

Feed-CLaRiCe: Uma abordagem neural para correção e feedbacks automáticos de redações

Title: Feed-CLaRiCe: A neural approach for automatic essay correction and feedback

Título: Feed-CLaRiCe: Un enfoque neuronal para la corrección automática y retroalimentación de redacciones

João Vitor Santos Tavares
Universidade Federal de Alagoas
ORCID: [0009-0003-4938-0489](https://orcid.org/0009-0003-4938-0489)
joaotavarestavares@hotmail.com

Luiz Rodrigues
Universidade Tecnológica Federal do Paraná
ORCID: [0000-0003-0343-3701](https://orcid.org/0000-0003-0343-3701)
luizrodrigues@utfpr.edu.br

Diego Dermeval
Universidade Federal de Alagoas /
Harvard University
ORCID: [0000-0002-8415-6955](https://orcid.org/0000-0002-8415-6955)
diego.matos@famed.ufal.br

Resumo

Escrever é uma habilidade essencial adquirida ao longo da formação educacional, empregada tanto na aquisição e representação de conhecimento quanto em sua avaliação. Em exames de larga escala como o Exame Nacional do Ensino Médio (ENEM), a redação dissertativo-argumentativa representa um componente decisivo na composição da nota final. Estudos anteriores investigaram a automatização da correção desses textos, mas poucos realizaram uma análise comparativa abrangente entre diferentes arquiteturas de redes neurais aplicadas a essa tarefa. Este trabalho propõe uma abordagem baseada em Redes Neurais Artificiais (ANNs) para prever simultaneamente as cinco competências avaliadas no ENEM, utilizando a base Extended Essay-BR. Os resultados obtidos indicam que modelos de Convolução 1D superam redes recorrentes, com Erro Absoluto Médio (MAE) variando entre 15,24 e 21,48 pontos, evidenciando sua robustez na tarefa de regressão. Como contribuição adicional, desenvolvemos uma aplicação que integra o modelo de correção com um Grande Modelo de Linguagem (LLM), permitindo a geração automatizada de feedback textual individualizado com base nas notas previstas. Essa funcionalidade visa enriquecer o processo de aprendizagem ao fornecer aos estudantes orientações qualitativas sobre seus pontos fortes e aspectos a melhorar, ampliando as possibilidades de uso de técnicas de Processamento de Linguagem Natural (NLP) no contexto educacional. A aplicação proposta representa um avanço significativo na direção de sistemas inteligentes de apoio à escrita avaliativa.

Palavras-chave: Correção automática de redações; Redes Neurais Artificiais; Modelo de Convolução 1D; ENEM; Processamento de Linguagem Natural; feedback automatizado; Grandes Modelos de Linguagem; Regressão multivariada; Extended Essay-BR; Avaliação educacional assistida por IA

Abstract

Writing is an essential skill acquired throughout educational development, used not only for knowledge acquisition and representation but also for its evaluation. In large-scale assessments such as the Brazilian National High School Exam (ENEM), the argumentative-essay plays a decisive role in the composition of the final score. Previous studies have explored the automation of essay scoring, but few have conducted a comprehensive comparative analysis of different neural network architectures for this task. This paper proposes an approach based on Artificial Neural Networks (ANNs) to simultaneously predict the five competencies evaluated in the ENEM, using the Extended Essay-BR dataset. The results indicate that 1D Convolutional models outperform recurrent networks, achieving Mean Absolute Error (MAE) values ranging from 15.24 to 21.48 across the five competencies, demonstrating robustness in regression tasks. As an additional contribution, we developed an application that integrates the scoring model with a Large Language

Model (LLM), enabling the automatic generation of individualized textual feedback based on predicted scores. This feature aims to enhance the learning process by providing students with qualitative guidance on their strengths and areas for improvement, thus expanding the applicability of Natural Language Processing (NLP) techniques in educational contexts. The proposed application represents a significant advancement toward intelligent systems for supporting evaluative writing and personalized education.

Keywords: Automated essay scoring; Artificial Neural Networks; 1D Convolutional Model; ENEM; Natural Language Processing; Automated feedback; Large Language Models; Multivariate regression; Extended Essay-BR; AI-assisted educational assessment

Resumen

Escribir es una habilidad esencial adquirida a lo largo del desarrollo educativo, utilizada no solo para la adquisición y representación del conocimiento, sino también para su evaluación. En evaluaciones a gran escala como el Examen Nacional de Educación Media (ENEM) de Brasil, la redacción disertativo-argumentativa cumple un papel decisivo en la composición de la nota final. Estudios anteriores han investigado la automatización de la corrección de ensayos, pero pocos han realizado un análisis comparativo profundo entre diferentes arquitecturas de redes neuronales aplicadas a esta tarea. Este trabajo propone un enfoque basado en Redes Neuronales Artificiales (ANNs) para predecir simultáneamente las cinco competencias evaluadas en el ENEM, utilizando la base de datos Extended Essay-BR. Los resultados indican que los modelos de Convolución 1D superan a las redes recurrentes, alcanzando errores absolutos medios (MAE) que oscilan entre 15.24 y 21.48 puntos, lo que demuestra su solidez en tareas de regresión. Como contribución adicional, se desarrolló una aplicación que integra el modelo de corrección con un Modelo de Lenguaje de Gran Escala (LLM), permitiendo la generación automática de retroalimentación textual individualizada a partir de las puntuaciones predichas. Esta funcionalidad tiene como objetivo enriquecer el proceso de aprendizaje proporcionando a los estudiantes orientación cualitativa sobre sus fortalezas y áreas de mejora, ampliando así las posibilidades de aplicación de técnicas de Procesamiento de Lenguaje Natural (PLN) en contextos educativos. La aplicación propuesta representa un avance significativo hacia sistemas inteligentes de apoyo a la escritura evaluativa.

Palabras clave: Corrección automática de ensayos; Redes Neuronales Artificiales; Modelo de Convolución 1D; ENEM; Procesamiento de Lenguaje Natural; Retroalimentación automatizada; Grandes Modelos de Lenguaje; Regresión multivariada; Extended Essay-BR; Evaluación educativa asistida por IA

1 Introdução

Em diversos países, é comum a utilização da escrita, sobretudo de textos dissertativo-argumentativos, não somente como forma de avaliação do conhecimento, mas também como parte importante de exames de admissão em instituições de nível superior. No cenário brasileiro, destaca-se o Exame Nacional do Ensino Médio (ENEM), que utiliza cinco provas de diferentes áreas do conhecimento para a composição das notas dos candidatos, sendo uma delas a redação. A nota da redação corresponde à soma das pontuações atribuídas às cinco competências avaliadas. Textos dissertativo-argumentativos também são amplamente utilizados no contexto escolar, tanto na composição dos currículos quanto na preparação para o ENEM. Nesse sentido, esse tipo textual pode ser explorado para a geração de *Learning Analytics*, conforme descrito por (Freitas et al., 2020), sendo útil para mapear níveis de proficiência em escrita em contextos educacionais diversos, desde salas de aula até populações mais amplas de candidatos.

A correção dessas redações é uma tarefa de alta demanda e relevância pedagógica, que exige leitura atenta por parte de avaliadores humanos. Entretanto, fatores como fadiga visual e cognitiva aumentam a probabilidade de erro, além de prolongarem o tempo necessário para a finalização das correções. Esse cenário pode ocasionar atrasos na divulgação dos resultados e dificultar a produção tempestiva de estatísticas de aprendizado, impactando negativamente a formulação de políticas educacionais baseadas em evidências.

Nos últimos anos, observou-se um avanço significativo nas pesquisas em Aprendizado de Máquina (em inglês, *Machine Learning* – ML), com destaque para técnicas de Aprendizado Profundo (do inglês *Deep Learning* - DL), que vêm se mostrando eficazes em diversas tarefas, incluindo a classificação de textos, a geração de textos e imagens, e a análise gramatical. Tais avanços impulsionaram o interesse na aplicação desses métodos no contexto educacional, especialmente no ensino gramatical e ortográfico, em que técnicas de Processamento de Linguagem Natural (em inglês, *Natural Language Processing* - NLP) podem automatizar tarefas tradicionalmente manuais (A. Barbosa & Campelo, 2020; Silva Filho et al., 2023). Nesse contexto, destaca-se o uso de Redes Neurais Artificiais (do inglês *Artificial Neural Networks* – ANNs), que têm se mostrado eficazes na modelagem de dados não estruturados, como o texto. Mais recentemente, o surgimento de Grandes Modelos de Linguagem (do inglês *Large Language Models* – LLMs) ampliou ainda mais as possibilidades de aplicação dessas tecnologias no cenário educacional.

