendizagem, e não como um substituto da construção do conhecimento em programação.

6 CONCLUSÕES E TRABALHOS FUTUROS

A análise conjunta do estudo exploratório, das entrevistas com professores e do survey com os alunos revela um panorama no contexto do campus da UFC em Quixadá sobre o impacto dos LLMs no aprendizado de desenvolvimento de software. O estudo exploratório mostrou uma tendência de crescimento na taxa de aprovação em disciplinas introdutórias de programação entre 2015 e 2024, com um aumento mais significativo a partir de 2021, período que coincide com a popularização dos LLMs, como ChatGPT e GitHub Copilot. Embora não seja possível estabelecer uma relação causal direta, os dados sugerem que essas ferramentas podem ter influenciado positivamente o aprendizado, auxiliando os alunos a compreender conceitos e resolver problemas com mais autonomia. No entanto, outros fatores também podem ter contribuído para essa melhora, como mudanças metodológicas e aprimoramentos nas práticas pedagógicas ao longo dos anos.

Os resultados do survey evidenciam que o impacto dos LLMs varia de acordo com o nível de experiência dos alunos. Aqueles com maior familiaridade em programação tendem a utilizar essas ferramentas de forma crítica, validando as respostas e extraíndo benefícios concretos para o aprendizado. Por outro lado, estudantes iniciantes demonstram maior confiança nas respostas fornecidas pelos modelos, muitas vezes aceitando sem questionamento. Isso levanta preocupações sobre o uso passivo dessas tecnologias, uma vez que a dependência excessiva pode comprometer o desenvolvimento da autonomia na resolução de problemas e a consolidação do conhecimento necessário para atividades mais complexas.

As entrevistas com os professores reforçam essa preocupação. Enquanto alguns docentes reconhecem o potencial dos LLMs para facilitar a aprendizagem e esclarecer dúvidas, outros percebem que muitos alunos recorrem a essas ferramentas como um atalho, copiando respostas sem compreender o raciocínio por trás delas. Essa prática pode prejudicar o desempenho em avaliações que exigem conhecimento consolidado e habilidades de programação mais estruturadas. Além disso, os professores relatam que o uso inadequado dos LLMs pode dificultar a diferenciação entre uma resposta precisa e uma incorreta, especialmente para aqueles que ainda estão no início da jornada de aprendizado.

Diante desses achados, torna-se evidente que os LLMs podem desempenhar um papel significativo na educação em programação, mas seu impacto depende diretamente da forma como são utilizados. Essas ferramentas podem contribuir para um aprendizado mais eficiente e aumentar a taxa de sucesso acadêmico, desde que sejam incorporadas de maneira consciente

e estratégica no processo de ensino. A mediação dos professores é essencial para orientar os alunos na validação das respostas geradas e incentivar a desenvolver um pensamento crítico sobre as soluções propostas pelos modelos. Os dados analisados apontam para a necessidade de estratégias pedagógicas que equilibrem o uso dos LLMs com abordagens tradicionais de ensino, promovendo uma melhor armonia entre LLMs e o aprendizado.

6.1 Trabalhos Futuros

Futuras pesquisas devem investigar metodologias que promovam um equilíbrio entre o uso dos LLMs e o desenvolvimento das competências essenciais em programação. Estudos mais aprofundados sobre a relação entre essas ferramentas e o desempenho acadêmico podem fornecer *feedbacks* para a criação de diretrizes educacionais que potencializem os benefícios das LLMs no ensino de desenvolvimento de software, ao mesmo tempo em que minimizem os riscos associados ao uso passivo ou excessivo. Além disso, seria interessante explorar estratégias pedagógicas que incentivem a autonomia dos alunos na verificação das respostas geradas pelos LLMs, promovendo uma aprendizagem mais crítica e eficiente.

Em relação aos trabalhos futuros, é fundamental executar experimentos que possam confirmar as hipóteses levantadas neste estudo, explorando de forma mais aprofundada a relação entre o uso dos LLMs e o desempenho acadêmico. A replicação do estudo em outras universidades poderia ampliar a base de dados, proporcionando uma análise mais robusta e generalizável sobre o impacto dessas ferramentas no aprendizado de programação. Além disso, para futuras pesquisas é importante investigar metodologias que promovam um equilíbrio entre o uso dos LLMs e o desenvolvimento das competências essenciais em programação, considerando tanto os benefícios quanto os riscos associados ao uso dessas tecnologias. Estudos mais aprofundados sobre a relação entre LLMs e desempenho acadêmico podem fornecer *feedbacks* importantes para a criação de diretrizes educacionais, visando maximizar os benefícios das ferramentas de IA enquanto minimizam os riscos de dependência ou uso excessivo.

