INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS PICOS
CURSO EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

AVELAR RODRIGUES DE SOUSA

SisTIRA: Plataforma Web De Tutoria Inteligente Para Avaliação Automática De Respostas Discursivas

PICOS
2025

AVELAR RODRIGUES DE SOUSA

SisTIRA: Plataforma Web De Tutoria Inteligente Para Avaliação Automática De Respostas Discursivas

Trabalho de Conclusão de Curso (artigo científico) apresentado como exigência parcial para obtenção do diploma do Curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Picos.

Orientador: Prof. Dr. Rogério Figueredo de Sousa.
Coorientador: Prof. Dr. Rafael Torres Anchiêta.

PICOS
2025

LISTA DE ABREVIATURAS E SIGLAS

AES	*Automated Essay Scoring*
AM	Aprendizado de Máquina
API	*Application Programming Interface*
ASAG	*Automatic Short Answer Grading*
AVA	Ambiente Virtual de Aprendizagem
BERT	*Bidirectional Encoder Representations from Transformers*
CSS	*Cascading Style Sheets*
DSR	*Design Science Research*
DTO	*Data Transfer Object*
EaD	Educação à Distância
EI	Extração de Informação
GPT	*Generative Pre-trained Transformer*
HQA	*Health Question Answering*
HTTP	*Hypertext Transfer Protocol*
IA	Inteligência Artificial
IE	Instituições de Ensino
JSON	*JavaScript Object Notation*
JWT	*JSON Web Token*
LLM	*Large Language Model*
LMS	*Learning Management System*
OAS	Especificação *OpenAPI*
ORM	*Object-Relational Mapping*
PLN	Processamento de Linguagem Natural
QR Code	*Quick Response Code*
SBERT	*Sentence-BERT*
SisTIRA	Sistema de Tutoria Inteligente de Respostas Automáticas
STI	Sistema Tutor Inteligente
WMD	*Word Mover's Distance*

SisTIRA: Plataforma Web De Tutoria Inteligente Para Avaliação Automática De Respostas Discursivas

Avelar Rodrigues de Sousa[1]
Rogério Figueredo de Sousa[2]
Rafael Torres Anchiêta[3]

RESUMO

Este trabalho apresenta o Sistema de Tutoria Inteligente de Respostas Automáticas (SisTIRA), uma plataforma baseada em técnicas de Processamento de Linguagem Natural (PLN) e Aprendizado de Máquina (AM) para avaliação automatizada de respostas discursivas em português. O SisTIRA integra um modelo de linguagem como avaliador automático no paradigma *LLM-as-judge*, proporcionando uma correção precisa e escalável. A pesquisa explora como a tecnologia de PLN, aliada a modelos de linguagem avançados como o *Gemini-1.5-flash-8b*, pode ser utilizada para avaliar respostas de forma mais dinâmica e justa, ampliando as possibilidades de personalização e *feedback*. O estudo inclui uma análise preliminar com 20 rodadas de avaliação, evidenciando a consistência do sistema e a viabilidade da automatização da correção discursiva em ambientes educacionais, como Ambientes Virtuais de Aprendizagem (AVAs).

Palavras-chave: Processamento de Linguagem Natural, Avaliação Automatizada, Respostas Discursivas, Inteligência Artificial, Educação a Distância, Sistema de Tutoria Inteligente.

ABSTRACT

This paper introduces the Intelligent Automatic Response Tutoring System (SisTIRA), a platform based on Natural Language Processing (NLP) and Machine Learning (ML) techniques for automated assessment of short answer responses in Portuguese. SisTIRA integrates a language model as an automatic evaluator in the LLM-as-judge paradigm, providing accurate and scalable grading. The research explores how NLP technology, combined with advanced language models like Gemini-1.5-flash-8b, can be used to assess responses in a more dynamic and fair manner, enhancing the possibilities for personalization and feedback. The study includes a preliminary analysis with 20 rounds of evaluation, demonstrating the system's consistency and the feasibility of automating the grading of discursive answers in educational environments like Virtual Learning Environments (VLEs).

Keywords: *Natural Language Processing, Automated Grading, Short Answer Responses, Artificial Intelligence, Distance Education, Intelligent Tutoring System.*

Data de aprovação: 05/09/2025 (Cinco de Setembro de Dois Mil e Vinte e Cinco)

[1] Graduando em Análise e Desenvolvimento de Sistemas. IFPI. avelarrodrigues89@gmail.com.
[2] Professor Doutor do IFPI Campus Picos. rogerio.sousa@ifpi.edu.br.
[3] Professor Doutor do IFMA Campus Caxias. rafael.torres@ifma.edu.br.

1 INTRODUÇÃO

A pandemia da Covid-19 desencadeou obstáculos sem precedentes ao setor educacional, como a interrupção das atividades presenciais nas Instituições de Ensino (IE) e a dificuldade de adaptação dos docentes e discentes à educação remota, o que demandou a rápida implementação de plataformas digitais e ferramentas interativas para garantir a continuidade das aulas e minimizar as perdas de aprendizagem (World Bank, UNESCO e UNICEF, 2021). Entre essas soluções tecnológicas, os Ambientes Virtuais de Aprendizagem (AVAs) destacam-se como sistemas digitais criados para oferecer suporte ao ensino em diferentes modalidades, promovendo interações educacionais, compartilhamento de materiais e a realização de avaliações (Gomes e Pimentel, 2021).

Diante desse cenário, a adoção dos AVAs ampliou as possibilidades de ensino, tornando-se um recurso indispensável para viabilizar a continuidade das atividades pedagógicas em um panorama de incertezas (Gomes e Pimentel, 2021). Segundo o Censo da Educação Superior 2023, o ensino remoto cresceu significativamente, com as matrículas na modalidade a distância alcançando 49,2% do total de estudantes em graduação (Inep, 2024). Entretanto, essa transição repentina também expôs desafios estruturais, como a dificuldade de acesso à internet para os estudantes e a falta de infraestrutura adequada nas IE (Costa, Lima e Passos, 2024). Além disso, os desafios da avaliação da aprendizagem tornaram-se ainda mais evidentes, exigindo dos docentes a adaptação a novas práticas metodológicas e ao uso de tecnologias digitais, muitas vezes sem a formação adequada para tal (Bortolini e Nauroski, 2022). Ainda assim, essas plataformas demonstraram potencial ao oferecer funcionalidades que vão além da simples substituição das aulas presenciais, expandindo as possibilidades pedagógicas e oferecendo ferramentas para avaliações interativas e personalização do ensino (Gomes e Pimentel, 2021).

Apesar dessas vantagens, um dos maiores desafios enfrentados pelos docentes nesse contexto foi a avaliação automática de questões discursivas, que, embora já fosse uma dificuldade pré-existente, se tornou ainda mais evidenciada devido à transição para o ensino remoto (Almeida e Moura, 2024). A necessidade de adaptar práticas de avaliação em um cenário de ensino digital, sem a presença física dos alunos, e com a ampliação da quantidade de avaliações a serem corrigidas,

exigiu dos docentes uma maior capacidade de gerenciar e analisar respostas discursivas de maneira eficiente (Costa, Lima e Passos, 2024). Esse desafio tornou-se ainda mais relevante, pois a avaliação dessas questões demanda um nível avançado de processamento textual (Almeida e Moura, 2024; Bortolini e Nauroski, 2022). Diferentemente das questões objetivas, que podem ser corrigidas de forma rápida e precisa, as respostas discursivas exigem uma análise semântica detalhada, tornando o processo de automatização mais complexo (Oliveira, Pozzebon e Santos, 2020). A pandemia evidenciou essa limitação, impulsionando o desenvolvimento e a adoção de tecnologias voltadas à melhoria das avaliações em AVAs (Bortolini e Nauroski, 2022). Pesquisas recentes demonstram que o Processamento de Linguagem Natural (PLN) tem sido empregado para corrigir questões discursivas, aplicando técnicas de similaridade textual e aprendizado profundo para refinar a acurácia do processo de avaliação (Almeida e Moura, 2024).

Com o avanço dessas técnicas, a automação da correção de respostas discursivas tem proporcionado ganhos expressivos em precisão semântica, contribuindo para avaliações mais justas e reduzindo a necessidade de intervenção humana (Almeida e Moura, 2024). O uso de algoritmos avançados, aliado à aplicação de técnicas de *embeddings,* que são representações vetoriais contextualizadas das palavras e sentenças, tem se mostrado eficaz na identificação de similaridades textuais entre respostas. Técnicas como o *Word Mover's Distance* (WMD) permitem mensurar com precisão a proximidade semântica entre conteúdos discursivos, contribuindo para agilizar e tornar mais confiável o processo de avaliação automática (Kusner et al., 2015). Estudos apontam que a integração dessas tecnologias em AVAs pode otimizar a retroalimentação dos estudantes, promovendo um ensino mais dinâmico e adaptável ao ritmo individual de aprendizado (Oliveira, Pozzebon e Santos, 2020). Além disso, a Inteligência Artificial (IA) aplicada à educação não apenas melhora a eficiência da avaliação, mas também viabiliza o monitoramento contínuo do desempenho discente, fornecendo subsídios para a adaptação de metodologias pedagógicas mais eficazes (Silva, 2023).

Neste cenário de digitalização e transformação educacional, a automação da avaliação tem desempenhado um papel fundamental na modernização dos processos educacionais, especialmente na Educação a Distância (EaD) (Bortolini e Nauroski, 2022). Tecnologias baseadas em PLN estão revolucionando a avaliação

de respostas discursivas, melhorando a precisão da correção e promovendo um aprendizado mais dinâmico e envolvente (Silva, 2023). A implementação dessas tecnologias nos AVAs aprimora o ensino, viabilizando o *feedback* e o acompanhamento contínuo do aluno, o que torna a avaliação mais acessível e eficiente (Almeida e Moura, 2024). Dessa forma, a incorporação da IA na educação representa um avanço essencial na personalização das estratégias pedagógicas e na modernização da avaliação acadêmica, fortalecendo a qualidade da aprendizagem (Oliveira, Pozzebon e Santos, 2020). Neste contexto, este trabalho tem como objetivo apresentar o Sistema de Tutoria Inteligente de Respostas Automáticas (SisTIRA)[4], uma plataforma que integra técnicas de PLN e Aprendizado de Máquina (AM) para a avaliação automática de respostas discursivas em português.