Diante desse cenário, o objetivo deste artigo é avaliar a capacidade de modelos neurais na tarefa de correção automática de redações do ENEM, testando diferentes topologias de ANNs. Embora existam pesquisas similares (G. Barbosa et al., 2022; Oliveira et al., 2022), esses estudos não realizam uma comparação direta entre diferentes modelos de DL, especialmente no que se refere à aplicação de modelos de sequência, nem propõem uma abordagem de regressão conjunta das cinco competências por meio de um único modelo. Além disso, utilizam técnicas mais básicas de pré-processamento e vetorização, como a TF-IDF (do inglês *Term Frequency–Inverse Document Frequency*).

Este artigo, portanto, contribui com a literatura ao comparar diferentes arquiteturas de ANNs utilizando vetorização e pré-processamento mais robustos, e ao propor a utilização de dois modelos: um capaz de prever simultaneamente as cinco competências, e outro capaz de gerar *feedback* textual interpretável via LLM, além de propor uma aplicação que integra esses dois modelos em um

cenário prático de uso. Os resultados dos experimentos demonstraram que o modelo de Convolução 1D superou os modelos recorrentes na tarefa de correção, apresentando os melhores resultados tanto para o Erro Absoluto Médio (do inglês *Mean Absolute Error* – MAE) quanto para a Raiz do Erro Quadrático Médio (do inglês *Root Mean Square Error* – RMSE) nas cinco competências. Adicionalmente, os resultados obtidos após a otimização de hiperparâmetros indicam que modelos mais profundos podem alcançar melhor desempenho, desde que adequadamente regularizados.

Além disso, este trabalho propõe o uso de um LLM para a geração de feedback textual automatizado, com base nas notas previstas por modelos de regressão para as cinco competências do ENEM. Tal abordagem visa ir além da simples atribuição de notas numéricas, fornecendo aos estudantes um retorno qualitativo alinhado aos critérios da prova. A capacidade dos LLMs de interpretar e gerar linguagem natural os torna ferramentas valiosas para promover uma aprendizagem mais personalizada e eficaz. Assim, a integração com o LLM permitiu a geração de feedbacks personalizados, coerentes com os critérios da prova, sugerindo um caminho promissor para sistemas de apoio à escrita que integrem predição automática e retorno qualitativo.

A estrutura deste artigo é organizada da seguinte forma: a Seção 2 apresenta os principais conceitos teóricos sobre ML, DL e NLP. Na Seção 3 é realizada uma revisão da literatura, destacando os trabalhos relacionados à correção automática de redações. A Seção 4 detalha os procedimentos metodológicos empregados, incluindo a preparação dos dados e a configuração dos modelos. A Seção 5 descreve os experimentos realizados e analisa os resultados obtidos. Por fim, a Seção 6 apresenta as conclusões do estudo e sugere direções para trabalhos futuros.

2 Fundamentação Teórica

Esta seção apresenta uma breve introdução ao ML, ao DL e ao NLP, descrevendo o funcionamento de algumas das técnicas empregadas neste trabalho.

2.1 Aprendizado de Máquina e Aprendizado Profundo

O ML é um subcampo da Inteligência Artificial (Em Inglês, *Artificial Intelligence* - AI) que se baseia em algoritmos capazes de identificar padrões e inferir relações a partir de dados (Pranav & Sarma, 2023). Esses algoritmos utilizam conceitos de cálculo, álgebra linear e estatística para construir modelos matemáticos que aproximam funções desconhecidas, permitindo realizar tarefas como classificação, regressão ou agrupamento. As principais abordagens de ML são:

- **Aprendizado Supervisionado:** o modelo é treinado com exemplos rotulados, buscando aprender uma função que mapeia entradas em saídas conhecidas, como em tarefas de classificação e regressão;
- **Aprendizado Não Supervisionado:** o modelo identifica padrões ou estruturas latentes em dados não rotulados, aplicando técnicas de agrupamento ou redução de dimensionalidade;
- **Aprendizado por Reforço:** um agente interage com um ambiente e aprende políticas de ação por meio de recompensas e penalidades, otimizando seu comportamento ao longo do tempo.

O DL é uma subárea do ML que emprega ANNs com múltiplas camadas ocultas, conhecidas como Redes Neurais Profundas (em inglês, *Deep Neural Networks*, DNNs). Essas redes conseguem aprender representações hierárquicas de dados, tornando-se especialmente eficazes em tarefas complexas de percepção, como visão computacional e NLP (LeCun et al., 2015).

Entre as arquiteturas mais utilizadas destacam-se:

- **Redes Neurais Convolucionais (em inglês, *Convolutional Neural Networks* - CNNs):** projetadas para processar dados com estrutura espacial, como imagens e sinais bidimensionais. Utilizam operações de convolução para extrair automaticamente características locais, reduzindo a necessidade de engenharia manual de atributos (LeCun et al., 1998);
- **Redes Neurais Recorrentes (em inglês, *Recurrent Neural Networks* - RNNs):** adequadas para modelar dependências temporais em sequências, por manterem um estado interno que captura informações de entradas anteriores. No entanto, podem sofrer com o problema do desaparecimento ou explosão do gradiente (Bengio et al., 1994);
- **Long Short-Term Memory (LSTM) e Gated Recurrent Unit (GRU):** variantes das RNNs que introduzem mecanismos de portas para controlar o fluxo de informações, permitindo o aprendizado de dependências de longo prazo em séries temporais (Cho et al., 2014; Hochreiter & Schmidhuber, 1997).

Essas arquiteturas são particularmente relevantes para este trabalho, por serem utilizadas nos experimentos realizados, conforme detalhado na Seção 4.

2.2 Processamento de Linguagem Natural

O NLP é um subcampo da AI dedicado ao desenvolvimento de sistemas capazes de compreender, interpretar e gerar linguagem humana de forma automatizada (Jurafsky & Martin, 2026). Em termos práticos, o NLP aplica técnicas de ML a dados textuais ou de áudio para realizar tarefas como classificação de sentimentos, análise sintática, tradução automática, sumarização e resposta a perguntas (Cambria & White, 2014; Young et al., 2018).

De modo geral, o treinamento de algoritmos de NLP segue um fluxo composto por etapas fundamentais (Chai, 2023; Zhao, 2017):

1. **Remoção de palavras irrelevantes**, conhecidas como *stopwords*, que aparecem com alta frequência e pouca contribuição semântica (por exemplo, “a”, “o” ou “de”);
2. **Normalização das palavras**, realizada por meio de técnicas como lematização e *stemming*. Enquanto o *stemming* é mais rápido, mas pode gerar radicais inexistentes, a lematização reduz uma palavra à forma base, garantindo maior precisão linguística;
3. **Tokenização**: processo que divide frases em unidades menores denominadas *tokens*, que podem corresponder a palavras, subpalavras ou até caracteres;
4. **Vetorização**: conversão de palavras em representações numéricas interpretáveis por algoritmos. Modelos de representação distribuída, como *Word2Vec* (Mikolov et al., 2013), *GloVe*

(Pennington et al., 2014) e *FastText* (Bojanowski et al., 2017), foram amplamente utilizados por sua capacidade de capturar relações semânticas entre termos;

5. **Treinamento de modelos de ML**, etapa na qual o algoritmo aprende padrões linguísticos e semânticos a partir dos dados vetorizados.

O avanço dessas técnicas foi determinante para o surgimento de abordagens baseadas em DL, que substituíram representações manuais por vetores contextuais aprendidos automaticamente, culminando na criação dos LLMs.

