Universidade Federal de São Carlos

Centro de Ciências Exatas e de Tecnologia
Departamento de Computação

Processamento de Linguagem Natural - Relatório Final

Professora: Dra. Helena Caseli

Allan Moreira de Almeida - 811404 - Engenharia de Computação
João Pedro Scarpat Bizineli - 811808 - Engenharia de Computação
Lucca Renato Guerino - 800224 - Engenharia de Computação
Victor Vendrera Vieira - 801645 - Engenharia de Computação

São Carlos - SP

1. Revisão da tarefa de PLN escolhida

1.1 Tarefa escolhida

Relembrando o que foi apresentado no primeiro relatório, a tarefa escolhida foi a de atribuição automática de notas em redações. Ela consiste em criar sistemas que avaliam textos com base em critérios definidos, imitando a correção humana. As redações são avaliadas em aspectos como coerência, gramática, argumentação e estrutura, com notas numéricas atribuídas a cada um.

Essa tarefa envolve desafios como interpretação de texto, detecção de erros, avaliação subjetiva e variação de estilo. Como há subjetividade na correção, os modelos automáticos são comparados à consistência entre avaliadores humanos, o objetivo portanto é que o sistema atribua automaticamente essas notas a partir do conteúdo textual.

Uma revisão feita por Lima et al. [2] mostrou que, apesar de avanços, a avaliação automática em português ainda é limitada em relação ao inglês. A maioria dos trabalhos usa aprendizado de máquina com extração de atributos linguísticos, e o uso de deep learning ainda é raro. Também há poucos dados disponíveis e falta de testes práticos em ambientes reais, dificultando a validação educacional no Brasil.

Essa tarefa enfrenta desafios comuns em PLN, como interpretação semântica e sintática, detecção de erros gramaticais e ortográficos, e avaliação subjetiva de argumentos e estrutura. Além disso, há variações no estilo e vocabulário dos estudantes, o que exige que os modelos sejam capazes de enfrentar ambiguidade e subjetividade, já que diferentes corretores podem dar notas diferentes para o mesmo texto.

1.2 Detalhamento do dataset

O dataset escolhido para o trabalho foi o “Brazilian Portuguese Narrative Essays Dataset” disponível em [3], o qual é uma coleção de redações narrativas escritas por estudantes brasileiros do ensino fundamental (do 5º ao 9º ano). Este dataset foi desenvolvido no contexto da competição PROPOR'24, cujo objetivo é fomentar o desenvolvimento de sistemas automáticos capazes de avaliar redações em português, auxiliando professores no processo de feedback formativo.
O dataset é fornecido no formato CSV e contém os seguintes campos:

- **ID**: Identificador único de cada redação.
- **Texto do Estudante**: A redação narrativa escrita pelo aluno.
- **Texto Motivador (Prompt)**: Texto que serviu de inspiração para a redação.
- **Notas de Avaliação**: Quatro competências avaliadas por dois corretores humanos, com notas de 1 a 5 (Registro formal, coerência, estrutura da retórica narrativa e coesão).

O corpus é formado por 1235 redações e foi dividido em três subconjuntos com a seguinte divisão: 60% (740) para treinamento, 10% (125) para validação e 30% (370) para teste.

É importante ressaltar que, como as anotações foram transcritas por um humano, existem algumas flags/tokens especiais utilizados para simbolizar diferentes elementos do texto, são eles:

Flag/Token	Significado
[P], [p], {P}, {p}	Indica a presença de um novo parágrafo naquele ponto
[S], [s]	Representa um símbolo presente na redação manuscrita
[T], [t], {t}	Os tokens seguintes representam o título da redação
[R], [X], [~], [r], [x], {x}	Indica uma rasura na redação manuscrita
[?], {?}, [?], {?}	Token desconhecido na redação manuscrita
[LC], [LT], [lt]	Os tokens seguintes não seguiram uma linha reta na redação

Um exemplo de prompt motivador para as redações dos estudantes é: “Eu encontrei em cima do armário alguns potes com tinta. Então resolvi pintar na parede do meu quarto alguns animais. Tempo depois, quando voltei no quarto, não havia mais nenhuma pintura e as roupas da gaveta estavam espalhadas pelo chão.”

Enquanto um exemplo de texto de um estudante é: “[T] A menina alice [P] Era um dia em solarado bem cedo por vouta de 9:30 da mãnha, a menina alice já estava acordada. Ela deçeu para tomar seu café da mãnha então ela [X] uns pote em cima do armário era de tinta. A menina pegou e foi para seu quarto, ela amaria fazendo então falou: -Vou fazer animais na parede do meu quarto! Então a garota pesou em que tipo de animais ela podia dezenha, a garota pensou e pensou e falou: -Um, uma galinha, vaquinhas, porquinhos e cavalos! E ela dezenhou, sua mãe chamou para almoçar e ela foi, quando voutou não tinha dezenho e suas roupas estava jogada no chão e ela falou: -meu deus que a aconteceu sua mãe viu e mandou ela arru-”

Para este texto de exemplo, as notas atribuídas foram: 3 para registro formal, 3 para coerência temática, 4 para estrutura retórica narrativa e 3 para coesão.