Outro aspecto relevante para pesquisas futuras é a análise ao longo prazo do impacto dos LLMs no desenvolvimento acadêmico e profissional dos estudantes durante sua trajetória no curso. Compreender como o uso dessas ferramentas influencia o aprendizado de programação em diferentes fases da formação pode contribuir para a formulação de políticas educacionais que garantam um aproveitamento mais eficaz e consciente das LLMs na educação, assegurando que sua aplicação seja um facilitador no processo de aprendizagem, sem comprometer o desenvolvimento

das habilidades essenciais dos alunos.

6.2 Limitações do Estudo

As limitações do estudo estão relacionadas a alguns fatores que impactaram a análise dos dados. Primeiramente, embora tenha sido observada uma correlação entre o aumento nas taxas de aprovação e a popularização dos LLMs a partir de 2021, não foi possível estabelecer uma relação causal direta entre o uso das ferramentas e a melhoria no desempenho acadêmico. Outros fatores, como mudanças metodológicas e aprimoramentos pedagógicos ao longo dos anos, também podem ter contribuído para esse aumento nas taxas de aprovação, dificultando a isolação do impacto específico dos LLMs.

Outro aspecto limitante refere-se à amostra, que foi restrita ao contexto da UFC, Campus Quixadá. Isso significa que os resultados podem não ser generalizáveis para outras instituições ou cursos de graduação em diferentes contextos. A diversidade de experiências e níveis de conhecimento dos alunos pode ter influenciado as respostas, tornando difícil realizar uma análise mais ampla sobre o impacto dos LLMs no aprendizado de programação.

Outra limitação foi a abordagem inicial de coleta de dados via e-mail, que não obteve o retorno esperado. Isso levou à necessidade de uma abordagem mais direta e corpo a corpo para alcançar os alunos, o que pode ter introduzido viés de amostragem, uma vez que nem todos os alunos foram alcançados da mesma forma.

REFERÊNCIAS

- ABAJYAN, A. **Web development**: frontend using css with modern techniques. Tese (Doutorado) – American University of Armenia, 2018.
- AGUILAR, L. J. **Fundamentos de programação**: algoritmos, estruturas de dados e objetos. [S. l.]: AMGH Editora, 2008.
- AN, J.; DING, W.; LIN, C. Chatgpt. **tackle the growing carbon footprint of generative AI**, v. 615, p. 586, 2023.
- BARBOSA, L. M.; PORTES LUIZA ALVES FERREIRA, R. d. J. Inteligência artificial. **Revista tecnologia educacional**, n. 236, p. 16–27, 2023.
- BLACKWELL, A. F. What is programming? In: CITESEER. **PPIG**. [S. l.], 2002. v. 14, p. 204–218.
- BRIGANTI, G. How chatgpt works: a mini review. **European archives of oto-rhino-laryngology**, Springer, v. 281, n. 3, p. 1565–1569, 2024.
- CASELI, H. d. M.; NUNES, M. d. G. V. **Processamento de linguagem natural**: conceitos, técnicas e aplicações em português. 2024.
- CENDÓN, B. V.; RIBEIRO, N. A.; CHAVES, C. J. Pesquisas de survey: análise das reações dos respondentes. **Informação & Sociedade**, v. 24, n. 3, p. 29–48, 2014.
- COPilot, G. **Github copilot**. 2023.
- COZMAN, F. G.; PLONSKI, G. A.; NERI, H. **Inteligência artificial**. 2021.
- DERIBA, F. G.; SANUSI, I. T.; SUNDAY, A. O. Enhancing computer programming education using chatgpt- a mini review. In: **Proceedings of the 23rd koli calling international conference on computing education research**. New York, NY, USA: Association for Computing Machinery, 2024. (Koli Calling '23). ISBN 9798400716539. Disponível em: <https://doi.org/10.1145/3631802.3631848>. Acesso em: 10 fev 2025.
- DOCKER, I. Docker. **linea**, 2020. Acesso em 13 jan 2025. Disponível em: Disponívelem:<https://www.docker.com/what-dock>.
- ENGHOLM, H. J. **Engenharia de software na prática**. [S. l.]: Novatec Editora, 2010.
- ERICKSON, B. J. Basic artificial intelligence techniques: machine learning and deep learning. **Radiologic clinics of north america**, v. 59, n. 6, p. 933–940, 2021.
- FILHO, L. C. P.; SOUZA, T. P. C. de; PAULA, L. B. de. Análise das respostas do chatgpt em relação ao conteúdo de programação para iniciantes. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 34., 2023, Passo Fundo. **Anais** [...]. Porto Alegre: SBC, 2023. p. 1738–1748. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/26794>. Acesso em: 28 fev. 2025.
- GIBIN, G. B.; EUGENIO, I. D.; SILVA, L. A. N. da; SANTOS, C. M. dos. Inteligência artificial frente a resolução de exercícios de química: Um estudo exploratório com o chatgpt. In: **Colloquium Humanarum**. [S. l.: s. n.], 2023. v. 20, n. 1, p. 461–476.