2.3 Grandes Modelos de Linguagem (LLMs)

Com o progresso das DNNs, especialmente após a introdução da arquitetura *Transformer* (Vaswani et al., 2023), possibilitou-se treinar LLMs em extensos corpora textuais para prever a próxima palavra em uma sequência. Esses modelos aprenderam representações contextuais ricas, permitindo que generalizassem para diversas tarefas de linguagem, sem a necessidade de ajustes específicos para cada uma (Devlin et al., 2018; Raffel et al., 2020).

Entre os exemplos mais conhecidos está o ChatGPT¹, desenvolvido pela OpenAI, inicialmente baseado no modelo GPT-3 (Brown et al., 2020). Essa aplicação tornou os LLMs amplamente acessíveis, permitindo interações em linguagem natural com qualidade quase humana.

No campo educacional, os LLMs têm se mostrado eficazes em auxiliar alunos e professores em atividades de escrita e revisão textual, fornecendo explicações de erros e sugerindo melhorias (Kasneci et al., 2023) e (Vogel et al., 2025). Considerando essa capacidade de compreensão e geração de linguagem natural, este trabalho emprega um LLM para oferecer feedback interpretável sobre as competências avaliadas no ENEM, explicando a nota atribuída e sugerindo formas de aprimoramento.

2.4 Engenharia de Prompts

Com o surgimento dos LLMs e dos modelos *Generative Pre-trained Transformers* (GPTs), novas estratégias de interação com esses sistemas passaram a ser exploradas. A Engenharia de Prompts (*Prompt Engineering*) é o processo de construção e refinamento de instruções textuais, denominadas *prompts*, que orientam o comportamento do modelo durante a geração de respostas (Liu et al., 2023). Essa prática visa maximizar a qualidade das saídas do modelo sem necessidade de re-treinamento.

Entre as técnicas mais relevantes estão o *zero-shot learning* e o *few-shot learning*, que permitem ao modelo resolver tarefas sem ajustes finos (*fine-tuning*) tradicionais (Brown et al., 2020; Radford et al., 2019). No *zero-shot learning*, o modelo utiliza seu conhecimento prévio para resolver tarefas não vistas durante o treinamento, como classificar sentimentos ou responder perguntas. A técnica de *few-shot learning* envolve o fornecimento de poucos exemplos representativos no próprio *prompt*, permitindo que o modelo aprenda o padrão desejado de saída a partir desses exemplos.

Essas abordagens demonstram a flexibilidade dos LLMs, viabilizando sua aplicação em múltiplos contextos a partir de simples instruções textuais. Uma prática comum na criação de

¹<https://chat.openai.com/>

sistemas conversacionais é a definição de uma *persona*, ou seja, um conjunto de instruções que molda o estilo, o tom e a personalidade do modelo durante a interação com o usuário, tornando o diálogo mais natural e direcionado.

3 Trabalhos Relacionados

As abordagens de ML têm se consolidado como solução para diversos problemas de pesquisa, sobretudo no cenário educacional brasileiro. Nesse contexto, esta seção apresenta artigos relacionados que também utilizaram técnicas de NLP para automatizar a avaliação de redações do ENEM.

Em Silva Filho et al., 2023 os autores realizaram um ensaio acerca da utilização de algoritmos clássicos de ML para a geração de *Learning Analytics* de textos narrativos, realizando comparações de menos de 10 modelos clássicos, além de descreverem métricas e métodos para extração de características suplementares às tarefas de ML. Dentre as limitações deste artigo têm-se a não testagem de qualquer modelo baseado em DNNs, tendo explorado somente modelos clássicos sem a utilização de representações neurais ou modelos de linguagem neural para a representação das palavras.

Por outro lado, Oliveira et al., 2022 apresentam questionamentos e possibilidades para o emprego de algoritmos de aprendizado para a correção de redações em exames de admissão, como o ENEM, além de apresentarem um impasse para a melhora do desempenho de modelos de aprendizado nesse sentido que consiste no desbalanceamento da base de dados, além da pouca disponibilidade de dados para um melhor refinamento desses modelos, simbolizando um problema principalmente para modelos neurais. Entretanto, os autores analisaram somente a pontuação associada a uma das cinco competências avaliadas pelo ENEM, não avaliando as correlações entre o modelo e as demais quatro notas que compõem a nota final da redação. Além disso, apesar do uso de uma ANN, não foi explorada a aplicação de modelos do tipo *transformer* para a extração de representações lexicais (*embeddings* derivados de *transformers*), que poderiam fornecer representações semânticas mais ricas e potencialmente melhorar o desempenho na competência 4 (coesão textual).

Além disso, Rosa e Mello, 2022 propõem a utilização do gênero textual para a tarefa de classificação, o que pode ser um indicativo de boa coesão textual. Ademais, os mesmos evidenciam ainda que a utilização de ML para a correção e análise de coesão textual e de demais características textuais já foram empregadas em outras línguas, afirmando ainda que no caso particular da língua portuguesa essa área ainda estaria dando seus passos iniciais.

Por fim, Dai et al., 2023 conduzem uma investigação sobre a utilização da plataforma ChatGPT para produção de feedbacks educacionais, com três objetivos principais: (i) avaliar se o modelo gera feedbacks mais claros e úteis para a aprendizagem dos estudantes; (ii) verificar em que medida o uso do modelo pode reduzir a sobrecarga de avaliadores humanos; e (iii) analisar a capacidade do modelo de indicar, no próprio feedback, direções concretas do que os alunos devem aprender. O trabalho demonstrou que modelos como o ChatGPT são capazes de proporcionar feedbacks eficientes e em boa concordância quando comparado com avaliadores humanos, além

dos feedbacks serem mais detalhados e serem plenamente capazes de guiar o aprendizado dos estudantes.

Em síntese, a revisão da literatura indica que, embora alguns trabalhos tenham utilizado modelos neurais, estes apoiaram-se sobretudo em técnicas tradicionais de vetorização. Abordagens mais sofisticadas (por exemplo, empregar modelos pré-treinados apenas para extração de características e geração de *embeddings*) foram pouco exploradas. Do mesmo modo, arquiteturas mais profundas ou alternativas (p. ex. CNNs e RNNs) não foram avaliadas quanto ao seu desempenho na tarefa de regressão. O mesmo cenário ocorreu com Aprendizagem de Múltiplas Tarefas (Em inglês, *Multitask Learning* - MTL), que não chegou a ser testada, uma vez que os trabalhos focaram em avaliar a geração de novas características ou focaram seus esforços na avaliação de somente uma das 5 competências presentes na nota final. Além disso, não empregaram a utilização de um modelo de linguagem menor em uma aplicação conjunta com modelos de correção para um feedback personalizado e detalhado sobre o texto desenvolvido pelo aluno.

O presente artigo contribui para a literatura ao se utilizar não somente de um modelo pré-treinado para extração dos vetores de palavras, mas também na avaliação de diferentes modelos, utilizando os diferentes tipos de ANN, que podem realizar a correção das 5 competências de um único texto, representando um passo importante uma vez que os modelos anteriores não realizavam a correção das 5 competências com um único texto.

Além disso, o presente artigo também contribui para a literatura ao apresentar uma aplicação que recorre ao modelo de correção de redações juntamente com um modelo generativo de pequena escala para a correção das notas da redação do aluno e uma apresentação posterior de um feedback descritivo acerca da nota do aluno e o que ele pode fazer para obter notas melhores ou somente melhorar a qualidade de seu texto.

4 Método

Esta seção apresenta o método do presente artigo, descrevendo a base de dados utilizada, bem como o processo de pré-processamento dos dados, seleção e avaliação dos modelos.

4.1 Base de dados

A base de dados empregada no uso deste estudo foi a base de dados Extended Essay-Br, descrita em (Marinho et al., 2021), contendo até a data deste estudo um total de 6577 redações corrigidas por profissionais humanos conforme as competências do ENEM. Essa base de dados dispõe das notas de cada aluno em cada uma das 5 competências, além de dispor da nota final total do mesmo, conforme detalhado na Tabela 1.

Tabela 1: Estatísticas das notas da base.