O restante deste artigo está organizado da seguinte forma: a Seção 2 introduz o referencial teórico do projeto, abordando os conceitos de Sistemas Tutores Inteligentes (STIs), os desafios da avaliação em AVAs e a aplicação de PLN para correção automática de respostas discursivas; a Seção 3 descreve a metodologia adotada para o desenvolvimento e validação do SisTIRA, abordando os processos de implementação da plataforma, análise de consistência e reprodutibilidade do sistema de correção automática, e os testes realizados para avaliar a estabilidade das avaliações; a Seção 4 discute os trabalhos relacionados, destacando soluções existentes e como o SisTIRA se diferencia, especialmente no uso de IA e PLN para automatizar a correção de respostas discursivas; a Seção 5 detalha a infraestrutura do sistema, descrevendo a arquitetura, os componentes tecnológicos utilizados e a integração com modelos de IA; a Seção 6 fornece uma análise preliminar da eficácia da IA integrada, destacando a contribuição do modelo *Gemini-1.5-flash-8b* na correção automática das respostas; por fim, a Seção 7 apresenta as conclusões do estudo, discutindo os resultados obtidos e apontando as possibilidades de melhorias futuras no sistema.

2 REFERENCIAL TEÓRICO

Com o objetivo de fundamentar teoricamente a presente pesquisa, esta seção apresenta os principais conceitos, métodos e abordagens relacionados ao tema

[4] Vídeo apresentando a ferramenta: <https://drive.google.com/drive/folders/10CwhtxfEr8gN1MaTCba4sC9oOeBeGu0x?usp=sharing>

investigado. A construção deste referencial visa contextualizar e fundamentar teoricamente a pesquisa desenvolvida, estabelecendo os aportes conceituais necessários à compreensão da proposta e de suas implicações metodológicas. Além disso, busca-se destacar as contribuições das tecnologias de PLN e dos STIs no aperfeiçoamento de práticas pedagógicas mediadas por computador. As subseções a seguir abordam, de forma estruturada, os fundamentos sobre STIs, os desafios da avaliação em AVAs, os princípios do PLN aplicado à educação e os modelos computacionais de linguagem.

2.1 SISTEMAS TUTORES INTELIGENTES

Os STIs são sistemas computacionais que integram conhecimentos pedagógicos, cognitivos e técnicos com o propósito de oferecer instrução personalizada por meio de mecanismos de diagnóstico, adaptação e *feedback* contínuo ao aprendiz (Nkambou, Bourdeau e Psyché, 2010). Em contextos educacionais digitais, esses sistemas têm ganhado destaque por ampliar o acompanhamento individualizado, sobretudo em cenários de ensino remoto e híbrido, nos quais o suporte personalizado se torna mais desafiador (Nkambou, Bourdeau e Psyché, 2010).

Do ponto de vista arquitetural, os STIs se organizam em três componentes principais. O modelo do domínio representa o conhecimento a ser ensinado e estrutura conteúdos e habilidades; o modelo do estudante armazena informações sobre desempenho, erros, estilos cognitivos e progresso; e o modelo pedagógico decide como ensinar, definindo estratégias didáticas, fornecendo explicações, selecionando exemplos e adaptando a dificuldade ao perfil do aluno (Ritter, Blessing e Wheeler, 2003). Essa separação em módulos reutilizáveis favorece a personalização efetiva da instrução, alinhando conteúdo e estratégia às necessidades específicas de cada aprendiz (Murray, 2003).

2.1.1 Ambientes Virtuais de Aprendizagem

No escopo de um STI, o AVA atua como componente central da infraestrutura pedagógica digital, centralizando usuários, conteúdos e fluxos avaliativos sobre os quais o tutor inteligente aplica personalização e *feedback*. Segundo Gomes e Pimentel (2021), AVAs são plataformas digitais projetadas para dar suporte aos

processos pedagógicos, possibilitando comunicação, acesso a conteúdos e execução de atividades didáticas. Além de viabilizar a continuidade do ensino a distância, os AVAs favorecem a personalização de práticas pedagógicas, permitindo interações mediadas por tecnologia que respeitam as individualidades dos estudantes (Gomes e Pimentel, 2021).

Ainda que sejam amplamente utilizados, os AVAs enfrentam restrições importantes relacionadas à formação docente para o uso de tecnologias digitais. Dados recentes indicam que 54% das escolas ofertaram formação (presencial ou a distância) aos professores sobre tecnologias digitais nos 12 meses anteriores, enquanto em áreas rurais essa oferta foi de 45%; a participação em programas de formação alcançou 57% (rural: 48%) em 2023[5, 6]. Além disso, em 2022, 56% dos professores declararam ter participado de formação continuada sobre o uso de tecnologias digitais nos 12 meses anteriores (Cetic.br, TIC Educação 2022, indicador H3[7]).

Como mecanismos de suporte adaptativo baseados em IA, os STIs articulam-se naturalmente com AVAs. Enquanto o AVA organiza o curso, orquestra atividades, gerencia matrículas e oferece canais de interação, o STI adiciona diagnóstico em tempo real, adaptação de trilhas e *feedback* formativo sobre tarefas, inclusive discursivas, consumindo e produzindo dados que o próprio AVA registra (por exemplo, notas, tentativas e engajamento) (Gomes e Pimentel, 2021; Nkambou, Bourdeau e Psyché, 2010). Em termos práticos, essa integração pode ocorrer como módulo nativo do AVA ou como serviço externo conectado à plataforma, preservando a experiência do usuário no mesmo ecossistema e potencializando recursos instrucionais e avaliativos disponíveis.

Diante desse contexto, torna-se relevante compreender o papel dos AVAs na mediação pedagógica e na avaliação da aprendizagem, especialmente diante das possibilidades abertas pelo uso de tecnologias inteligentes. O SisTIRA surge com a proposta de integrar técnicas de PLN à avaliação discursiva nesses ambientes, oferecendo uma solução escalável e precisa para um dos principais desafios da educação digital contemporânea.

---

[5] Cetic.br, TIC Educação 2022, indicador J1: https://cetic.br/pt/tics/educacao/2023/escolas/J1/
[6] Cetic.br, TIC Educação 2022, indicador J3: https://cetic.br/pt/tics/educacao/2023/escolas/J3/
[7] Cetic.br, TIC Educação 2022, indicador H3: https://www.cetic.br/pt/tics/educacao/2022/professores/H3/

2.1.1.1 Avaliação da Aprendizagem em Ambientes Virtuais de Aprendizagem

A avaliação da aprendizagem desempenha papel central no processo educativo, atuando como instrumento de acompanhamento, regulação e orientação das práticas pedagógicas, ao permitir que o docente identifique progressos e necessidades dos estudantes (Perrenoud, 1999; Luckesi, 2011). No contexto dos AVAs, esse papel da avaliação se redefine, exigindo metodologias adaptadas e recursos tecnológicos capazes de atender às especificidades do meio digital (Bortolini e Nauroski, 2022).

Conforme Bortolini e Nauroski (2022), o ensino remoto imposto pela pandemia forçou uma revisão das práticas avaliativas tradicionais, impulsionando a adoção de estratégias mais dinâmicas do que os questionários objetivos. Entretanto, mesmo com essa transformação, os AVAs ainda carecem de soluções eficazes para avaliação de questões discursivas, que envolvem raciocínio, interpretação e argumentação, elementos que desafiam abordagens baseadas apenas em palavras-chave (Oliveira, Pozzebon e Santos, 2020; Silva, 2023).

Além da complexidade semântica envolvida, a avaliação discursiva manual demanda alto investimento de tempo por parte dos docentes, o que se torna inviável em turmas numerosas (Silva, 2023). O julgamento humano também está sujeito a variabilidade, cansaço e viés, o que compromete a equidade e a consistência do processo (Oliveira, Pozzebon e Santos, 2020).

Sob essa ótica, surgem propostas de aplicação de técnicas de PLN como alternativa viável para automação da avaliação discursiva. Silva (2023) demonstrou que abordagens baseadas em Extração de Informação (EI) são eficazes na análise de respostas curtas, obtendo níveis satisfatórios de concordância com corretores humanos. Essas abordagens se baseiam na detecção da presença de conceitos-chave previamente definidos.

Outro exemplo relevante é o trabalho de Oliveira, Pozzebon e Santos (2020), que propuseram um sistema automático de avaliação utilizando medidas de similaridade semântica, como a *Cosine Similarity* e o WMD (Kusner et al., 2015), em conjunto com *embeddings* para representar respostas em níveis vetoriais. Essa proposta contribui com a discussão sobre o uso de técnicas de similaridade semântica para avaliação automatizada de textos, especialmente no contexto educacional digital.

Os estudos analisados sugerem que o uso de técnicas de PLN pode promover transformações significativas nos processos avaliativos em AVAs, sobretudo ao lidar com a complexidade da linguagem natural. A adoção de tecnologias baseadas em PLN permite ampliar a escala da avaliação discursiva sem comprometer sua qualidade, promovendo maior equidade, agilidade e personalização no acompanhamento do desempenho dos estudantes. Com isso, os AVAs tendem a se consolidar não apenas como plataformas de ensino, mas como espaços avaliativos mais sensíveis às demandas cognitivas e linguísticas dos alunos.

2.2 PROCESSAMENTO DE LINGUAGEM NATURAL NA EDUCAÇÃO

O ponto de partida para compreender o PLN é entender o que é a linguagem natural em si e sua relação com o seu processamento. Segundo Silva (2023, p. 21),

> “A linguagem natural é qualquer linguagem desenvolvida naturalmente pelo ser humano, de forma não premeditada e sem planejamento consciente, com objetivo de permitir a comunicação entre indivíduos. [...] o processamento de linguagem natural permite que os computadores entendam e extraiam informações expressas em linguagem natural como humanos.”

É com base na forma de comunicação espontânea supracitada que o PLN se consolida como um ramo da Ciência da Computação e da IA, voltado ao desenvolvimento de métodos que permitam aos computadores compreender, interpretar e gerar linguagem humana (Almeida e Moura, 2024; Silva, 2023). Segundo Dande e Pund (2023), o PLN busca capacitar as máquinas a realizar tarefas que normalmente exigiriam inteligência humana, possibilitando a interação entre humanos e computadores por meio da linguagem natural. Entre os objetivos centrais do PLN está a criação de sistemas capazes de compreender e produzir linguagem de forma eficaz, com aplicações em áreas como análise de sentimentos, tradução automática, reconhecimento de voz e avaliação textual (Jurafsky e Martin, 2025; Dande e Pund, 2023; Oliveira, Pozzebon e Santos, 2020).