- HONDRO, R. K. Peningkatan pemahaman siswa dalam belajar coding interaktif melalui pengenalan aplikasi codecademy. **Jurnal pengabdian kepada masyarakat**, v. 1, n. 01, p. 18–22, 2024.
- HÖRNEMALM, A. **Chatgpt as a software development tool**: The future of development. 2023.
- INDURKHYA, N.; DAMERAU, F. J. **Handbook of natural language processing**. [S. l.]: Chapman and Hall/CRC, 2010.
- KALLA, D.; SMITH, N.; SAMAAH, F.; KURAKU, S. Study and analysis of chat gpt and its impact on different fields of study. **International journal of innovative science and research technology**, v. 8, n. 3, 2023.
- KITCHENHAM, B. A.; PFLEEGER, S. L. Personal opinion surveys. In: **Guide to advanced empirical software engineering**. [S. l.]: Springer, 2008. p. 63–92.
- KOVANIUK, M. Online compiler «replit» usage during the study of the programming discipline. **Publishing house “Baltija Publishing”**, 2023.
- LEE, K.-F. **Inteligência artificial**. [S. l.]: Globo livros, 2019.
- LIMA, J. Como o chatgpt afeta a educação e o desenvolvimento universitário. **The trends hub**, n. 3, 2023.
- LOPES, A.; GARCIA, G. **Introdução à programação**: 500 algoritmos resolvidos. Rio de Janeiro: Campus, 2002.
- LÜDKE, M.; ANDRÉ, M. E. Abordagens qualitativas de pesquisa: a pesquisa etnográfica e o estudo de caso. **Pesquisa em educação**, EPU, 1986.
- MARTINEZ, A. L. Agressividade tributária: um survey da literatura. **Revista de educação e pesquisa em contabilidade**, Academia Brasileira de Ciências Contábeis, v. 11, p. 106–124, 2017.
- MEDEIROS, R. P.; RAMALHO, G. L.; FALCÃO, T. P. A systematic literature review on teaching and learning introductory programming in higher education. **IEEE transactions on education**, IEEE, v. 62, n. 2, p. 77–90, 2018.
- MENON, A.; TAMUZ, O.; GULWANI, S.; LAMPSON, B.; KALAI, A. A machine learning framework for programming by example. In: PMLR. **International conference on machine learning**. [S. l.], 2013. p. 187–195.
- MORAES, L. d. C.; SILVÉRIO, I. C.; MARQUES, R. A. S.; ANAIA, B. d. C.; PAULA, D. F. de; FARIA, M. C. S. de; CLEVESTON, I.; CORREIA, A. d. S.; FREITAG, R. M. K. Análise de ambiguidade linguística em modelos de linguagem de grande escala (llms). **Arxiv preprint arXiv:2404.16653**, 2024.
- MURPHY-HILL, E.; LEE, D. Y.; MURPHY, G. C.; MCGRENERE, J. How do users discover new tools in software development and beyond? **Computer supported cooperative work (CSCW)**, Springer, v. 24, p. 389–422, 2015.