#	Competência	Média	Desvio Padrão
1	norma-padrão da língua portuguesa	133,25	33,17
2	gênero, tema e repertório	132,75	39,59
3	autoria	115,56	38,17
4	coesão textual	134,99	45,38
5	proposta de intervenção	115,71	51,79
Nota Total		632,25	179,81

Quanto às competências do ENEM, as mesmas consistem em:

- Competência 1: Demonstrar domínio da modalidade escrita forma da língua portuguesa;
- Competência 2: Compreender a proposta de redação e aplicar conceitos das várias áreas de conhecimento para desenvolver o tema, nos limites estruturais do texto dissertativo-argumentativo em prosa;
- Competência 3: Selecionar, relacionar, organizar e interpretar informações, fatos, opiniões e argumentos em defesa de um ponto de vista;
- Competência 4: Demonstra conhecimento dos mecanismos linguísticos necessários para a construção da argumentação;
- Competência 5: Elaborar proposta de intervenção para o problema abordado que respeite os direitos humanos.

4.2 Preparação dos Dados

Para atingir o objetivo deste artigo, foi recorrido ao teste e avaliação de diferentes configurações de modelos neurais. A primeira etapa para o treinamento dos modelos foi a execução de processos de limpeza no *corpus*; dessa forma foram aplicadas as seguintes etapas de pré-processamento:

1. Remoção de *stopwords*.
2. Remoção de pontuação e acentuação.
3. Normalização dos textos por meio de lematização.

Tais técnicas foram empregadas com o intuito de tornar o texto presente na base o mais limpo possível, proporcionando aos modelos neurais uma melhor extração do conhecimento presente e melhor desempenho final na tarefa de regressão. Por exemplo, a remoção de *stopwords*, pontuação e acentuação foi empregada com o intuito de reduzir a quantidade de palavras que não apresentam tanta importância semântica, uma vez que o modelo utilizado para a extração de características é baseado em atenção, o BERTimbau, descrito em (Souza et al., 2020), e as muitas ocorrências dessas palavras podem fazer com que o algoritmo atribua maior atenção às mesmas.

Quanto à utilização da lematização, procedeu-se a sua aplicação para padronizar formas flexionadas e reduzir ruído morfológico, de modo a melhorar a extração de características pelo BERTimbau e favorecer uma representação semântica mais estável. O emprego de tais técnicas é notado também em outros trabalhos como em (Silva Filho et al., 2023), (A. Barbosa & Campelo, 2020) e (Oliveira et al., 2022), sendo utilizadas para a contabilização de ocorrências de palavras, contabilização de características e a utilização das formas base (lemmas) para a análise de melhoria de desempenho em tarefas de correção de textos.

Para a extração das características foi utilizado um modelo pré-treinado BERT adaptado para a língua portuguesa, o BERTimbau, realizando a extração de vetores de palavras (*word embeddings*) para alimentar as camadas posteriores nas topologias de ANN avaliadas. A utilização do BERTimbau foi feita para não ser necessário o retreinamento dos modelos a serem testados na tarefa de representação e modelagem da língua portuguesa, proporcionando um maior foco no treinamento e ajustes de hiperparâmetros dos regressores.

4.3 Seleção de Modelos de correção

A seleção dos modelos de correção se deu através do treinamento dos modelos na base de dados anteriormente citada com uma divisão de 75/10/15 em treino, validação e teste, realizada após a limpeza dos textos da base. Para definição do melhor modelo, foram realizados testes usando camadas LSTM, GRU, Convoluções 1D, camadas *feed-forward* e camadas LSTM e GRU bidirecionais.

Também foram realizadas alterações na forma que se era realizada a tokenização e geração da sequência de palavras para vetorização pelo BERTimbau, sendo testada inicialmente uma sequência de tamanho de 64 *tokens* e, posteriormente, uma sequência de 128 *tokens*, utilizando como base as métricas do dataset disponibilizadas em (Marinho et al., 2021), demonstrando as métricas de média de *tokens* por texto, *tokens* por parágrafo e *tokens* em cada sentença, além de realizar essa variação para verificar se a utilização de sequências de entrada maiores poderia proporcionar uma melhora no desempenho final dos modelos testados.

Ademais, também foi utilizada a biblioteca Optuna² para realizar ajustes de hiperparâmetros nos modelos e selecionar aqueles que apresentaram o melhor desempenho. O Optuna é um *framework* de otimização de hiperparâmetros, apresentado e descrito em (Akiba et al., 2019), que possui implementações eficientes para diversos tipos de métodos de otimização de hiperparâmetros, como *Grid Search*, *Random Search* e Otimização Bayesiana, sendo o método de otimização utilizado no presente trabalho o *Tree-Structured Parzen Estimator* (Watanabe, 2023).

Devido à natureza computacionalmente intensiva das ANNs, a primeira etapa do experimento foi verificar a integração de um modelo pré-treinado com as camadas subsequentes para não ser necessário recorrer ao treinamento de um modelo que tivesse que aprender o vocabulário antes de poder ser utilizado para testagem.

Em seguida foram realizados alguns testes iniciais com a utilização de um modelo com 5 saídas de regressão, para verificar a viabilidade do aprendizado e o desempenho do modelo num momento inicial e, por se tratar de um modelo de DL com 5 saídas de regressão independentes, fez-se necessário o uso do Optuna para a otimização dos hiperparâmetros da rede, na tentativa

²<https://optuna.readthedocs.io/en/stable/>

de se obter uma melhor configuração do modelo e poupar tempo de treinamento, uma vez que a otimização desses parâmetros pode tornar-se exaustiva.

Dentre os parâmetros utilizados como espaço de busca, foram utilizadas os seguintes valores:

1. Taxa de *dropout*: variando de 0,1 até 0,5
2. Regularização L2: variando de $1e-10$ até $1e-1$
3. Tipos de camada: *feed-forward*, convolução 1D, LSTM, GRU e camadas LSTM ou GRU bidirecionais.
4. Número de camadas: variando de 1 até 3
5. Número de filtros convolucionais: variando de 16 até 64
6. Número de células recorrentes: de 8 até 32
7. Otimizadores: Adam, RMSProp e SGD
8. *Learning rate*: variando de $1e-3$ até $1e-1$
9. Momentum: variando de $1e-3$ até $1e-1$

Além disso, também foi configurada uma função de *callback* personalizada para realizar o *EarlyStopping*, uma vez que o *Early Stop* padrão do Keras só consegue atentar-se a uma métrica por vez; nesta função personalizada o *EarlyStopping* era realizado baseando-se na mediana dos 5 *Root Mean Squared Error* (RMSEs) da rede no conjunto de validação, devido ao desbalanceamento da base nas notas das competências fornecidas, juntamente com o *Mean Absolute Error* (MAE) como função de perda. Juntamente com a função de *EarlyStopping* também foi definido o valor de *patience* = 3, pois em testes iniciais do experimento observou-se que com o valor utilizado neste parâmetro sendo 7, alguns modelos atingiam uma etapa de treinamento sem nenhuma alteração significativa nas métricas avaliadas.

4.4 Seleção dos modelos de feedback

A seleção dos modelos de feedback foi feita empiricamente, através do teste por meio de engenharia de prompt com a aplicação de *zero-shot learning*, utilizando apenas um prompt com a descrição da persona, tarefa e critérios para a realização da tarefa, sendo a mesma descrita como "proporcionar ao usuário um feedback conciso acerca de sua redação".

Os modelos escolhidos para a realização dos testes por meio de *zero-shot* foram os modelos Llama3.1-8B e Gemma3-4B; a escolha desses modelos ocorreu devido ao tamanho e à possibilidade de execução sem a necessidade de um ambiente em nuvem, permitindo uma avaliação em hardware local e proporcionando a possibilidade de avaliar esses modelos sem transferência de dados, aspecto relevante caso avaliações futuras envolvam bases com informações sensíveis (neste estudo, contudo, foram empregados apenas dados públicos). Além disso, essa configuração facilita a análise da qualidade de modelos de menor porte destinados a execução em hardware de borda (edge).

Dentre esses modelos, foram realizados testes de geração de feedback com redações do *dataset* Extended Essay Br, descrito em (Marinho et al., 2021), utilizado entre dois modelos, sendo estes o Gemma3-4B e o Llama3.1-8B.

As instruções utilizadas nos prompts para realizar o teste dos modelos foram:

Instruções para o LLM

Persona: Você é Clarice, uma auxiliar de correção de redações amigável e atenciosa.