A aplicação do PLN no campo educacional tem se intensificado, sobretudo em AVAs, onde há uma demanda crescente por ferramentas que automatizem a análise e interpretação de produções textuais (Almeida e Moura, 2024; Silva, 2023). Para isso, são empregadas tarefas como tokenização, análise morfossintática, extração

de entidades nomeadas, reconhecimento de relações e análise semântica (Silva, 2023). Essas técnicas viabilizam a conversão de textos livres em estruturas formais e computacionalmente tratáveis, permitindo que sistemas educacionais interpretem o conteúdo produzido por estudantes de forma mais precisa e sistematizada (Silva, 2023).

O surgimento de modelos baseados em redes neurais profundas, como o *Bidirectional Encoder Representations from Transformers* (BERT) e o *Generative Pre-trained Transformer* (GPT), revolucionou o PLN ao permitir representações contextuais mais sofisticadas da linguagem (Dande e Pund, 2023). Esses modelos são capazes de compreender o significado das palavras considerando seu contexto, o que os torna especialmente eficazes em tarefas de interpretação semântica, como a comparação entre textos discursivos (Dande e Pund, 2023).

Nesse contexto, técnicas como a WMD têm demonstrado alta precisão ao medir a similaridade entre documentos, mesmo quando não compartilham termos em comum, ao basear-se em *embeddings* como os gerados pelo word2vec (Kusner et al., 2015). Essa abordagem se destaca por ser interpretável, sem necessidade de hiperparâmetros e capaz de capturar nuances semânticas entre palavras por meio de um modelo de transporte ótimo. No âmbito da avaliação da aprendizagem, esses avanços têm possibilitado o desenvolvimento de sistemas capazes de analisar respostas em linguagem natural, identificar *nuances* de significado e promover uma avaliação mais justa, rápida e escalável (Almeida e Moura, 2024; Silva, 2023).

Considerando os desafios da educação contemporânea, sobretudo no ensino remoto e híbrido, a integração de PLN aos AVAs configura-se como uma resposta tecnológica viável para a modernização dos instrumentos de avaliação e acompanhamento da aprendizagem (Almeida e Moura, 2024).

2.2.1 Modelos de Processamento de Linguagem Natural

Os modelos de PLN são arquiteturas computacionais projetadas para interpretar, classificar e gerar linguagem humana, baseando-se em métodos estatísticos e, mais recentemente, em redes neurais profundas treinadas com grandes volumes de dados textuais (Jurafsky e Martin, 2025). Tais modelos são pré-treinados em grandes corpora textuais e refinados por meio de técnicas de *fine-tuning*, isto é, um ajuste supervisionado posterior no qual o modelo pré-treinado

é adaptado a uma tarefa específica usando um conjunto menor de exemplos rotulados (geralmente com a adição de uma camada de saída apropriada e a atualização de todos ou parte dos parâmetros com taxa de aprendizado reduzida), permitindo sua adaptação a tarefas como a avaliação automática de questões discursivas (Devlin et al., 2019).

Esses modelos geram *embeddings*, que codificam as relações semânticas entre os termos a partir do contexto em que estão inseridos (Jurafsky e Martin, 2025). Ao contrário das representações tradicionais baseadas em frequência de palavras, os *embeddings* produzidos por modelos como o BERT são sensíveis à ordem e à coocorrência, o que os torna especialmente eficazes na análise de respostas abertas (Jurafsky e Martin, 2025). Essa capacidade é particularmente útil no contexto de AVAs, onde a avaliação discursiva demanda sensibilidade às variações linguísticas dos alunos (Oliveira, Pozzebon e Santos, 2020).

Nesse sentido, os modelos de PLN viabilizam a identificação de similaridades semânticas entre as respostas fornecidas pelos estudantes e os enunciados de referência, permitindo a automatização da correção com maior profundidade interpretativa, especialmente em tarefas que envolvem argumentação, explicação ou justificação textual (Almeida e Moura, 2024).

2.3 *LARGE LANGUAGE MODELS* E ABORDAGENS AVANÇADAS

Os *Large Language Models* (LLMs) são sistemas de IA projetados para compreender e gerar linguagem humana, frequentemente caracterizados como modelos de base (*foundation models*) treinados em larga escala e adaptáveis a múltiplas tarefas (Bommasani et al., 2021; Jurafsky & Martin, 2025). Eles são treinados com grandes volumes de dados textuais e utilizam a arquitetura *Transformer*, que modela dependências contextuais por meio de mecanismos de atenção (Vaswani et al., 2017; Devlin et al., 2019). Modelos como BERT e o GPT tornaram-se referências e têm mostrado alta eficácia em tarefas de PLN como tradução, sumarização, análise de sentimentos e geração de texto (Devlin et al., 2019; Brown et al., 2020; Achiam et al., 2023). O treinamento em grandes corpora permite que esses modelos aprendam padrões de coocorrência e uso contextual da linguagem, gerando respostas coerentes e adequadas a diferentes domínios (Brown et al., 2020; Bommasani et al., 2021). Entre as vantagens centrais dos LLMs estão a

robustez a variações linguísticas e a capacidade de lidar com ambiguidades, o que os torna apropriados para tarefas que exigem interpretação aprofundada, como a correção de respostas discursivas (Achiam et al., 2023; Jurafsky & Martin, 2025).

No campo educacional, a abordagem *LLM-as-judge* tem ganhado destaque, na qual modelos de linguagem atuam como avaliadores automáticos em tarefas abertas (Zheng et al., 2023; Gu et al., 2024). Essa abordagem baseia-se na capacidade dos LLMs de analisar dimensões como coerência, precisão conceitual, completude e relevância, ao mesmo tempo em que a literatura mapeia limites e recomendações para garantir confiabilidade (Gu et al., 2024; Zheng et al., 2023). Com *prompts* (instruções textuais que orientam o modelo sobre a tarefa e os critérios de julgamento), rubricas[8], matrizes de avaliação com descritores e níveis de desempenho que padronizam o julgamento e o *feedback*, bem definidas, estudos têm mostrado resultados promissores em *autograding* (atribuição automática de notas e *feedback* por sistemas computacionais) de respostas curtas e ensaios, ainda que se recomende validação humana (Schneider; Schenk; Niklaus, 2024; Kortemeyer, 2024).

Em AVAs como o SisTIRA, a integração de LLMs desponta como via promissora para automatizar a avaliação discursiva e acelerar ciclos de *feedback*, desde que se mantenha monitoramento e calibração pedagógica contínuos, pois modelos podem ser sensíveis ao *prompt* e à rubrica adotada, sofrer deriva de distribuição (mudanças no tipo de respostas ao longo do tempo), expressar vieses e apresentar variação entre turmas; por isso, auditorias periódicas com respostas-âncora, amostragens revisadas por avaliadores humanos e ajustes de critérios/ponderações são necessários para preservar validade, confiabilidade e equidade das notas (Grévisse, 2024; Schneider; Schenk; Niklaus, 2024; Zheng et al., 2023; Gu et al., 2024).

2.3.1 *Gemini-1.5-flash-8b*

O *Gemini-1.5-flash-8b* é um modelo de linguagem avançado desenvolvido pelo *Google* como parte da linha *Gemini*[9]. Esse modelo é uma versão mais poderosa

[8] Rubrica é o conjunto de critérios com descritores de desempenho que padroniza o julgamento e orienta o *feedback*.

[9] *Google AI for Developers:* https://ai.google.dev/gemini-api/docs?hl=pt-br

e aprimorada da arquitetura *Gemini*, projetada para realizar tarefas de PLN com um foco particular em eficiência e precisão em contextos de alta complexidade[10, 11, 12].

O *Gemini-1.5-flash-8b* é um modelo de 8 bilhões de parâmetros, o que permite que ele capture uma quantidade consideravelmente grande de padrões linguísticos e contextuais em dados textuais. Ele foi treinado em grandes volumes de dados de texto para ser capaz de entender e gerar linguagem humana com um nível muito alto de sofisticação, realizando tarefas como tradução, geração de texto, sumarização e análise semântica[13].

O modelo adota a arquitetura *Transformer*, que foi inicialmente proposta por Vaswani et al. (2017), e é um dos pilares dos modelos de linguagem modernos. Sua capacidade de entender relações contextuais e gerar respostas com base em dados fornecidos torna-o especialmente útil para aplicações em ambientes como AVAs, onde a análise e a correção de respostas discursivas são necessárias. A vantagem do *Gemini-1.5-flash-8b* em comparação com outros modelos de linguagem é sua alta precisão na geração de respostas e *feedbacks* detalhados, fazendo com que seja uma ferramenta muito útil para automatizar o processo de avaliação de respostas abertas, como as de questões discursivas, e oferecendo resultados mais dinâmicos e com uma maior compreensão do conteúdo textual[14].

Esse modelo de linguagem avançada é particularmente eficaz no contexto de *LLM-as-judge,* onde ele pode ser empregado para realizar correções automáticas, não apenas com base em palavras-chave, mas também considerando a semântica e a estrutura argumentativa das respostas. Isso permite que plataformas de avaliação, como o SisTIRA, utilizem o *Gemini-1.5-flash-8b* para fornecer uma análise mais profunda das respostas dos alunos, acompanhada de *feedbacks* contextuais e bem estruturados, ajustados conforme o nível de complexidade da questão.

3 TRABALHOS RELACIONADOS

A adoção de ferramentas em AVAs tem avançado com soluções que automatizam parte da avaliação de respostas abertas, tendência mapeada em

[10] Google For Developers: https://developers.googleblog.com/en/gemini-15-flash-8b-is-now-generally-available-for-use/
[11] Google AI For Developers: https://ai.google.dev/gemini-api/docs/models
[12] Google Cloud: https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/1-5-flash
[13] Google Research: https://research.google.com
[14] Google Research: https://research.google.com

revisões sobre *Automatic Short Answer Grading* (ASAG), isto é, correção automática de respostas curtas (Burrows; Gurevych; Stein, 2015). No ecossistema do *Moodle*[15, 16], plataforma de gestão de aprendizagem de código aberto, *plugins*[17] como o *Essay (auto-grade)*[18, 19] permitem configurar regras baseadas em padrões textuais, com nota preliminar e *feedback* curto. A documentação oficial e o diretório de *plugins* enfatiza a natureza configurável e os requisitos de instalação e manutenção por versão do *Learning Management System* (LMS), plataforma de gestão do ensino (Bateson, 2024; Bateson, 2025). Em paralelo, a *Teachy* divulga um conjunto de ferramentas com IA para apoiar avaliações e gestão de atividades, destacando correção automática para itens objetivos, inclusive com fluxo via cartão-resposta e *Quick Response Code (QR Code)*, um código bidimensional legível por câmera, e revisão manual para respostas discursivas, o que evidencia o desafio semântico em perguntas abertas[20, 21].