NIEVES, T.; MENDONÇA, E. A. de; FERREIRA, S. L. Processamento de linguagem natural na indústria aec: uma abordagem para tradução de regulamentos edilícios brasileiros para o domínio bim. In: SIMPOSIO BRASILEIRO DE TECNOLOGIA DE INFORMACAO E COMUNICAO NA CONSTRUCAO, 32. **Anais** [...]. Florianópolis: SBTICC, 2021. p. 178–256. Disponível em: <https://sol.sbc.org.br/index.php/sbie/article/view/26794>. Acesso em: 28 fev. 2025.

PAREDES, C. M. G.; MACHUCA, C.; CLAUDIO, Y. M. S. Chatgpt api: Brief overview and integration in software development. **International journal of engineering insights**, v. 1, n. 1, p. 25–29, 2023.

PENG, S.; KALLIAMVAKOU, E.; CIHON, P.; DEMIRER, M. The impact of ai on developer productivity: Evidence from github copilot. **Arxiv preprint arxiv**, 2023.

PHUNG, T.; PaDUREAN, V.-A.; CAMBRONERO, J.; GULWANI, S.; KOHN, T.; MAJUMDAR, R.; SINGLA, A.; SOARES, G. Generative ai for programming education: Benchmarking chatgpt, gpt-4, and human tutors. In: **Proceedings of the 2023 ACM conference on international computing education research - volume 2**. New York, NY, USA: Association for Computing Machinery, 2023. (ICER '23), p. 41–42. ISBN 9781450399753. Disponível em: <https://doi.org/10.1145/3568812.3603476>. Acesso em: 22 dez 2024.

PRAVEEN, S.; VAJROBOL, V. Understanding the perceptions of healthcare researchers regarding chatgpt: a study based on bidirectional encoder representation from transformers (bert) sentiment analysis and topic modeling. **Annals of biomedical engineering**, Springer, v. 51, n. 8, p. 1654–1656, 2023.

PURYEAR, B.; SPRINT, G. Github copilot in the classroom: learning to code with ai assistance. **Journal of computing sciences in colleges**, Consortium for Computing Sciences in Colleges, v. 38, n. 1, p. 37–47, 2022.

RAHMANIAR, W. Chatgpt for software development: Opportunities and challenges. **It professional**, IEEE, v. 26, n. 3, p. 80–86, 2024.

RAM, B.; VERMA, P. Artificial intelligence ai-based chatbot study of chatgpt, google ai bard and baidu ai. **World journal of advanced engineering technology and sciences**, v. 8, n. 01, p. 258–261, 2023.

RODRIGUES, O. S.; RODRIGUES, K. S. A inteligência artificial na educação: os desafios do chatgpt. **Texto Livre**, SciELO Brasil, v. 16, p. e45997, 2023.

RUSSELL, S. J.; NORVIG, P. **Artificial intelligence**: a modern approach. [S. l.]: pearson, 2016.

SARRION, E. What is chatgpt? In: **Exploring the power of ChatGPT, applications, techniques, and implications**. [S. l.]: Springer, 2023. p. 3–8.

SCHEUREN, F. What is a survey: **American Statistical Association Alexandria**. In: . [S. l.: s. n.], 2004. p. 345.

SHARMA, D.; VIDHATE, D. A.; OSEI-ASIAMAH, J.; KUMARI, R. M.; MAHAJAN, V.; RAJAGOPAL, K. Exploring the evolution of chatgpt: From origin to revolutionary influence. **Educational administration**, v. 30, n. 5, 2024.