Tarefa: Você deve fornecer um feedback conciso acerca da redação do usuário, proporcionando uma boa revisão acerca de quais pontos o usuário deve melhorar para obter melhores notas em sua redação.

Critérios de avaliação: As notas da redação foram avaliadas em cinco competências, cada uma valendo 200 pontos. As competências são:

1. Demonstrar domínio da norma padrão da língua escrita;
2. Compreender a proposta da redação e aplicar conceitos das várias áreas de conhecimento para desenvolver o tema;
3. Selecionar, relacionar, organizar e interpretar informações, fatos, opiniões e argumentos em defesa de um ponto de vista;
4. Demonstrar conhecimento dos mecanismos linguísticos necessários para a construção da argumentação;
5. Elaborar proposta de intervenção para o problema abordado, respeitando os direitos humanos.

Ao final do feedback, informe ao usuário quais foram as notas dele em cada uma das competências bem como sua nota total.

4.5 Desenvolvimento de uma aplicação

Devido à necessidade de se testar a viabilidade técnica da proposta de geração de feedback a partir da utilização também das suas notas previstas pelo modelo de correção, fez-se necessário o desenvolvimento de uma aplicação que possibilitasse os testes de serem realizados de maneira eficiente, sendo a mesma responsável por:

1. Disponibilizar o modelo de correções em um endpoint web, capaz de ser facilmente utilizado para a correção das redações e obtenção das notas do usuário.
2. Prover uma infraestrutura para a comunicação entre o modelo de correção e o modelo de feedbacks.

3. Prover uma interface de fácil utilização para o envio da redação a ser corrigida bem como a exibição do feedback e nota gerados, além de proporcionar uma interface de fácil acesso e comunicação com o modelo de feedback numa estrutura de chat.

Para a utilização de modelos de linguagem de menor escala em ambientes locais, sem a dependência de ambientes de nuvem, foi utilizado o *framework* de código aberto Ollama³, o qual oferece uma diversa biblioteca de modelos de código aberto facilmente utilizáveis, proporcionando um avanço na democratização do uso da inteligência artificial generativa (em inglês, *Generative Artificial Intelligence- Gen AI*).

Para o desenvolvimento dessa aplicação, recorreu-se às bibliotecas e *frameworks* Streamlit⁴ para a criação da interface gráfica de conversação com o modelo, LangChain⁵ para a criação da estrutura/backend conversacional com o modelo, bem como definição de cadeias de conversação e gerenciamento da interação com o modelo e o já citado Ollama para disponibilizar os modelos para uso no ambiente local por meio de requisições HTTP. Também foram configurados prompts proporcionando as instruções a serem seguidas pelos dois modelos durante a rodada de testes, sendo realizados ajustes finos nos prompts para os mesmos extraírem o máximo de desempenho dos modelos testados.

Além disso, também foi utilizado o Docker como serviço de containerização para que a utilização desta aplicação fosse simples e disponível para qualquer um interessado na sua utilização, bem como a disponibilização do melhor modelo de correção em um repositório no ambiente da Hugging Face⁶.

5 Resultados e Discussão

5.1 Modelos de correção

Ao longo do trabalho desenvolvido, foram executados, ao todo, testes em 40 modelos diferentes, utilizando-se o espaço de busca citado na seção 4.3. Ou seja, os 40 modelos são definidos a partir das combinações entre camadas (e.g., LSTM, GRU e Convolução 1D) e hiperparâmetros (e.g., taxa de *dropout*, nível de regularização L2 e número de camadas) considerados em nossos experimentos, sendo os 5 melhores modelos os seguintes:

- Convolução 1D
- LSTM Bidirecional
- 2 modelos GRU Bidirecional
- *Feed-Forward*

³<https://ollama.com/>

⁴<https://streamlit.io/>

⁵<https://www.langchain.com/>

⁶<https://huggingface.co/>

Em termos de topologia, todos os modelos adotam uma arquitetura base comum composta por: (i) uma camada de entrada; (ii) um modelo BERT pré-treinado utilizado para a geração de representações vetoriais das palavras; e (iii) uma camada de *dropout* seguida por cinco regressores independentes, cada um responsável por prever a nota associada a uma competência específica. Para cada regressor, o número de camadas e o tipo de camada foram determinados de forma aleatória durante a execução do experimento, resultando em arquiteturas heterogêneas entre os regressores. Ressalta-se que a escolha do tipo de camada segue uma consistência estrutural: caso o modelo selecionado seja baseado em convolução unidimensional, todos os cinco regressores são convolucionais; se for LSTM, todos os regressores utilizam camadas LSTM, e assim sucessivamente.

Para fins de completude, a tabela com os resultados de todos os 40 modelos e o código fonte da aplicação podem ser encontrados em nosso material suplementar⁷. Adicionalmente o código fonte da aplicação também pode ser encontrado no repositório do github⁸.

Dentre os modelos acima, com exceção do modelo convolucional, os demais apresentaram métricas medianas das 5 competências para o RMSE em torno de 35,4 a 35,91, enquanto os 5 modelos que obtiveram os piores valores nessa métrica tiveram valores observados em torno de 107,03 a 126,44, sendo o modelo com pior desempenho também um modelo convolucional.

Apesar de se obter aumento no tempo de treino com o aumento do tamanho da sequência, foi perceptível que esse aumento proporcionou um benefício significativo no desempenho final do modelo, ficando evidente nos valores do MAE, que apresentaram uma convergência melhor na segunda rodada de otimização através do Optuna, conseguindo atingir os valores verificados no modelo convolucional. A Tabela 2 apresenta os valores atingidos para o MAE em cada competência.

Tabela 2: MAE de cada competência nos modelos.

#	Modelo	Comp. 1	Comp. 2	Comp. 3	Comp. 4	Comp. 5
1	Convolução 1D	15,24	16,34	17,32	19,8	21,48
2	LSTM Bidirecional	14,69	22,51	26,24	27,84	33,28
3	GRU Bidirecional	23,05	24,81	25,57	25,54	33,39
4	GRU Bidirecional 2	18,72	24,17	24,36	27,39	30,74
5	Feed-Forward	21,29	26,24	26,69	28,87	32,88

Apesar de o modelo convolucional ter se sobressaído quanto aos demais modelos, é importante observar que todos os 5 modelos atingiram métricas relativamente próximas em cada uma das competências analisadas, obtendo-se uma maior discrepância na competência 5, onde o modelo convolucional conseguiu obter resultados com pouco mais de 10 pontos de diferença em relação ao segundo melhor modelo avaliado. A Tabela 3 apresenta os valores atingidos para o RMSE em cada competência.

O resultado apresentado pelo modelo convolucional mais uma vez foi superior em relação aos demais, conseguindo manter quase todas as suas métricas em torno de 20, o que implica que modelos convolucionais utilizados em conjunto com *transformers* para a extração de características

⁷https://osf.io/kqdbc/overview?view_only=005598998a074f31bda022102a913e66

⁸<https://github.com/JT4v4res/CLaRiCe>

Tabela 3: RMSE de cada competência nos modelos.

#	Modelo	Comp. 1	Comp. 2	Comp. 3	Comp. 4	Comp. 5
1	Convolução 1D	22,71	23,89	24,38	27,56	31,03
2	LSTM Bidirecional	27,88	33,92	35,40	38,45	46,06
3	GRU Bidirecional	31,28	34,90	37,71	35,58	43,60
4	GRU Bidirecional 2	29,97	33,36	35,82	38,58	41,14
5	Feed-Forward	29,48	35,31	35,92	38,61	43,77

possam ser uma abordagem ideal para a representação das notas em contextos textuais, dado que as notas no contexto do ENEM para cada competência da redação tendem a ser incrementadas ou decrescidas de 20 em 20 pontos, enquanto nosso modelo alcançou, em média, um erro de 25,91 no RMSE e conseguiu atingir um erro médio de 18,04 no MAE.

Ao comparar os resultados referentes à competência 4 obtidos pelo modelo com melhor desempenho neste estudo com aqueles do algoritmo *Extremely Randomized Trees*, descrito em (Batista et al., 2022) como o que apresentou as melhores métricas, verificou-se que o modelo convolucional unidimensional alcançou desempenho superior. Especificamente, o modelo proposto obteve um *RMSE* de 27,56, em contraste com 35,94 reportado por (Batista et al., 2022), e um *MAE* de 19,80, em comparação a 26,97 observados no estudo de referência.