Do ponto de vista metodológico, a literatura de ASAG consolidou uma vertente clássica baseada na engenharia de atributos (*features*) e no cálculo de similaridade entre a resposta do aluno e gabaritos de referência (Burrows; Gurevych; Stein, 2015). Revisões abrangentes descrevem a evolução do uso de traços léxico-sintáticos, alinhamentos de dependência e medidas semânticas, apontando limites de portabilidade entre idiomas e domínios (Burrows; Gurevych; Stein, 2015). Resultados seminais indicam ganhos ao combinar medidas de similaridade com alinhamentos de grafos de dependência para prever notas em respostas curtas (Mohler; Bunescu; Mihalcea, 2011). Para garantir comparabilidade, *benchmarks*, conjuntos de dados e protocolos padronizados de avaliação, como o *SemEval-2013 Task 7 (Student Response Analysis)* e o ASAP *Short Answer Scoring*[22], estabeleceram protocolos, cenários *question-unseen* (teste em questões não vistas no treinamento) e métricas de concordância com humanos, fomentando reprodutibilidade e análises por domínio (Dzikovska et al., 2013).

---

[15] *Moodle*: https://moodle.org
[16] *Moodle Docs*: https://docs.moodle.org/en/About_Moodle
[17] *Moodle Plugins Directory*: https://moodle.org/plugins/
[18] *Moodle Essay auto-grade*: https://docs.moodle.org/en/Essay_%28auto-grade%29_question_type
[19] *Moodle Essay auto-grade question-type*: https://moodle.org/plugins/qtype_essayautograde
[20] *Teachy Helping Center*: https://intercom.help/teachyeducation/pt-BR/articles/11473333-enviar-avaliacoes-para-os-alunos
[21] *Teachy*: https://www.teachy.com.br/
[22] *The Hewlett Foundation*: https://www.kaggle.com/c/asap-sas

Em contraste com essa trilha, a indústria de *Automated Essay Scoring* (AES), correção automática de redações/ensaios, concentra-se na avaliação de textos extensos. O *e-rater*[23] reporta concordância elevada com avaliadores humanos em tarefas de redação (Attali; Burstein, 2006). De modo similar, o *IntelliMetric* apresenta bons níveis de concordância adjacente em estudos empíricos (Rudner; Garcia; Welch, 2006). Apesar da maturidade, esses motores priorizam a qualidade de escrita em ensaios e não a verificação conceitual típica de respostas curtas. A transferência direta para questões objetivas ou discursivas de pequeno porte requer cautela metodológica (Attali; Burstein, 2006; Burrows; Gurevych; Stein, 2015).

Avanços recentes em *embeddings* vetoriais (representações densas de significado) e em modelos *Transformers* (arquitetura neural dominante em PLN) ampliaram a robustez a paráfrases e à variação lexical. O *Sentence-BERT* (SBERT), variação do BERT voltada a similaridade entre sentenças, demonstra ganhos consistentes em tarefas de similaridade semântica (Reimers; Gurevych, 2019), e há evidências de que abordagens com SBERT melhoram a generalização para questões não vistas em ASAG, reduzindo a dependência de traços de superfície (Condor; Litster; Pardos, 2021).

Estudos sobre *LLM-as-judge*, isto é, o uso de um modelo de linguagem de grande porte como avaliador guiado por rubricas, critérios explícitos de avaliação, indicam correlações competitivas com humanos quando se empregam *prompts* cuidadosos e saídas estruturadas (Schneider; Schenk; Niklaus, 2024). Em áreas próximas, como *Health Question Answering* (HQA), combinam-se traços textuais e sinais contextuais para predizer qualidade de respostas, discutindo calibração e vieses, temas também relevantes para avaliação educacional (Qiu et al., 2022; Hu et al., 2017). No cenário lusófono, aplicações com *embeddings* e medidas de similaridade sugerem viabilidade para respostas curtas em português do Brasil, embora a escassez de corpora públicos amplos ainda imponha desafios de comparação (Almeida; Moura, 2024; Oliveira, 2019).

Apesar dos avanços tecnológicos, persistem algumas lacunas como a sensibilidade a domínio e idioma, dependência de traços de superfície e de *benchmarks* concentrados no eixo anglófono, além da necessidade de protocolos com juízes humanos em cenários *question-unseen* que assegurem reprodutibilidade e análise de vieses (Burrows; Gurevych; Stein, 2015; Dzikovska et al., 2013). Em

[23] Página do *e-rater*: https://www.ets.org/erater.html

resposta, estudos recentes com representações distribuídas e *Transformers*, incluindo abordagens de autocorreção (*autograding*) com LLM e uso de *prompts* e rubricas explícitas, têm mostrado aumento de robustez semântica e de correlação com avaliadores humanos, preservando saídas estruturadas que favorecem auditabilidade (Reimers; Gurevych, 2019; Condor; Litster; Pardos, 2021; Schneider; Schenk; Niklaus, 2024).

À luz desse panorama, o SisTIRA é situado como um avaliador de respostas discursivas em português do Brasil alinhado ao paradigma *LLM-as-judge*, com um modelo de linguagem atuando como avaliador externo guiado por rubricas e gabaritos para produzir nota normalizada e *feedback* breve. Essa posição dialoga com evidências de que *embeddings* e *Transformers* ampliam a robustez semântica e a capacidade de generalização para questões não vistas, ao mesmo tempo em que preservam interpretabilidade por meio de instruções e saídas estruturadas (Reimers; Gurevych, 2019; Condor; Litster; Pardos, 2021; Schneider; Schenk; Niklaus, 2024).

A seguir, no Quadro 1, segue um quadro comparativo de cada um dos trabalhos relacionados, a tarefa para as quais eles se aplicam, suas abordagens, a maneira como foram avaliados e as principais características de cada um de maneira resumida.

Quadro 1 – Síntese comparativa dos trabalhos relacionados e do SisTIRA.

Trabalho (ano)	**Tarefa**	**Abordagem**	**Avaliação (evidência)**	**Observações-chave**
e-rater (Attali & Burstein, 2006)	AES	*Features* linguísticas + regras	Concordância alta com humanos	Foco em ensaios; sistema proprietário; EN
IntelliMetric (Rudner; Garcia; Welch, 2006)	AES	AM com *features* linguísticas (sistema proprietário)	Concordância adjacente com humanos	Foco em ensaios longos; não visa verificação conceitual de respostas curtas; EN
Mohler et al. (2011)	ASAG	Similaridade semântica + dependências	Regressão/clas-sificação vs. humanos	Bom para curtas; portabilidade limitada; EN
SemEval-2013 Task 7	*Benchmark* ASAG	Tarefa/avaliação padrão para múltiplos sistemas	Protocolo *question-unsee*	*Benchmark* em EN; facilita

(Dzikovska et al., 2013)			*n*; métricas de concordância	comparabilidade; não é um sistema
SBERT (Reimers & Gurevych, 2019)	Similaridade de sentenças	*Transformer* siamês (embeddings)	Ganhos em STS/robustez a paráfrases	Base útil para ASAG; melhora generalização
Condor; Litster; Pardos (2021)	ASAG	*SBERT* + ajustes (regressão/classificação)	Ganhos em question-unseen vs. features tradicionais	Melhor generalização; reduz dependência de traços de superfície; corpora EN
Schneider et al. (2024)	LLM-*as-judge*	LLM guiado por *prompts*/rubricas	Correlação competitiva com humanos	Requer rubricas sólidas e validação humana
Kortemeyer (2024)	*Short-answer* com LLM	LLM generalista (GPT-4) como avaliador	Correlação/acurácia por item vs. humanos	Resultados fortes; recomenda supervisão humana; cenário de graduação; EN
SisTIRA	LLM-as-judge	LLM + rubricas + gabaritos	Consistência	Integração AVA; baixo custo/latência; nota 0.0–1.0 + *feedback*

Fonte: Própria do autor.

Como pode-se perceber analisando o Quadro 1, o SisTIRA se difere de sistemas consolidados no eixo anglófono, como o *e-rater* (Attali; Burstein, 2006) e o *IntelliMetric* (Rudner; Garcia; Welch, 2006), que se concentram na avaliação de redações extensas e privilegiam aspectos formais da escrita, pois volta-se especificamente para respostas discursivas curtas, priorizando a verificação conceitual em português do Brasil, um domínio ainda pouco explorado (Almeida; Moura, 2024; Oliveira, 2019). Concomitantemente, em contraste com motores de AES voltados à qualidade de escrita de ensaios, privilegia-se aqui a verificação de adequação conceitual em respostas discursivas.

4 METODOLOGIA

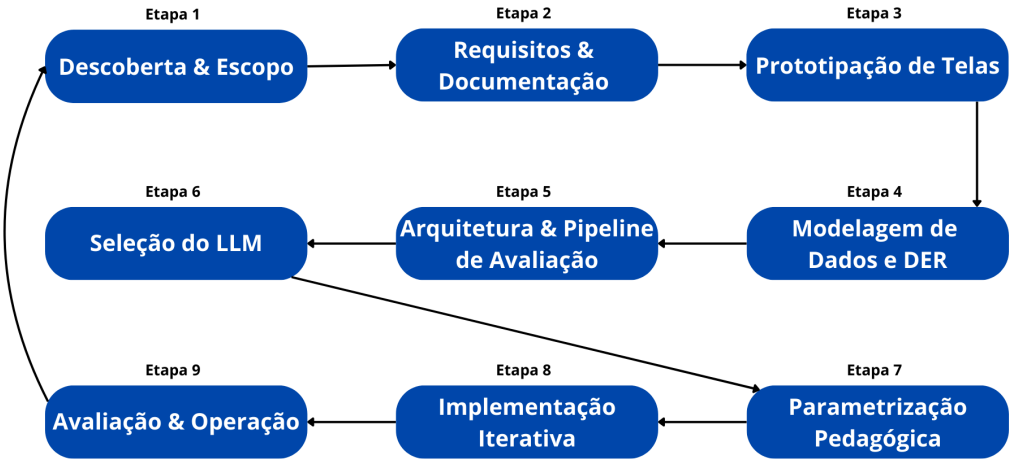
A metodologia adotada para o desenvolvimento e validação do SisTIRA caracteriza-se como experimental e de desenvolvimento, pois combina a construção de um artefato de *software* com a avaliação empírica do seu comportamento em condições controladas. O enquadramento segue o paradigma de *Design Science Research* (DSR), no qual o conhecimento avança por ciclos de construir e avaliar

aplicados a problemas reais, sustentados por base teórica e critérios explícitos de verificação (Peffers et al., 2007; Wieringa, 2014).