- SILVA, B. A. C. **Uso de grandes modelos de linguagem em aplicações:** aplicação web para auxiliar profissionais de qualidade de software. 2023. 67 f. }. In: . Ouro Preto: [S. n.], 2023.
- SILVA, T. R. D.; MEDEIROS, T.; MEDEIROS, H.; LOPES, R.; ARANHA, E. Ensino-aprendizagem de programação: uma revisão sistemática da literatura. **Revista brasileira de informática na educação**, v. 23, n. 01, p. 182, 2015.
- SOARES, R. P. Queryaugment-rag: Aprimorando a recuperação e geração de informações em modelos de linguagem de grande escala. **Revista contemporânea**, v. 4, n. 6, p. e4470–e4470, 2024.
- SUN, X.; DONG, L.; LI, X.; WAN, Z.; WANG, S.; ZHANG, T.; LI, J.; CHENG, F.; LYU, L.; WU, F. *et al.* Pushing the limits of chatgpt on nlp tasks. **Arxiv preprint arxiv**, 2023.
- TAULLI, T. **Introdução à inteligência artificial como uma abordagem não técnica.** [S. l.]: Novatec Editora, 2020.
- TAVARES, L. A.; MEIRA, M. C.; AMARAL, S. F. do. Inteligência artificial na educação: Survey. **Brazilian journal of development**, v. 6, n. 7, p. 48699–48714, 2020.
- TEEL, Z. A.; WANG, T.; LUND, B. Chatgpt conundrums. **Teel, ZA, Wang, T., & Lund, B.(2023). ChatGPT conundrums of probing plagiarism and parroting problems in higher education practices. College e research libraries news**, v. 84, n. 6, p. 205–208, 2023.
- VIEIRA, J. K. M.; FARIA, I. H. D.; MOURA, H. P. D. Evaluating a conceptual model for projects observatories through a survey. In: **Proceedings of the XIX brazilian symposium on information systems**. New York, NY, USA: Association for Computing Machinery, 2023. (SBSI '23), p. 380–387. ISBN 9798400707599. Disponível em: <https://doi.org/10.1145/3592813.3592928>. Acesso em: 23 jul 2024.
- VOGEL, D.; RAMOS, A. M.; FRANZONI, A. M. B. Transformando a educação com large language models (llms): benefícios, limitações e perspectivas. **Caderno Pedagógico**, v. 22, n. 4, p. e13846–e13846, 2025.
- WANG, P. On defining artificial intelligence. **Journal of artificial general intelligence**, De Gruyter Poland, v. 10, n. 2, p. 1–37, 2019.
- WELSBY, P.; CHEUNG, B. M. **ChatGPT.** [S. l.]: Oxford University Press, 2023. 1047–1048 p.
- YAO, Y.; DUAN, J.; XU, K.; CAI, Y.; SUN, Z.; ZHANG, Y. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. **High confidence computing**, Elsevier, p. 100211, 2024.
- YETISTIREN, B.; OZSOY, I.; TUZUN, E. Assessing the quality of github copilot's code generation. In: **Proceedings of the 18th international conference on predictive models and data analytics in software engineering.** [S. l.: s. n.], 2022. p. 62–71.
- YILMAZ, R.; YILMAZ, F. G. K. Augmented intelligence in programming learning: Examining student views on the use of chatgpt for programming learning. **Computers in human behavior**, Elsevier, v. 1, n. 2, p. 100005, 2023.
- ZHOU, Z.-H. **Machine learning.** [S. l.]: Springer nature, 2021.

APÊNDICE A – PERGUNTAS DO SURVEY SOBRE O USO DO CHATGPT COMO RECURSO PARA O APRENDIZADO DE DESENVOLVIMENTO DE SOFTWARE

Apresentação do Formulário e Termo de Livre Consentimento

Obrigado por dedicar seu tempo para participar deste estudo! Este survey é parte de uma pesquisa acadêmica que investiga o uso de Modelos de Linguagem de Grande Escala (LLMs) como ferramenta de apoio ao aprendizado no desenvolvimento de software. O objetivo principal é compreender como essas tecnologias influenciam o processo de aprendizagem e o desempenho acadêmico nas disciplinas de Fundamentos de Programação (FUP) e Introdução à Programação para Design (IPD). Sua participação é voluntária, e todas as informações fornecidas serão tratadas de forma confidencial. Este estudo é voltado para estudantes que cursaram essas disciplinas no semestre 2024.1. Ao continuar o preenchimento deste formulário, você concorda com os seguintes termos:

- Participação Voluntária: Sua participação é totalmente opcional. Você pode interrompê-la a qualquer momento, sem necessidade de justificativa ou prejuízo.
- Confidencialidade: Todas as informações coletadas serão mantidas em sigilo. Nenhum dado pessoal que permita sua identificação será compartilhado sem o seu consentimento explícito.
- Objetivo da Pesquisa: As respostas serão utilizadas exclusivamente para fins acadêmicos, com o propósito de analisar o impacto de ferramentas de IA no aprendizado de programação.
- Consentimento Informado: Ao prosseguir, você declara que leu, compreendeu e concorda com as informações apresentadas neste termo, consentindo livremente com sua participação.