A Figura 1 apresenta os gráficos de convergência do MAE dos modelos de convolução unidimensional e do modelo LSTM Bidirecional.

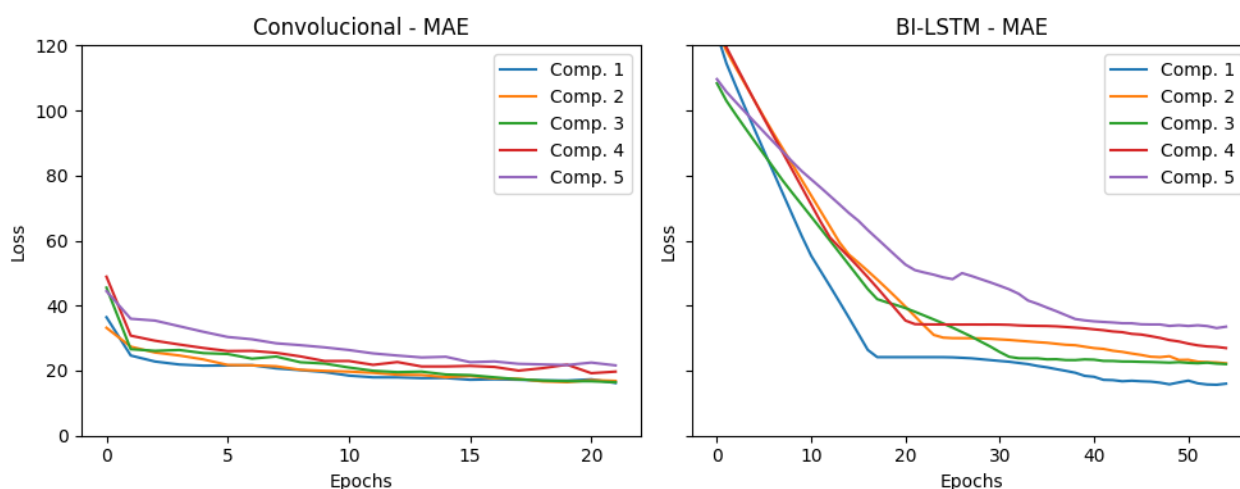


Figura 1: Gráfico de convergência do modelo convolucional.

Os dois gráficos apresentaram uma disparidade na curva de aprendizado. Enquanto o gráfico do modelo convolucional, apesar de não apresentar flutuações tão abruptas na curva de perda, conseguiu obter resultados mais satisfatórios, o modelo recorrente apresenta um gráfico típico que pode simbolizar uma boa convergência do modelo. No entanto, ele não foi capaz de obter valores menores para a métrica utilizada como perda; uma suposição plausível é de que, com alguns ajustes

nos hiperparâmetros, o modelo recorrente possa atingir resultados ainda mais satisfatórios que o modelo convolucional.

Por fim, as Tabelas 4 e 5 apresentam respectivamente os hiperparâmetros obtidos pelas duas melhores configurações: convolucional e recorrente.

Tabela 4: Hiperparâmetros dos regressores do modelo convolucional.

Parâmetro	Comp. 1	Comp. 2	Comp. 3	Comp. 4	Comp. 5
Total de camadas	7	5	5	7	5
Filtros/convolução	28/39	22	53	51/21	28
Neurônios na camada densa	90	20	126	39	123
Otimizador	Adam				
<i>Learning Rate</i>	7,06e-03				
<i>Dropout rate</i>	3,03e-01				
<i>Weight Decay</i> (L2)	6,61e-04				

Tabela 5: Hiperparâmetros dos regressores do modelo recorrente.

Parâmetro	Comp. 1	Comp. 2	Comp. 3	Comp. 4	Comp. 5
Total de camadas	3	2	2	1	2
Unidades/Célula	10/26/21	26/16	22/13	18	29/11
Otimizador	Adam				
<i>Learning Rate</i>	1,01e-03				
<i>Dropout rate</i>	1,64e-01				
<i>Weight Decay</i> (L2)	4,67e-06				

Apesar de serem modelos que possuem abordagens diferentes, a diferença se dá em duas principais características:

1. O modelo recorrente não apresenta uma camada densa nem camada de achatamento (em inglês, *flatten*) antes de enviar os resultados de suas iterações para a camada de saída do regressor.
2. O número total de camadas do modelo convolucional foi maior, sobretudo nas competências 1 e 4, o que pode significar que um modelo recorrente que possua uma abordagem com maior profundidade na rede, além de acrescentar uma última camada densa, possa beneficiar os resultados das notas de cada competência.

Também é importante destacar que em todas as camadas foi aplicada a regularização L2, além de possuírem camadas de *dropout* entre cada camada da rede para melhorar o desempenho de generalização das camadas intermediárias da rede.

5.2 Modelos de feedback

Nesta etapa, foram avaliados dois modelos de linguagem de grande porte (LLMs) para a geração automática de feedback textual, Gemma3-4B e Llama3.1-8B, em termos de viabilidade técnica. Ambos demonstraram desempenho satisfatório na tarefa proposta, evidenciando potencial de aplicação na oferta de feedback personalizado para textos dissertativo-argumentativos no contexto do Exame Nacional do Ensino Médio (ENEM).

O Gemma3-4B apresentou respostas mais estáveis e consistentes, com baixa ocorrência de alucinações, entendidas como trechos desconectados do contexto ou das instruções fornecidas. Observou-se também o uso espontâneo de elementos comunicativos, como emojis, que contribuíram para um tom que pode ser compreendido como mais humanizado na interação. Embora esse comportamento não tenha sido um objetivo explícito da modelagem, pode representar um aspecto relevante em cenários educacionais, nos quais o engajamento e a empatia são fatores importantes. As Figuras 2 e 3 apresentam um exemplo de avaliação a partir das notas e sugestões de melhoria no texto geradas pelo modelo Gemma3-4B.

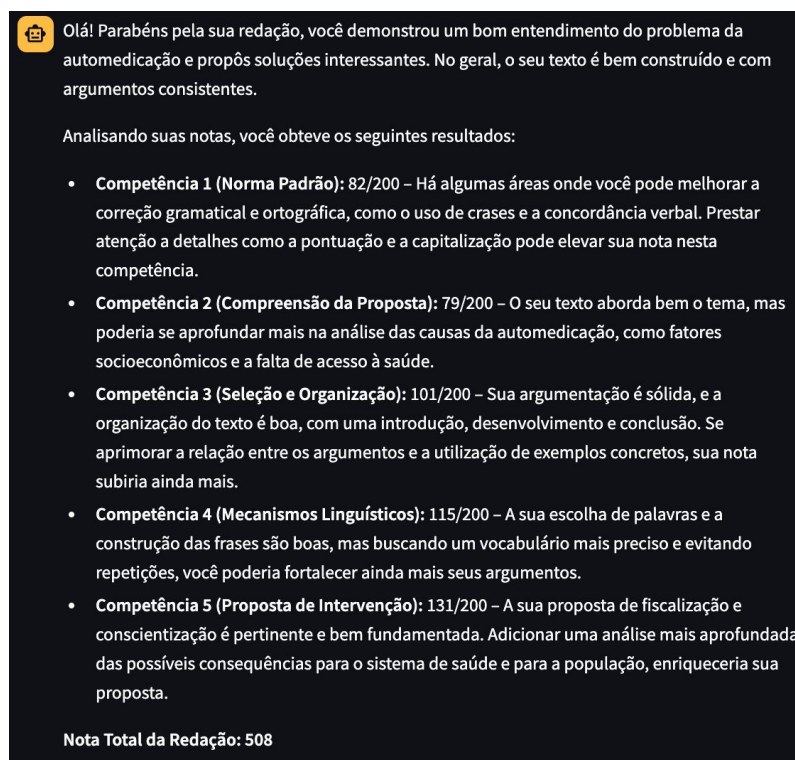


Figura 2: Exemplo de avaliação gerado pelo modelo Gemma3-4B..