No presente estudo, a problemática, encontrada por meio de análise bibliográfica, é a correção de respostas discursivas curtas em AVAs; o artefato é um avaliador baseado em PLN capaz de atribuir nota contínua (0.0–1.0) e gerar *feedback* estruturado; e a avaliação inicial foca consistência e reprodutibilidade.

O encadeamento iterativo e o fluxo lógico entre etapas estão sintetizados no fluxograma da Figura 1, que guia a narrativa desta seção e será retomado ao longo do texto para situar as decisões metodológicas.

Figura 1 – Fluxo metodológico do SisTIRA.



Fonte: Própria do autor.

Etapa 1 - Descoberta e Escopo: Como indicado no primeiro nó da Figura 1, foram delimitados o problema, as metas de solução e as restrições de contexto, combinando revisão do estado da arte e análise do ecossistema institucional. Essa etapa estabelece relevância e objetivos mensuráveis, conforme preconiza a DSR para o início do ciclo construir-avaliar (Peffers et al., 2007).

Etapa 2 - Requisitos e Documentação: A partir do escopo, especificaram-se requisitos funcionais (por exemplo, cadastro de itens discursivos, associação de critérios e exemplos, avaliação automática com nota e *feedback* e registro de tentativas) e não funcionais (ligados à qualidade de produto e operação), como

latência, escalabilidade, custo por processamento, segurança, privacidade e auditabilidade, tomando como referência literatura de requisitos e modelos de qualidade (Sommerville, 2015; ISO/IEC 25010). Estes artefatos documentais dão rastreabilidade entre objetivos, desenho e verificação, sustentando as transições indicadas para as etapas seguintes na Figura 1.

Etapa 3 - Prototipação de Telas: Conforme o encadeamento da Figura 1, foram construídos protótipos de baixa e média fidelidade para os fluxos do docente e do estudante. A validação por inspeção considerou objetivos pedagógicos e heurísticas de usabilidade, o que reduz retrabalho anterior à implementação e realiza avaliação formativa dentro do ciclo iterativo recomendado pela DSR (Wieringa, 2014).

Etapa 4 - Modelagem de Dados e DER: A modelagem conceitual e o diagrama entidade-relacionamento estabeleceram entidades como itens, tentativas, critérios, exemplos de respostas e relatórios, além de relacionamentos e restrições que garantem rastreabilidade entre critérios aplicados, justificativas geradas e notas atribuídas. Essa base informa a conexão com a arquitetura mostrada na Figura 1.

Etapa 5 - Arquitetura e *Pipeline* de Avaliação: Definiu-se a arquitetura lógica do sistema e o *pipeline*: ingestão da resposta do estudante, montagem do *prompt* com rubrica e exemplos de referência, chamada ao modelo de linguagem, normalização da saída em nota contínua e justificativa mapeada aos critérios, persistência e exibição. O *pipeline* explicita pontos de controle para telemetria, logs e trilhas de auditoria, necessários à confiabilidade do artefato em DSR (Peffers et al., 2007).

Etapa 6 - Seleção do LLM: A escolha do modelo seguiu princípios sobre modelos de base e sua adoção responsável, priorizando custo e latência compatíveis com ciclos rápidos de *feedback*, janela de contexto suficiente para instruções e exemplos e estabilidade operacional para uso contínuo (Bommasani et al., 2021; Achiam et al., 2023). À luz desses critérios, optou-se por um modelo da família *Gemini*, adequado a cenários de alto volume com respostas rápidas. A decisão alinha-se à evidência de que LLMs podem atuar como avaliadores quando orientados por *prompts* e critérios claros, sobretudo em contextos que exigem escala (Zheng et al., 2023; Schneider; Schenk; Niklaus, 2024; Kortemeyer, 2024).

Etapa 7 - Parametrização Pedagógica: O mecanismo avaliador foi parametrizado com rubricas e gabaritos[24] de referência. Essa estratégia segue a tradição de *short-answer grading*, que utiliza respostas de referência e medidas de adequação para aproximar o julgamento automático do padrão docente e reduzir ambiguidades (Andrade, 2005; Brookhart, 2013; Dzikovska et al., 2013; Reimers; Gurevych, 2019; Condor; Litster; Pardos, 2021). A apresentação estruturada de critérios e exemplos no *prompt* utiliza *in-context learning*, prática comprovada para orientar LLMs a produzir saídas alinhadas (Brown et al., 2020).

Etapa 8 - Implementação Iterativa: A construção do *front end* e do *back end* ocorreu em ciclos curtos com instrumentação de *logs*, controle de parâmetros do modelo e validações de formato de saída. As iterações promoveram ajustes em *prompts*, descritores de critérios e curadoria de exemplos, mantendo o artefato alinhado ao contexto e ao rigor metodológico previsto pela DSR ao longo do desenvolvimento (Wieringa, 2014).

Etapa 9 - Avaliação e Operação: A avaliação utilizou um protocolo de consistência com 20 rodadas sobre um conjunto fixo de 20 questões distribuídas entre níveis básico, médio e difícil. Para estimar a estabilidade, mantiveram-se constantes os parâmetros de geração do modelo e calculou-se o desvio-padrão das notas por item, além de observar a uniformidade dos *feedbacks* entre rodadas. Os resultados desse acompanhamento retornam à Etapa 1, reabrindo o ciclo e orientando novos refinamentos.

A Figura 1 não apenas encadeia as atividades, mas também torna visíveis os laços de realimentação previstos por DSR. Resultados de avaliação e uso retroalimentam escopo, requisitos, arquitetura, seleção do LLM e parametrização pedagógica, garantindo relevância, rigor e utilidade do artefato ao longo do tempo

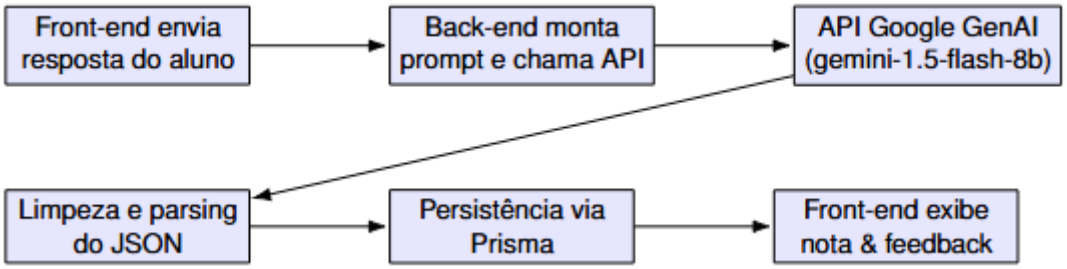
5 SISTIRA

O SisTIRA é um sistema de avaliação automatizada voltada a respostas discursivas. A aplicação disponibiliza operações para criação de avaliações, organização de questões em bancos reutilizáveis e gestão de usuários, incluindo mecanismos de registro e autenticação.

[24] Gabaritos são exemplos de respostas corretas, parcialmente corretas e incorretas fornecidos pelo docente para ancorar semanticamente a interpretação do modelo.

O fluxo principal (Figura 2) inicia com o envio da resposta do estudante, segue com a geração de um *prompt* de correção a partir de gabaritos e rubricas e conclui com a devolução de nota e *feedback*. Por padrão, o SisTIRA atribui notas na escala de 0.0 a 1.0 com incrementos de 0.1. A escala pode ser mapeada linearmente para 0 a 10 ou para porcentagem, o que facilita integração com diferentes formatos de boletins.

Figura 2 – Fluxograma horizontal em duas linhas do processo de correção automática no SisTIRA, mostrando o caminho completo da resposta do aluno até a exibição do *feedback*.



Fonte: Própria do autor.

Como ilustrado na Figura 2, a resposta do estudante chega à aplicação, que compõe o *prompt* com base no enunciado, nos gabaritos e nas rubricas. Em seguida ocorre a chamada ao serviço de PLN e a aplicação realiza a limpeza e validação do retorno em *JavaScript Object Notation* (JSON), formato leve, textual e independente de linguagem usado para estruturar e trocar dados em pares chave–valor. Depois, o sistema persiste os resultados e devolve ao cliente a nota e o *feedback*. As operações de leitura e escrita são registradas para auditoria e posterior análise de desempenho.

5.1 CAMADA DE *BACK END*

A camada de *back end* foi implementada com o *framework NestJS*[25] para *Node.js*[26], utilizando *TypeScript*[27] e arquitetura modular. A *Application Programming Interface* (API) organiza-se em módulos, em *controllers* que recebem e roteiam as requisições e em *services* que encapsulam a lógica de negócio. A documentação segue a Especificação *OpenAPI*[28] (OAS) e é publicada com *Swagger*[29], conjunto de

[25] Página do *NestJS*: https://nestjs.com/
[26] Página do *Node.js*: https://nodejs.org/pt
[27] Página do *TypeScript*: https://www.typescriptlang.org/
[28] Fonte da OAS: https://swagger.io/specification/
[29] Página do *Swagger*: https://swagger.io/

ferramentas que gera e testa a documentação de forma automatizada. Os *controllers* tratam o transporte e a validação estrutural dos *Data Transfer Objects* (DTOs), enquanto os *services* implementam as regras relacionadas a avaliações, questões e respostas discursivas. A documentação exposta pelo *Swagger* facilita a evolução controlada do contrato e a geração de clientes.

O fluxo de correção segue um roteiro estável. Ao receber a resposta do estudante, o serviço recupera gabaritos de referência, quando disponíveis, e compõe um *prompt* para avaliação automática no paradigma de *LLM-as-judge*. Em seguida ocorre a limpeza e a validação do JSON retornado, a aplicação de regras de formatação de nota na escala 0.0 a 1.0 com incrementos de 0.1 e o registro persistente do resultado junto à resposta. Em caso de inconsistência de formato ou tempo excedido, o serviço retorna erro e não grava estado parcial.