Siglas Utilizadas:

- FUP: Fundamentos de Programação
- IPD: Introdução à Programação para Design
- LLM: Grande Modelo de Linguagem (ChatGPT, GPT-4, Copilot)

A.1 Informações Demográficas e Acadêmicas

1. Qual seu nome completo?

- (a) Resposta aberta

2. Quantos anos você tem?

- (a) Resposta aberta

3. Qual seu curso de graduação?

- (a) Engenharia de Software
- (b) Engenharia da Computação
- (c) Ciência da Computação
- (d) Design Digital
- (e) Redes de Computadores
- (f) Sistemas de Informação

4. Quem foi seu professor de FUP/IPD em 2024.1?

- (a) Fábio Dias
- (b) Bruno Góes
- (c) Criston Pereira
- (d) Enyo José
- (e) Paulo Henrique
- (f) Outro:

5. Marque a opção que melhor se aplica ao seu rendimento final na disciplina de FUP/IPD em 2024.1.

- (a) Aprovado por nota
- (b) Aprovado por conceito B (avaliação final)
- (c) Reprovado por nota
- (d) Reprovado por falta
- (e) Reprovado por nota e por falta
- (f) Trancou a disciplina

6. Se você já teve algum contato com programação antes de cursar as disciplinas de FUP/IPD, como você classificaria o seu nível de experiência?

- (a) Nenhuma experiência
- (b) Iniciante
- (c) Intermediário
- (d) Avançado

7. Caso você tenha tido algum conhecimento prévio, como você adquiriu esse conhecimento?

- (a) Recursos da internet para apoio
 - (b) Livros específicos
 - (c) Uso de ferramentas de IA como LLMs para aprender programação de forma autônoma
 - (d) Sou egresso de curso técnico de TI
 - (e) Outros...
- 8. Você utilizou alguma ferramenta de LLM (ex.: ChatGPT, GPT-4, Copilot, outros) em suas atividades da disciplina de FUP/IPD em 2024.1?**
- (a) Sim
 - (b) Não

A.2 Uso e Aplicações dos LLMs

As perguntas desta seção são opcionais, considerando que nem todos os estudantes podem ter utilizado esse tipo de recurso nas disciplinas de FUP ou IPD durante o semestre 2024.1.

- 1. Você utilizou alguma ferramenta de LLMs como apoio às disciplinas de FUP/IPD em 2024.1? Indique qual(is).**
- (a) ChatGPT
 - (b) Copilot
 - (c) Gemini
 - (d) Outros...
- 2. Dentro do contexto de FUP/IPD, quais atividades de programação você utilizou os LLMs em 2024.1? (Selecione todas que se aplicam.)**
- (a) Resolução de problemas de codificação
 - (b) Depuração de código (debugging)
 - (c) Obtenção de exemplos ou snippets de código
 - (d) Desenvolvimento de projetos ou sistemas completos
 - (e) Aprendizado de conceitos de programação
 - (f) Outros...
- 3. Em sua experiência, o uso de LLM ajudou a superar desafios que você encontrava durante o aprendizado na disciplina de FUP/IPD?**
- (a) Sim, ajudou significativamente

- (b) Sim, ajudou em alguns aspectos
- (c) Não ajudou, mas também não prejudicou
- (d) Não ajudou e dificultou em alguns momentos
- (e) Não utilizei LLMs durante a disciplina

4. Como você avalia a qualidade das soluções fornecidas pelos LLMs para problemas de programação em FUP/IPD?

- (a) Muito úteis e precisas
- (b) Úteis, mas com necessidade de ajustes
- (c) Às vezes úteis, mas frequentemente imprecisas
- (d) Pouco úteis ou imprecisas

5. Em sua opinião, qual o impacto dos LLMs no seu desempenho na disciplina de FUP/IPD em 2024.1?

- (a) Impacto muito positivo: melhorou significativamente meu desempenho
- (b) Impacto positivo: ajudou em alguns aspectos do meu desempenho
- (c) Impacto neutro: não percebi diferença no meu desempenho
- (d) Impacto negativo: dificultou meu aprendizado em alguns momentos
- (e) Não utilizei LLMs durante a disciplina