A distribuição das notas para cada uma das quatro competências avaliadas demonstra padrões específicos:

- **Registro Formal (formal_register)**: A maioria das redações recebeu nota 3, indicando uma tendência centralizada. Há uma menor frequência de notas 1 e 5.
- **Coerência Temática (thematic_coherence)**: Há uma distribuição mais ampla, com picos nas notas 1 e 3.
- **Estrutura da Retórica Narrativa (narrative_rhetorical_structure)**: A nota 4 é a mais frequente, seguida pela nota 3, com poucas ocorrências de notas 1 e 2.
- **Coesão (cohesion)**: Similar ao registro formal, a nota 3 é predominante, seguida pelas notas 2 e 4.

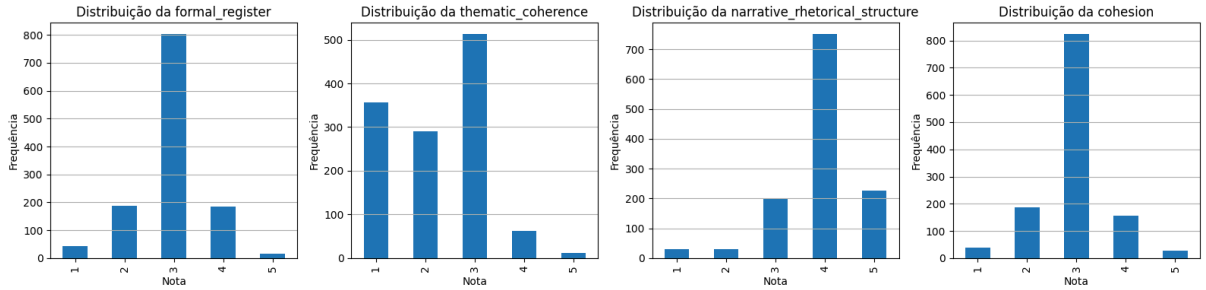

figura 1: Distribuição de notas

Essas distribuições de notas ressaltam a importância de modelos que possam aprender os padrões de correção para essa faixa etária e lidar com a subjetividade e desequilíbrio das classes. Esse desequilíbrio é percebido em torno das notas médias, no caso do formal_register e cohesion, enquanto na thematic_coherence as notas se concentram na metade inferior e na narrative_rhetorical_structure se concentram na metade superior.

Isso indica o viés de correção dos corretores e aquilo que foi usado como critério para corrigir os textos dos alunos, como por exemplo a estrutura narrativa tende a ser mais fácil de ser mantida em um texto escrito por uma criança, enquanto a coerência e coesão exigem mais dos alunos.

1.3 Medidas de avaliação escolhidas

Considerando que as notas são valores discretos de 1 a 5 e a distribuição de notas não é uniforme, é importante se atentar para os métodos de avaliação a serem utilizados nessa tarefa. Dessa forma, foram escolhidos os seguintes métodos:

- Precisão, Recall e F1-Score: Métricas clássicas que permitem avaliar o desempenho do modelo na predição de cada categoria de nota, importantes para analisar o comportamento do modelo em classes específicas, considerando os conjuntos de validação e teste. Para o relatório, optou-se por fazer uma análise mais geral. Por isso, não serão avaliadas diretamente, porém podem ser consultadas a partir do código fonte fornecido no Github do projeto.
- Quadratic Weighted Kappa (QWK): Métrica utilizada para classificação ordinal, permitindo medir concordância entre notas preditas e reais, ponderando os erros pela distância entre as categorias. Dessa forma, Penaliza mais os erros "distantes".
- Spearman's rank correlation: Aplica correlação entre ordenações, medindo a monotonicidade entre as notas verdadeiras e preditas, captura se o modelo acerta a tendência (mesmo errando o valor exato).

2. Estratégias adotadas

2.1 Estimativa a partir de um LLM

A técnica de Processamento de Linguagem Natural adotada nesta estratégia consiste na utilização de um modelo pré-treinado da família BERT, especificamente o ‘neuralmind/bert-base-portuguese-cased’, para a tarefa de regressão aplicada à avaliação

automática de textos dissertativos. O objetivo é treinar individualmente um modelo para cada um dos quatro atributos avaliativos.