Como princípio de segurança, o *back end* assume a camada cliente como potencialmente não confiável e, por isso, aplica controles na borda do servidor. Essa medida é importante porque o cliente pode ser manipulado, interceptado ou comprometido, o que expõe o sistema a adulteração de dados, elevação de privilégios e ataques de injeção; ao validar tudo no servidor, preservam-se a integridade e a confidencialidade das informações. A camada aplica validação de esquema, normalização de tipos, limites de tamanho de campo e rejeição de propriedades não previstas. A autenticação utiliza *JSON Web Token*[30] (JWT), credencial compacta assinada, e a autorização verifica propriedade de recursos e papéis de colaboração, garantindo que apenas usuários habilitados criem avaliações, associem questões e consultem resultados. Os dados sensíveis que compõem a nota e o *feedback* são sempre calculados no servidor, e quaisquer valores enviados pelo cliente para esses campos são ignorados. Para robustez operacional, adotam-se tempos limite em chamadas externas, tratamento explícito de erros, respostas consistentes em códigos do *Hypertext Transfer Protocol* (HTTP) e estrutura de mensagem, além de política controlada de repetição em falhas transitórias.

A orquestração com o serviço de PLN isola detalhes do provedor em um componente próprio, o que permite parametrizar o modelo, ajustar limites de geração e evoluir o *prompting* (técnicas de redação de instruções para o LLM) sem alterar o contrato da API. O serviço registra marcas temporais de criação e

[30] Página do JWT: https://www.jwt.io/

atualização das respostas, o que viabiliza consultas agregadas como médias por questão e séries temporais. Esse desenho mantém a camada de *back end* como guardiã da integridade dos dados, controlando todo o ciclo de vida da correção automática desde a chegada da resposta até a publicação do resultado para o cliente.

5.2 CAMADA DE INTERFACES (*FRONT END*)

A camada de interfaces é responsável por capturar respostas discursivas, exibir notas e *feedback* e apoiar, com uma interface intuitiva, docentes na criação e organização de avaliações. O cliente web foi desenvolvido com *Next.js*[31] (*framework React*[32] para aplicações web com renderização no servidor e no cliente) e *TypeScript* (*superset* tipado de *JavaScript*[33]), adotando componentes tipados e contratos de dados alinhados aos objetos transferidos pela API. Essa combinação reduz erros em tempo de execução e facilita a manutenção evolutiva da interface. O estilo visual utiliza *Tailwind CSS*[34] (*framework* utilitário de *Cascading Style Sheets*[35] (CSS)) em abordagem *utility-first*, permitindo compor *layouts* responsivos e consistentes a partir de utilitários reutilizáveis, com atenção a contraste, tamanhos escaláveis de fonte e hierarquia visual para melhorar a legibilidade em respostas longas.

A comunicação com o serviço de avaliação ocorre por meio de *Axios*[36] (cliente HTTP), com interceptadores para anexar credenciais, mapear erros em mensagens compreensíveis e aplicar política controlada de repetição em falhas transitórias. As páginas de envio de respostas adotam validação de campos, estados explícitos de carregamento e salvamento automático para minimizar perda de conteúdo em cenários de conectividade instável. Em rotas que exibem resultados agregados, a renderização do lado do servidor do *Next.js* pode antecipar conteúdo e melhorar indicadores de desempenho percebido, enquanto consultas subsequentes são feitas no cliente para atualização reativa de notas e *feedback*.

Princípios de acessibilidade e internacionalização são considerados desde o desenho dos componentes, com rotulagem semântica, navegação por teclado e

---

[31] Página do *Next.js*: https://nextjs.org/
[32] Página do *React*: https://react.dev/
[33] Fonte do *JavaScript*: https://developer.mozilla.org/pt-BR/docs/Web/JavaScript
[34] Página do *Tailwind CSS*: https://tailwindcss.com/
[35] Fonte do CSS: https://developer.mozilla.org/pt-BR/docs/Web/CSS
[36] Página do *Axios*: https://axios-http.com/ptbr/docs/intro

mensagens de erro claras. A organização da interface provê fluxos para docentes e estudantes: criação e gerenciamento de avaliações e bancos de questões, submissão de respostas discursivas, visualização de nota normalizada e *feedback* textual.

5.3 PERSISTÊNCIA E BANCO DE DADOS

A persistência é implementada sobre *PostgreSQL*[37] (um sistema de gerenciamento de banco de dados relacional) com o *ORM Prisma*[38] (*Object-Relational Mapping*, uma ferramenta que facilita a interação entre o banco de dados e a aplicação). O objetivo principal do *schema*[39] é sustentar o fluxo de avaliações discursivas: criação de avaliações, associação de questões, submissão de respostas pelos estudantes e registro do resultado da avaliação automática. Para cada resposta discursiva, armazenam-se o texto enviado pelo estudante, a pontuação produzida pela IA e o *feedback* correspondente, todos em formato estruturado, o que facilita a realização de consultas agregadas (consultas que combinam dados) essenciais ao relato dos resultados.

As respostas modelo por questão são opcionais e servem como referências para orientar a IA durante a correção. Quando presentes, essas respostas de exemplo ajudam a calibrar o *prompting* e a padronizar expectativas de conteúdo, favorecendo a consistência da pontuação e do *feedback*. Quando ausentes, a correção baseia-se apenas no enunciado, preservando a flexibilidade do autor da questão.

Do ponto de vista funcional, o armazenamento prioriza apenas o necessário para rastrear cada correção automática: vínculo entre avaliação, questão e autor da resposta, texto submetido, saída da IA (pontuação e *feedback*) e marcas temporais de criação (data e hora). Informações auxiliares, como alternativas para perguntas de múltipla escolha e bancos de questões reutilizáveis, são mantidas de forma a não interferir na análise de respostas discursivas. Boas práticas de integridade referencial (manter a consistência dos dados entre tabelas), índices para consultas frequentes (estruturas que otimizam a busca de dados) e migrações versionadas

[37] Página do *PostgreSQL*: https://www.postgresql.org/
[38] Página do *ORM Prisma*: https://www.prisma.io/
[39] *Overview on Prisma Schema*: https://www.prisma.io/docs/orm/prisma-schema/overview/

(atualizações controladas do esquema do banco de dados) asseguram consistência e evolução controlada do esquema[40].

6 RESULTADOS OBTIDOS

Esta seção descreve um experimento controlado para avaliar a consistência do processo de correção automática empregado pelo SisTIRA. Para realização dos testes, foram executadas 20 rodadas completas de avaliação com o modelo *gemini-1.5-flash-8b* sobre um conjunto fixo de 20 questões abrangendo níveis básico, médio e difícil. Todas as rodadas utilizaram o mesmo *prompt* e a mesma escala de notas (0.0 a 1.0 com incrementos de 0.1), de modo a isolar a variabilidade intrínseca do modelo por meio de condições idênticas de entrada.

Os resultados, presentes na Tabela 1 e na Tabela 2, indicam estabilidade elevada. Em 19 das 20 questões o desvio-padrão das notas foi 0.00, e apenas a questão #16 apresentou desvio-padrão 0.05. O *feedback* textual também variou pouco por questão, usualmente entre 1 e 2 versões, com maior diversidade nos itens #9 e #18. A nota média global foi 0.638, com mediana 0.70, mínimo 0.00 e máximo 1.00, o que é compatível com um cenário misto de acertos parciais e totais em parte do conjunto, e respostas com pontuação baixa em outros itens.

Para contextualizar as métricas, abaixo existem três exemplos sintéticos de questões e respostas, separadas por nível de dificuldade. Esses exemplos ilustram o tipo de julgamento que o modelo produz quando o enunciado demanda definições, distinções conceituais ou enunciados de teoremas.

- **Básico**
  - **Pergunta:** “O que significa CPU em computação?”
  - **Resposta do aluno:** “CPU significa *Central Processing Unit*, ou Unidade Central de Processamento, responsável por executar instruções e processar dados no computador.”
  - **Gabarito ERRADO:** “CPU é um tipo de memória.”
  - **Gabarito MÉDIO:** “CPU é um componente que processa instruções, conhecido como processador ou unidade central.”

[40] Diagrama entidade relacionamentos (DER): https://drive.google.com/drive/folders/10CwhtxfEr8gN1MaTCba4sC9oOeBeGu0x?usp=sharing

    - **Gabarito CORRETO:** “CPU significa Central Processing Unit, ou Unidade Central de Processamento, responsável por executar instruções e processar dados no computador.”
    - **Nota:** 1.0
    - **Feedback:** “Resposta completa e precisa, demonstrando total compreensão do conceito.”
- **Médio**
    - **Pergunta:** “Qual a diferença entre HTTP e HTTPS?”
    - **Resposta do aluno:** “HTTPS usa criptografia mas funciona de forma semelhante ao HTTP.”
    - **Gabarito ERRADO: “**Não tem diferença, ambos são protocolos de e-mail.”
    - **Gabarito MÉDIO:** “HTTPS tem um S que significa seguro e usa criptografia, mas funciona similar ao HTTP.”
    - **Gabarito CORRETO:** “HTTPS (HTTP Secure) é HTTP sobre TLS/SSL, criptografando a comunicação entre cliente e servidor, garantindo confidencialidade e integridade, enquanto HTTP transmite dados em texto claro.”
    - **Nota:** 0.7
    - **Feedback:** “A resposta menciona a criptografia, mas não explica a diferença fundamental de segurança e integridade da comunicação.”
- **Difícil**
    - **Pergunta:** “O que é o Teorema CAP em bancos de dados distribuídos?”
    - **Resposta do aluno:** “O Teorema CAP diz que não dá pra ter consistência, disponibilidade e tolerância a partições ao mesmo tempo em sistemas distribuídos.”
    - **Gabarito ERRADO:** “CAP é uma marca de cerâmica.”
    - **Gabarito MÉDIO:** “CAP indica que um sistema distribuído não pode ter consistência, disponibilidade e tolerância a partições ao mesmo tempo.”
    - **Gabarito CORRETO:** “O Teorema CAP afirma que em sistemas distribuídos só é possível garantir, simultaneamente, duas entre Consistência, Disponibilidade e Tolerância a Partições; ao se perder

partições de rede, deve-se escolher entre consistência e disponibilidade.”

- **Nota:** 0.9
- **Feedback:** “Resposta correta e concisa. A menção de que a escolha entre consistência e disponibilidade é necessária em caso de partições reforçaria ainda mais a resposta.”

Esses casos sinalizam como a pontuação tende a ser máxima quando a definição é direta e completa, intermediária quando a resposta menciona apenas um aspecto relevante sem detalhar implicações, e alta porém não máxima quando o conteúdo central está correto, mas faltam elementos de reforço conceitual.