O Llama3.1-8B, por sua vez, gerou respostas mais extensas e detalhadas, mantendo coerência com o contexto proposto. Essa característica pode favorecer a explicitação de aspectos específicos do texto do estudante, embora exija maior controle para evitar redundâncias ou desvios das instruções. Diferentemente do Gemma3-4B, o Llama3.1-8B não produziu elementos comunicativos não verbais (como emojis), o que pode estar relacionado às diferenças em sua arquitetura, tokenização ou conjunto de treinamento. As Figuras 4 e 5 apresentam um exemplo de avaliação a partir das notas e sugestões de melhoria no texto geradas pelo modelo Llama3.1-8B.

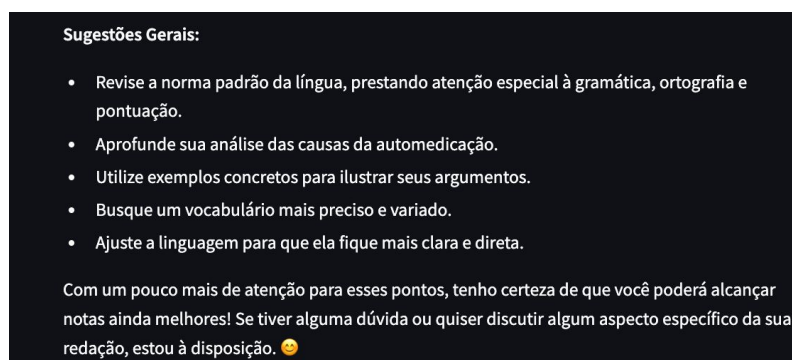


Figura 3: Exemplo de sugestão de melhoria gerado pelo modelo Gemma3-4B..

No que diz respeito à eficiência computacional, o Gemma3-4B apresentou menor demanda de armazenamento (aproximadamente 3,3 GB) e consumo reduzido de recursos durante a inferência, enquanto o Llama3.1-8B exigiu cerca de 4,9 GB. Essa diferença pode ser um fator de decisão em contextos com restrições de infraestrutura, como sistemas de baixo custo ou aplicações embarcadas.

De forma geral, os resultados obtidos indicam que ambos os modelos são capazes de produzir feedbacks coerentes e úteis, cada um com características distintas quanto à extensão, expressividade e eficiência. Assim, a escolha do modelo mais adequado dependerá do perfil de uso e das prioridades do sistema, sejam elas relacionadas à profundidade do feedback, à naturalidade da comunicação ou às limitações computacionais.

5.3 Ameaças à validade

Apesar dos resultados promissores apresentados pelos modelos avaliados, algumas ameaças à validade devem ser consideradas. Em termos de *validade interna*, a heterogeneidade das arquiteturas dos regressores e a randomização dos hiperparâmetros podem ter introduzido variabilidade nos resultados. Quanto à *validade externa*, os achados são baseados exclusivamente na base Extended Essay-Br, o que limita a generalização dos modelos para outros contextos ou tipos de redações.

Em relação à *validade de conclusão*, embora métricas como RMSE e MAE indiquem diferenças de desempenho, inferências sobre a superioridade de um modelo sobre outro devem ser interpretadas com cautela, considerando a amostra limitada de textos avaliados. Além disso, a avaliação dos feedbacks foi realizada em termos de viabilidade técnica, sendo limitada pela falta de uma avaliação externa em termos de fatores como efetividade pedagógica. Por fim, a *validade de constructo* pode ter sido impactada pela escolha das métricas e pela tokenização das sequências, que podem não capturar integralmente todas as nuances linguísticas das redações.

6 Considerações Finais e Trabalhos Futuros

Este trabalho investigou o uso de DNNs como uma abordagem alternativa para a correção e geração de feedback automático de redações do Exame Nacional do Ensino Médio (ENEM). Os resultados obtidos demonstram que modelos baseados em *embeddings* e arquiteturas neurais podem prever,

 Olá! Que bom que você buscou feedback sobre sua redação. No geral, o seu texto apresenta uma boa argumentação sobre a automedicação, e suas notas refletem isso. Abaixo, detalho o meu feedback para que você possa aprimorá-la ainda mais:

Análise da Redação:

- **Competência 1: Domínio da Norma (82/200):** Há algumas pequenas questões de pontuação e concordância verbal que podem ser melhoradas, como no uso de vírgulas e na forma como alguns termos são empregados. Concentre-se em revisar a pontuação e garantir a concordância verbal em todos os trechos.
- **Competência 2: Compreensão da Proposta (79/200):** A sua redação demonstra um bom entendimento do tema e apresenta argumentos relevantes. No entanto, poderiam ser aprofundados alguns conceitos relacionados à saúde pública e ao acesso a medicamentos.
- **Competência 3: Seleção e Organização (101/200):** A organização do texto é clara e a argumentação é bem construída. No entanto, a conexão entre os parágrafos poderia ser mais explícita, facilitando a compreensão do fluxo de ideias.
- **Competência 4: Mecanismos Linguísticos (115/200):** O seu uso da linguagem é correto e adequado. No entanto, a aprofundação em alguns mecanismos linguísticos, como a utilização de conectivos e a variedade de vocabulário, poderia enriquecer ainda mais a sua redação.
- **Competência 5: Proposta de Intervenção (131/200):** A sua proposta de fiscalização e orientação é pertinente e bem-intencionada. No entanto, seria interessante explorar outras soluções, como programas de educação em saúde e o fortalecimento do SUS.

Nota Total: 508

Figura 4: Exemplo de avaliação gerado pelo modelo Llama3.1-8B..

Sugestões Gerais:

- **Revisão da Normatização:** Revise cuidadosamente a pontuação e a concordância verbal.
- **Aprofundamento:** Explore conceitos da saúde pública e do acesso a medicamentos.
- **Conexões:** Fortaleça as conexões entre os parágrafos para garantir uma leitura mais fluida.
- **Variedade:** Utilize um vocabulário mais variado e explore o uso de conectivos para ligar ideias de forma mais eficaz.

Espero que este feedback seja útil para você aprimorar suas habilidades de escrita! Se tiver mais alguma dúvida, não hesite em perguntar.

Figura 5: Exemplo de sugestão de melhoria gerado pelo modelo Llama3-4B..

com erro reduzido, as pontuações atribuídas a cada uma das cinco competências avaliadas. A utilização de representações semânticas extraídas por modelos *transformers*, em contraste com métodos mais tradicionais, como o TF-IDF, revelou-se uma estratégia eficaz para capturar as nuances do texto fortemente associadas à avaliação humana.

Além disso, foi proposta e avaliada uma segunda etapa complementar, voltada à geração automática de feedback textual. Essa etapa, baseada em LLMs, demonstra o potencial de integração entre modelos de predição e sistemas interativos de apoio à aprendizagem. Essa proposta tem como finalidade ampliar o impacto pedagógico da ferramenta, oferecendo ao estudante orientações mais detalhadas sobre seus pontos fortes e áreas que demandam melhoria, com potencial de aplicação em ambientes de autoestudo ou plataformas educacionais.

Apesar dos avanços demonstrados, algumas limitações precisam ser consideradas. Em primeiro lugar, o conjunto de dados utilizado, embora amplamente adotado na literatura, possui

volume restrito (6577 redações) limitando o refinamento dos modelos, pois modelos de DL, em geral, demandam grandes volumes de dados para atingir pleno desempenho. Em segundo lugar, a implantação prática de ferramentas baseadas em DNNs exige infraestrutura computacional robusta, dificultando sua adoção em contextos com conectividade limitada ou recursos computacionais escassos, como ocorre em muitas regiões de países como o Brasil. Além disso, o uso de LLMs para geração de feedback também suscita preocupações legítimas sobre privacidade e confidencialidade dos dados, especialmente em contextos formais como o ENEM.

Diante disso, propõem-se como direções futuras:

1. Ampliação e diversificação de corpora textuais: desenvolvimento e curadoria de bases de dados mais amplas e variadas, contemplando diferentes gêneros textuais e estilos de escrita, com vistas a expandir a aplicabilidade da ferramenta para além do modelo dissertativo-argumentativo.
2. Exploração de métodos de aprendizado por ranqueamento: reformulação da tarefa de regressão como um problema de ordenação ou classificação, a fim de investigar se abordagens neurais oferecem vantagens adicionais em contextos de avaliação relativa.
3. Desenvolvimento de modelos leves para ambientes com baixa conectividade: pesquisa voltada à compressão de modelos, inferência embarcada e uso de agentes inteligentes, visando viabilizar o uso da tecnologia em cenários de infraestrutura limitada e promover maior equidade no acesso à educação assistida por AI.