APÊNDICE B – ROTEIRO PARA ENTREVISTA COM DOCENTES QUE MINISTRARAM FUP/IPD EM 2024.1 SOBRE O USO DE LLMS NO APRENDIZADO DO DESENVOLVIMENTO DE SOFTWARE

Apresentação do Formulário e Termo de Livre Consentimento

Obrigado por participar deste estudo! Esta entrevista faz parte de um estudo acadêmico sobre o uso de LLMs como recurso para o aprendizado de desenvolvimento de software. O objetivo é entender como essas ferramentas influenciam a aprendizagem e o desempenho dos alunos em disciplinas de Fundamento de Programação e Introdução à Programação para Design. Sua participação é voluntária e confidencial.

Termo de Consentimento

- Participação Voluntária: Sua participação neste estudo é totalmente voluntária. Você pode desistir a qualquer momento, sem qualquer prejuízo ou necessidade de justificativa.
- Confidencialidade: As informações coletadas serão tratadas de forma confidencial. Nenhuma informação pessoal identificável será divulgada sem sua permissão explícita.
- Objetivo do Estudo: As respostas fornecidas serão usadas exclusivamente para fins de pesquisa acadêmica, visando analisar o uso de ferramentas de IA no aprendizado de programação.

B.1 Perfil do Docente

Questão: Qual o seu nome completo?

- Resposta aberta

Questão: Quantas vezes você já ministrou a disciplina de FUP/IPD?

- Resposta aberta

B.2 Percepção sobre uso de LLMs

Questão: Qual é o seu conhecimento geral sobre LLMs?

- Resposta aberta

Questão: Você já utilizou algum recurso de LLMs para o ensino de programação?

- Resposta aberta

Questão: Em sua opinião, como os LLMs influenciam a aprendizagem de programação dos alunos?

- Resposta aberta

B.3 Políticas e práticas em sala de aula

Questão: Você acredita que os LLMs têm potencial para atuar como ferramentas complementares no ensino de programação? Por quê?

- Resposta aberta

Questão: Durante o semestre de 2024.1, havia alguma política oficial, institucional ou pessoal (sua) em relação ao uso de LLMs na disciplina de FUP? Caso sim, por favor, descreva como essa política era aplicada e o que motivou sua adoção!

- Resposta aberta

Questão: Durante o semestre de 2024.1, havia alguma política oficial, institucional ou pessoal (sua) em relação ao uso de LLMs na disciplina de FUP? Caso não, se você identificou alunos utilizando LLMs durante as atividades acadêmicas, como isso foi percebido ou tratado em sala de aula?

- Resposta aberta

Questão: Que tipos de tarefas ou atividades, em sua opinião, os alunos têm mais dificuldade e acabam consultando ferramentas de LLMs para ter possíveis respostas rápidas (ex.: resolução de problemas, desenvolvimento de algoritmos, projetos práticos)?

- Resposta aberta

B.4 Impacto nos Resultados Acadêmicos

Questão: Durante o semestre de 2024.1, em que você lecionou, houve alguma alteração nas taxas de aprovação e reprovação em FUP/IPD que você atribua ao uso de LLMs?

- Resposta aberta

Questão: Você percebeu mudanças no comportamento dos alunos relacionadas à prática de programação ou à dependência de ferramentas como LLMs? Por favor, detalhar exemplos específicos, se aplicável.

- Resposta aberta

B.5 Experiências e Opiniões Pessoais

Questão: Em sua experiência, quais são as principais vantagens dos LLMs no contexto do ensino de programação?

- Resposta aberta

Questão: Quais são as maiores limitações ou riscos que você identifica no uso de LLMs por alunos de programação?

- Resposta aberta

B.6 Sugestões e Recomendação de Uso

Questão: Com base na sua experiência, você recomendaria a utilização de LLMs para alunos de FUP/IPD? Se sim, em que condições ou contextos específicos? Se não, por quê?

- Resposta aberta

Questão: Há algo mais que gostaria de compartilhar sobre sua experiência ou sobre como você percebe a influência das LLMs no ensino?

- Resposta aberta