Prompt de correção utilizado:

Questão: <texto da questão>

Respostas de referência:
ERRADA: <texto da resposta ERRADA>
MÉDIA: <texto da resposta MÉDIA>
CORRETA: <texto da resposta CORRETA>

Resposta do aluno: <texto da resposta do aluno>

Instruções:

1. *Atribua uma nota entre 0.0 e 1.0 (em incrementos de 0.1).*
2. *Forneça um feedback breve (máx. 2 frases) justificando a nota.*
3. *Retorne apenas um JSON válido (sem texto extra) com as chaves “nota” e “feedback”.*
4. *Use ponto (.) como separador decimal.*

Tabela 1 – Métricas de análise preliminar por questão (20 rodadas)

ID	Nota Média	Desvio-padrão	*Feedbacks* únicos
#1	1.00	0.00	2
#2	0.00	0.00	1

#3	0.70	0.00	1
#4	0.70	0.00	1
#5	0.00	0.00	1
#6	1.00	0.00	1
#7	0.90	0.00	1
#8	0.20	0.00	3
#9	0.90	0.00	5
#10	0.90	0.00	1
#11	0.90	0.00	1
#12	0.70	0.00	1
#13	0.60	0.00	4
#14	0.00	0.00	2
#15	0.50	0.00	1
#16	0.96	0.05	2
#17	0.90	0.00	1
#18	0.60	0.00	7
#19	0.70	0.00	1
#20	0.60	0.00	2

Fonte: Própria do autor.

Tabela 2 – Posição das questões por faixa de nota média.

Nota média	IDs das questões
1.00	#1, #6
0.96	#16
0.90	#7, #9, #10, #11, #17
0.70	#3, #4, #12, #19
0.60	#13, #18, #20
0.50	#15
0.20	#8
0.00	#2, #5, #14

Fonte: Própria do autor.

A Tabela 1 mostra que, em 20 rodadas para 20 questões, as médias permaneceram estáveis e o desvio foi nulo na maioria dos itens. Para facilitar a leitura, a Tabela 2 organiza as questões por faixas de nota média. Observa-se concentração de itens com médias altas entre 0.90 e 1.00, além de um grupo com

médias muito baixas em 0.00, o que reforça que o conjunto de questões contempla diferentes níveis de exigência. O número de *feedbacks* únicos por questão foi baixo em média (1.95), com casos de maior variabilidade nos itens #9 e #18, possivelmente por abrangerem respostas com maior espaço interpretativo.

Em termos de interpretação, a dispersão praticamente nula em 19 itens sugere que, com *prompt* fixo e ambiente controlado, o processo de correção automática apresenta comportamento determinístico, ou quase determinístico, para a maioria das questões. A leve variação observada na questão #16 indica um ponto de atenção para inspeção do enunciado ou dos exemplos de referência usados no *prompt*, que podem abrir margem a múltiplas leituras de conteúdo. A diversidade reduzida de *feedback* por questão está alinhada com a instrução de produzir mensagens breves e estruturadas, o que favorece auditabilidade e reprodutibilidade.

Embora a consistência seja um requisito importante, este experimento não mede acurácia. A principal limitação é que a análise valida apenas repetibilidade, isto é, a capacidade do sistema de retornar avaliações idênticas para a mesma entrada em rodadas distintas. Um sistema pode ser perfeitamente consistente e, ainda assim, consistentemente incorreto. Para sustentar a eficácia do SisTIRA como ferramenta de avaliação, é indispensável comparar as notas e os *feedbacks* gerados pela IA com avaliações de especialistas humanos, estimando concordância por métricas como correlação, acerto por faixa de nota e coeficientes de concordância entre avaliadores. Essa comparação deve considerar cenários com e sem respostas modelo no *prompt*, além de análises por tópico e por nível de dificuldade.

Em resumo, a análise dos resultados obtidos indica alta reprodutibilidade do processo de correção com *gemini-1.5-flash-8b* sob *prompt* fixo e condições controladas. O próximo passo metodológico é incorporar avaliação de acurácia contra julgamentos humanos e conduzir estudos de sensibilidade a variações de *prompt* e parâmetros do modelo. A partir dessas extensões, será possível consolidar a evidência empírica sobre a efetividade do SisTIRA na avaliação de respostas discursivas em contextos educacionais.

7 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho apresentou o SisTIRA, uma plataforma web para avaliação automática de respostas discursivas em português, integrando um serviço de PLN

no paradigma *LLM-as-judge*. O sistema organiza avaliações, questões e respostas, e produz notas em escala de 0.0 a 1.0, com incrementos de 0.1, acompanhadas de *feedback* breve e estruturado, o que favorece auditabilidade e reuso pedagógico. Os resultados colhidos com 20 rodadas em 20 questões indicou alta consistência sob *prompt* fixo, com desvio-padrão nulo em 19 itens e apenas uma questão com leve variação, além de baixa variabilidade textual nos *feedbacks*, conforme a tabela de métricas apresentada. Esses achados reforçam a reprodutibilidade do processo de correção em condições controladas.

Entretanto, a principal limitação do presente trabalho é que a análise valida apenas repetibilidade, isto é, a capacidade do sistema de retornar avaliações idênticas para a mesma entrada em rodadas distintas. Sendo assim, um sistema pode ser perfeitamente consistente e, ainda assim, consistentemente incorreto. Além disso, existe o risco de manipulação adversarial por parte do respondente, como a inserção de instruções explícitas ou ocultas no texto para influenciar o modelo, o que pode enviesar notas e *feedbacks*; essa possibilidade deve ser considerada no desenho experimental e mitigada por medidas no lado do servidor. Para sustentar a eficácia do SisTIRA como ferramenta de avaliação, é indispensável comparar as notas e os *feedbacks* gerados pela IA com avaliações de especialistas humanos, estimando concordância por métricas como correlação, acerto por faixa de nota e coeficientes de concordância entre avaliadores. Essa comparação deve considerar cenários com e sem respostas modelo no *prompt*, além de análises por tópico e por nível de dificuldade.

Também é recomendável explicitar ameaças à validade. Quanto à validade interna, diferentes *prompts* ou pequenas variações de instrução podem alterar o *score* e o *feedback*. Tentativas de injeção de instruções no corpo da resposta também afetam a validade interna, inclusive por meio de texto oculto, caracteres invisíveis e marcações que busquem orientar o modelo. Esses riscos podem ser mitigados por sanitização e normalização de entrada, remoção de marcações e caracteres invisíveis, limites de tamanho de campo e validações sintáticas e semânticas antes do envio do conteúdo ao modelo. Quanto à validade externa, o tamanho reduzido do conjunto de questões e a ausência de testes com usuários finais limitam a generalização dos achados. A ampliação do conjunto para múltiplos domínios e a execução de estudos com docentes e discentes permitirão avaliar utilidade prática, interpretabilidade do feedback e impacto pedagógico.

Como trabalhos futuros, propõe-se: i) validação com avaliadores humanos, estimando concordância por coeficientes apropriados como o *Cohen's Kappa* e análises de correlação por questão e por faixa de nota; ii) experimentos de sensibilidade a *prompts* e resistência a instruções adversariais inseridas nas respostas, incluindo variações de instruções, exemplos e uso opcional de respostas modelo, com controle de parâmetros de geração; iii) ampliação do conjunto de questões e diversidade de domínios, acompanhada de estudos com docentes e discentes para aferir utilidade prática, interpretabilidade do *feedback* e impacto pedagógico; iv) instrumentação de métricas de acurácia e de erro por critério, explorando variações de escala e normalização de nota; v) extensão da plataforma para um aplicativo móvel que permita submissões e devolutivas em tempo real, notificações e fluxo de uso em contextos de baixa conectividade; vi) melhorias de engenharia para robustez e observabilidade, incluindo telemetria de falhas de integração, políticas de reprocessamento e monitoramento de deriva de dados; vii) estudo sistemático de robustez a ataques por parte do usuário, avaliando impacto de texto oculto e instruções adversariais e testando contramedidas, como sanitização e normalização de entrada, remoção de caracteres invisíveis, restrições de formatação, composição de *prompt* em camadas com campos delimitados, verificações automáticas de conformidade do formato de saída e auditoria amostral com revisão humana.

Em síntese, o SisTIRA configura uma solução viável para apoiar a correção de respostas discursivas em português, combinando produção de *feedback* breve com padronização de notas e integração a fluxos avaliativos já existentes. A consolidação de estudos de acurácia, a ampliação do escopo experimental e a disponibilização em plataformas móveis representam os próximos passos para sustentar, em escala, ganhos de agilidade, transparência e democratização na avaliação em AVAs.

REFERÊNCIAS

ACHIAM, Josh et al. **GPT-4 technical report**. arXiv preprint arXiv:2303.08774, 2023.

ALMEIDA, José Augusto O. S.; MOURA, Raimundo Santos. **Investigação de métodos de similaridade textual no contexto da avaliação automática de questões discursivas**. In: Anais da XII Escola Regional de Computação do Ceará, Maranhão e Piauí (ERCEMAPI 2024). 2024.

ANDRADE, Heidi. **Teaching with rubrics**. Educational Leadership, v. 57, n. 5, p. 13–18, 2000.

ATTALI, Yigal; BURSTEIN, Jill. **Automated essay scoring with e-rater® V.2**. Journal of Technology, Learning, and Assessment, v. 4, n. 3, 2006.

BOMMASANI, Rishi et al. **On the opportunities and risks of foundation models**. arXiv preprint arXiv:2108.07258, 2021.

BORTOLINI, Luana Cassol; NAUROSKI, Everson Araujo. **Desafios e emergências da avaliação da aprendizagem no contexto de pandemia: impactos na profissão docente**. Revista Educação & Formação, v. 7, n. e8252, p. 1–15, 2022. DOI: 10.25053/redufor.v7.e8252.

BROOKHART, Susan M. **How to create and use rubrics for formative assessment and grading**. Alexandria: ASCD, 2013.

BROWN, Tom B. et al. **Language models are few-shot learners**. In: Advances in Neural Information Processing Systems (NeurIPS 2020). 2020.

BRASIL. Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (Inep). **Censo da Educação Superior 2023: notas estatísticas**. Brasília: Inep, 2024.

BURROWS, Steven; GUREVYCH, Iryna; STEIN, Benno. **The eras and trends of automatic short answer grading**. International Journal of Artificial Intelligence in Education, v. 25, n. 1, p. 60–117, 2015. DOI: 10.1007/s40593-014-0026-8.