Com base nos achados apresentados, conclui-se que o uso combinado de modelos de correção baseados em DL e ferramentas interativas de feedback automatizado apresenta um potencial relevante na interseção entre tecnologias educacionais e AI, com implicações práticas para a melhoria do processo de ensino-aprendizagem em larga escala.

Artigo Premiado Estendido

Esta publicação é uma versão estendida do melhor artigo da Trilha 4 – Inteligência Artificial na Educação, do XXXV Simpósio Brasileiro de Informática na Educação (SBIE 2024), intitulado “CLaRiCe: Uma abordagem neural para a correção automática de redações”. DOI: <https://doi.org/10.5753/sbie.2024.242576>

Referências

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A Next-generation Hyperparameter Optimization Framework. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. <https://doi.org/10.48550/arXiv.1907.10902> [GS Search].

- Barbosa, A., & Campelo, C. (2020). Processamento de Linguagem Natural em Artefatos Textuais Educacionais: Um Mapeamento Sistemático no Contexto Brasileiro. *Anais do XXXI Simpósio Brasileiro de Informática na Educação*, 1433–1442. <https://doi.org/10.5753/cbie.sbie.2020.1433> [GS Search].
- Barbosa, G., Batista, H., Miranda, P., Santos, J., Isotani, S., Cordeiro, T., Bittencourt, I. I., & Mello, R. F. (2022). Aprendizagem de Máquina para Classificação de Tipos Textuais: Estudo de Caso em Textos escritos em Português Brasileiro. *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*, 920–931. <https://doi.org/10.5753/sbie.2022.224769> [GS Search].
- Batista, H., Barbosa, G., Miranda, P., Santos, J., Isotani, S., Cordeiro, T., Bittencourt, I. I., & Mello, R. F. (2022). Detecção Automática de Clímax em Produções de Textos Narrativos. *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*, 932–943. <https://doi.org/10.5753/sbie.2022.224770> [GS Search].
- Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2), 157–166. <https://doi.org/10.1109/72.279181> [GS Search].
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5, 135–146. <https://doi.org/10.48550/arXiv.1607.04606> [GS Search].
- Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., ... Amodei, D. (2020). Language Models are Few-Shot Learners. <https://doi.org/10.48550/arXiv.2005.14165> [GS Search].
- Cambria, E., & White, B. (2014). Jumping NLP Curves: A Review of Natural Language Processing Research. *IEEE Computational Intelligence Magazine*, 9(2), 48–57. <https://doi.org/10.1109/MCI.2014.2307227> [GS Search].
- Chai, C. P. (2023). Comparison of text preprocessing methods. *Natural Language Engineering*, 29(3), 509–553. <https://doi.org/10.1017/S1351324922000213> [GS Search].
- Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder–decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1724–1734. <https://doi.org/10.3115/v1/D14-1179> [GS Search].
- Dai, W., Lin, J., Jin, H., Li, T., Tsai, Y.-S., Gašević, D., & Chen, G. (2023). Can large language models provide feedback to students? A case study on ChatGPT. *2023 IEEE international conference on advanced learning technologies (ICALT)*, 323–325. <https://doi.org/10.1109/ICALT58122.2023.00100> [GS Search].
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*. <https://doi.org/10.48550/arXiv.1810.04805> [GS Search].
- Freitas, E., Falcão, T. P., & Mello, R. F. (2020). Desmistificando a adoção de learning analytics: um guia conciso sobre ferramentas e instrumentos. *Sociedade Brasileira de Computação*. <https://doi.org/10.5753/sbc.5627.6.4> [GS Search].
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735> [GS Search].

- Jurafsky, D., & Martin, J. H. (2026). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, with Language Models* (3rd) [Online manuscript released August 19, 2026]. <https://web.stanford.edu/~jurafsky/slp3/>
- Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Malaka, R., Menzel, S., Nerdel, C., Pfeffer, J., Sailer, M., Schmidt, A., Seidel, T., ... Weller, A. (2023). ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education. *Learning and Individual Differences*, 103, 102274. <https://doi.org/10.1016/j.lindif.2023.102274> [GS Search].
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 521(7553), 436–444. <https://doi.org/10.1038/nature14539> [GS Search].
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324. <https://doi.org/10.1109/5.726791> [GS Search].
- Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, 55(9), 1–35. <https://doi.org/10.1145/3560815> [GS Search].
- Marinho, J., Anchiêta, R., & Moura, R. (2021). Essay-BR: a Brazilian Corpus of Essays. *Anais do III Dataset Showcase Workshop*, 53–64. <https://doi.org/10.5753/dsw.2021.17414> [GS Search].
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *Proceedings of ICLR*. <https://doi.org/10.48550/arXiv.1301.3781> [GS Search].
- Oliveira, H., Miranda, P., Isotani, S., Santos, J., Cordeiro, T., Bittencourt, I. I., & Mello, R. F. (2022). Estimando Coesão Textual em Redações no Contexto do ENEM Utilizando Modelos de Aprendizado de Máquina. *Anais do XXXIII Simpósio Brasileiro de Informática na Educação*, 883–894. <https://doi.org/10.5753/sbie.2022.224736> [GS Search].
- Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. <https://doi.org/10.3115/v1/D14-1162> [GS Search].
- Pranav, K., & Sarma, K. (2023). Origin, development and uses of machine learning. *International Journal For Multidisciplinary Research*, 5(1), 1–18. <https://doi.org/10.36948/ijfmr.2023.v05i01.1367> [GS Search].
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. <https://doi.org/10.48550/arXiv.2406.14491> [GS Search].
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research*, 21(140), 1–67. <https://doi.org/10.48550/arXiv.1910.10683> [GS Search].
- Rosa, B. A., & Mello, R. F. (2022). Análise automatizada de coesão em redações do ensino fundamental por meio de técnicas de processamento de linguagem natural. *Anais Estendidos do XI Congresso Brasileiro de Informática na Educação*, 144–149. https://doi.org/10.5753/cbie_estendido.2022.226737 [GS Search].

- Silva Filho, M., Nascimento, A., Miranda, P., Rodrigues, L., Cordeiro, T., Isotani, S., Bittencourt, I., & Mello, R. (2023). Automated Formal Register Scoring of Student Narrative Essays Written in Portuguese. *Anais do II Workshop de Aplicações Práticas de Learning Analytics em Instituições de Ensino no Brasil*, 1–11. <https://doi.org/10.5753/wapla.2023.236084> [GS Search].
- Souza, F., Nogueira, R., & Lotufo, R. (2020). BERTimbau: Pretrained BERT Models for Brazilian Portuguese. Em R. Cerri & R. C. Prati (Ed.), *Intelligent Systems* (pp. 403–417). Springer International Publishing. https://doi.org/10.1007/978-3-030-61377-8_28 [GS Search].
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2023). Attention Is All You Need. <https://doi.org/10.48550/arXiv.1706.03762> [GS Search].
- Vogel, D., Ramos, A. M., & Franzoni, A. M. B. (2025). Transformando a educação com Large Language Models (LLMs): benefícios, limitações e perspectivas. *Caderno Pedagógico*, 22(4), e13846. <https://doi.org/10.54033/cadpedv22n4-013> [GS Search].
- Watanabe, S. (2023). Tree-Structured Parzen Estimator: Understanding Its Algorithm Components and Their Roles for Better Empirical Performance. <https://doi.org/10.48550/arXiv.2304.11127> [GS Search].
- Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent Trends in Deep Learning Based Natural Language Processing. *IEEE Computational Intelligence Magazine*, 13(3), 55–75. <https://doi.org/10.1109/MCI.2018.2840738> [GS Search].
- Zhao, F. (2017). Natural Language Processing—Part I: Primer—Unveiling the Hidden Information in Earnings Calls. *S&P Global Market Intelligence Quantamental Report*, September. <https://doi.org/10.2139/ssrn.4852446> [GS Search].