COHEN, Jacob. **A coefficient of agreement for nominal scales**. Educational and Psychological Measurement, v. 20, n. 1, p. 37–46, 1960. DOI: 10.1177/001316446002000104.

CONDOR, Aubrey; LITSTER, Max; PARDOS, Zachary. **Automatic short answer grading with SBERT on out-of-sample questions**. In: Proceedings of the 14th International Conference on Educational Data Mining (EDM 2021). 2021.

COSTA, Diana Barreto; LIMA, Perla Risette Alves; PASSOS, Raissa Evelyn Araujo de Almeida. **Reflexões sobre a avaliação da aprendizagem no ensino remoto, sob as perspectivas docente e discente, nos cursos de Letras-Inglês da UEMASUL. Contribuciones a las Ciencias Sociales**, v. 17, n. 3, p. 1–25, 2024. DOI: 10.55905/revconv.17n.3-231.

DANDE, Abhay A.; PUND, M. A. **A review study on applications of Natural Language Processing**. International Journal of Scientific Research in Science, Engineering and Technology, v. 10, n. 2, p. 122–126, 2023. DOI: 10.32628/IJSRSET2310214.

DEVLIN, Jacob; CHANG, Ming-Wei; LEE, Kenton; TOUTANOVA, Kristina. **BERT: pre-training of deep bidirectional transformers for language understanding**. In: Proceedings of NAACL-HLT 2019. 2019. DOI: 10.18653/v1/N19-1423.

DZIKOVSKA, Myroslava O. et al. **SemEval-2013 Task 7: The student response analysis**. In: Proceedings of SemEval 2013. 2013. p. 263–274.

GOMES, Alexandre Santos; PIMENTEL, Edson. **Ambientes Virtuais de Aprendizagem para uma educação mediada por tecnologias digitais. In: Informática na Educação: ambientes de aprendizagem, objetos de aprendizagem e empreendedorismo**. Porto Alegre: Sociedade Brasileira de Computação, 2021.

GRÉVISSE, Christian. **LLM-based automatic short answer grading in undergraduate medical education**. BMC Medical Education, v. 24, art. 1060, 2024.

GU, Jiawei et al. **A survey on LLM-as-a-judge**. arXiv preprint arXiv:2411.15594, 2024.

HEVNER, Alan R. et al. **Design science in information systems research**. MIS Quarterly, v. 28, n. 1, p. 75–105, 2004.

HU, Ze et al. **A deep learning approach for predicting the quality of online health expert question-answering services**. Journal of Biomedical Informatics, v. 71, p. 241–253, 2017. DOI: 10.1016/j.jbi.2017.06.012.

ISO/IEC. ISO/IEC 25010:2011 — **Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — System and software quality models**. Genebra: ISO/IEC, 2011.

JURAFSKY, Daniel; MARTIN, James H. **Speech and language processing**. 3. ed. [s.l.]: [s.n.], 2025.

KOO, T. K.; LI, M. Y. **A guideline of selecting and reporting intraclass correlation coefficients for reliability research**. Journal of Chiropractic Medicine, v. 15, n. 2, p. 155–163, 2016.

KORTEMEYER, Gerd. **Performance of the pre-trained large language model GPT-4 on automated short answer grading**. Discover Artificial Intelligence, v. 4, art. 47, 2024. DOI: 10.1007/s44163-024-00147-y.

KUSNER, Matt; SUN, Yu; KOLKIN, Nicholas; WEINBERGER, Kilian Q. **From word embeddings to document distances. In: Proceedings of the 32nd International Conference on Machine Learning** (ICML 2015). 2015. v. 37, p. 957–966.

LUCKESI, Cipriano Carlos. **Avaliação da aprendizagem escolar: estudos e proposições**. 23. ed. São Paulo: Cortez, 2011.

MEDEIROS, Josiane dos Santos de. **A aceitação tecnológica quanto ao uso do sistema tutor inteligente MAZK pelos docentes da educação básica: um estudo de caso em tempos de pandemia**. 2021. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, Araranguá, 2021.

MOHLER, Michael; BUNESCU, Razvan; MIHALCEA, Rada. **Learning to grade short answer questions using semantic similarity measures and dependency graph alignments**. In: Proceedings of ACL-HLT 2011. 2011. p. 752–762.

MURRAY, Tom. **An overview of intelligent tutoring system authoring tools: updated analysis of the state of the art**. In: MURRAY, Tom; BLESSING, Stephen B.; AINSWORTH, Shaaron (ed.). Authoring tools for advanced technology learning environments. Dordrecht: Springer, 2003. p. 491–544. DOI: 10.1007/978-94-017-0819-7_21.

NKAMBOU, Roger; BOURDEAU, Jacqueline; PSYCHÉ, Valéry. **Building intelligent tutoring systems: an overview.** In: NKAMBOU, Roger; BOURDEAU, Jacqueline; MIZOGUCHI, Riichiro (ed.). Advances in intelligent tutoring systems. Berlin; Heidelberg: Springer, 2010. p. 377–406. DOI: 10.1007/978-3-642-14363-2_18.

OLIVEIRA, Douglas Camilo de. **Aplicação das técnicas de Processamento de Linguagem Natural Cosine Similarity e Word Mover's Distance na automatização da correção de questões discursivas no sistema tutor inteligente MAZK**. 2019. Trabalho de Conclusão de Curso (Graduação) — Universidade Federal de Santa Catarina, Araranguá, 2019.

OLIVEIRA, Douglas; POZZEBON, Enzo; SANTOS, Tiago. **Aplicação de técnicas de processamento de linguagem natural na automatização de correção de questões discursivas**. In: Anais do Congresso Brasileiro de Informática na Educação (CBIE 2020). 2020.

PERRENOUD, Philippe. **Avaliação: da excelência à regulação das aprendizagens**. Porto Alegre: Artmed, 1999.

PREECE, Jenny; ROGERS, Yvonne; SHARP, Helen. **Interaction design: beyond human-computer interaction**. 5. ed. Hoboken: Wiley, 2019.

REIMERS, Nils; GUREVYCH, Iryna. **Sentence-BERT: sentence embeddings using Siamese BERT-networks**. In: Proceedings of EMNLP-IJCNLP 2019. 2019. p. 3982–3992.

RITTER, Steven; BLESSING, Stephen B.; WHEELER, Leslie. **Tools for component-based learning environments**. In: MURRAY, Tom; BLESSING, Stephen B.; AINSWORTH, Shaaron (ed.). Authoring tools for advanced technology learning environments. Dordrecht: Springer, 2003. p. 467–489. DOI: 10.1007/978-94-017-0819-7_20.

RUDNER, Lawrence; GARCIA, Veronica; WELCH, Catherine. **An evaluation of the IntelliMetric™ essay scoring system**. Journal of Technology, Learning, and Assessment, v. 4, n. 4, 2006.

SCHNEIDER, Johannes; SCHENK, Bernd; NIKLAUS, Christina. **Towards LLM-based autograding for short textual answers**. In: Proceedings of CSEDU 2024. 2024. DOI: 10.5220/0012552200003693.

SILVA, Alércio Charles. **Correção automática de questões discursivas de resposta curta: uma abordagem baseada em extração de informação**. 2023. Monografia (Graduação) — Universidade Federal do Maranhão, São Luís, 2023.

SOMMERVILLE, Ian. **Software engineering**. 10. ed. Boston: Pearson, 2015.

VASWANI, Ashish et al. **Attention is all you need**. In: Advances in Neural Information Processing Systems (NeurIPS 2017). 2017.

WORLD BANK; UNESCO; UNICEF. **The state of the global education crisis: a path to recovery**. Washington, DC; Paris; New York: [s.n.], 2021. Relatório conjunto.

WIERINGA, Roel. **Design science methodology for information systems and software engineering**. Berlin: Springer, 2014.

ZHENG, Lianmin et al. **Judging LLM-as-a-judge with MT-Bench and Chatbot Arena**. arXiv preprint arXiv:2306.05685, 2023.

AGRADECIMENTOS

Dedico este trabalho ao meu pai, Aldemar, que trabalhou sob o sol para que eu pudesse estudar à sombra.

Ao meu irmão, Adriano, pelo apoio constante — material e afetivo — que tornou possível esta caminhada. Com você, compartilhei risos, desafios e o sustento de cada dia.

Ao meu irmão, Arnaldo, pela dádiva do meu maior tesouro, meu primeiro sobrinho, e pelas memórias luminosas da nossa infância, que seguem me guiando.

À minha mãe, Josefa, por cada ensinamento e por palavras que, ditas com carinho, tornaram-se abrigo.

Aos amigos de longa data, Jean e Karielly, agradeço pelo aprendizado construído em conjunto, pela lealdade nas diferentes fases desta jornada e pela presença que sempre renovou meu ânimo de seguir.

Às minhas primeiras amigas de Picos, Camilla e Livya, com quem conheci novos ambientes acadêmicos, sorri e me senti acolhido.

Aos amigos Victor e Carlos, minha sincera gratidão. Com vocês, compartilhei alegrias e travessias; em cada uma, encontrei companhia e força.

Ao meu amigo João, com quem tanto ri, ficam guardados os momentos inesquecíveis — viagens, bolos e boas histórias. Admiro como o destino nos uniria em qualquer caminho acadêmico que escolhêssemos.

Ao meu namorado, presente que o IFPI me deu, agradeço pelo afeto sereno, pela paciência e pela companhia nos dias longos. No seu abraço, encontrei descanso; no seu sorriso, alegria; no seu cuidado, pertença.

Ao meu orientador, professor Rafael Torres Anchiêta, agradeço pelo apoio firme, pela generosidade intelectual e pela acolhida de sua família. Com o senhor, aprendi mais do que caberia nestas linhas.

Ao professor Rogério, a quem pude recorrer em tantos momentos, agradeço pelas lições, pela paciência e pela presença segura ao longo do percurso.

Ao IFPI, instituição que me acolheu por quase seis anos, agradeço pelos horizontes abertos, pelos mestres e pelos encontros que moldaram minha formação acadêmica e humana.

Agradeço, também, ao ChatGPT que, com minha supervisão, me acompanhou em toda a graduação, me ajudando quando ninguém mais pôde.

A todas as pessoas que, de perto ou de longe, tocaram esta obra com gestos, conselhos e esperança, deixo meu muito obrigado. Sem vocês, este caminho não teria a mesma luz.