2.1.2 Dificuldades encontradas

A dificuldade encontrada nesta abordagem está no desafio de fazer o modelo entender e relacionar notas que têm pouca ocorrência. Por exemplo, no atributo '*formal_register',* há muitas amostras de valor '3' e poucas amostras para as demais. Isso causa um "vício" no modelo e uma possível interpretação errada dos resultados, os quais serão abordados na seção de resultados.

2.1.3 Aplicação do modelo

O processo começa com o carregamento e pré-processamento dos dados de treino, validação e teste, que são compostos por textos (ensaios) acompanhados das respectivas notas atribuídas para cada atributo. As redações continham "flags/tokens especiais" (como [P], [T], [X], entre outros) que são marcadores da transcrição manual das redações e não faziam parte do conteúdo textual original a ser avaliado. Dessa forma, optou-se por eliminá-los do texto das redações.

Como próximo passo, os textos são tokenizados utilizando o tokenizador do modelo BERT, com padding e truncamento para um comprimento máximo de 512 tokens, o que garante compatibilidade com a arquitetura do modelo base. Em seguida, realiza-se a tokenização dos textos e a definição do modelo BERT com uma única saída contínua (num_labels=1), adequada à tarefa de regressão.

Durante o treinamento, o modelo é avaliado a cada época com base na métrica QWK, buscando o maior valor dessa pontuação, cujos cálculos são definidos em uma função personalizada. A função de métricas implementada no código tem como objetivo medir o desempenho do modelo em prever corretamente as notas atribuídas aos textos para cada um dos atributos avaliativos.

Após o treinamento, para tratar os dados de predição, os valores são achatados com flatten() para garantir que estejam em uma dimensão compatível com os rótulos reais. Em seguida, os valores preditos são arredondados para o inteiro mais próximo, simulando o processo de atribuição de notas discretas. Como a escala de avaliação vai de 1 a 5, os valores arredondados são então limitados a esse intervalo por meio da função np.clip(), evitando predições fora da faixa válida (por exemplo, 0 ou 6). Em seguida, as métricas são calculadas e retornadas para o treinador.

2.2 Extração de Informações + Random Forests

Esta estratégia teve como objetivo transformar as redações textuais em um conjunto de características numéricas e categóricas, que seriam então utilizadas para treinar um modelo de Regressão Random Forest para prever as notas das redações. A implementação seguiu as etapas de pré-processamento, extração de características e aplicação do modelo.

2.2.1 Recursos em língua portuguesa usados e Pré-processamento

A etapa inicial envolveu a preparação dos dados. Os datasets de treinamento, validação e teste, que contêm as redações e suas respectivas notas, foram carregados. Um passo fundamental de pré-processamento foi a remoção de informações que não seriam relevantes para a avaliação da redação em si: a coluna 'prompt' (que representa o texto motivador da redação) e a coluna 'id' (um identificador único) foram descartadas.

Adicionalmente, as redações continham "flags/tokens especiais" (como [P], [T], [X], entre outros) que eram marcadores da transcrição manual das redações e não faziam parte do conteúdo textual original a ser avaliado. Foi decidido realizar a remoção desses tokens, utilizando expressões regulares para identificá-los e eliminá-los do texto das redações. Isso garantiu que o modelo trabalhasse apenas com o texto puro da redação.

Foi realizada uma análise exploratória da distribuição das notas para cada uma das quatro competências avaliadas: "Registro formal", "Coerência temática", "Estrutura da retórica narrativa" e "Coesão". Essa análise é vital para compreender a representatividade de cada nota no dataset. Os gráficos gerados mostraram que as notas não seguem uma distribuição uniforme para todas as competências. Por exemplo, para "Registro formal" e "Coesão", a nota 3 é a mais frequente, enquanto "Coerência temática" apresenta picos nas notas 1 e 3, e "Estrutura da retórica narrativa" tem a nota 4 como a mais comum. Essa heterogeneidade na distribuição das notas é um desafio, pois modelos tendem a predizer as classes majoritárias com mais frequência, impactando o desempenho em classes menos representadas.

2.2.2 Extração de Características

A etapa de extração de características transformou o texto das redações em atributos numéricos. Para isso, foi utilizada uma função customizada extrair_caracteristicas, que calcula: o número de palavras, o número de frases, o comprimento médio das frases, o número de conectivos e o número de advérbios terminados em '-mente'. Essas características foram escolhidas por serem indicadores simples de complexidade textual e organização.

2.2.3 Dificuldades encontradas

A principal dificuldade encontrada com esta abordagem reside na capacidade das características extraídas em capturar a complexidade semântica e contextual das redações. Embora características como número de palavras ou comprimento médio de frases possam indicar aspectos superficiais da escrita, elas são limitadas na compreensão de nuances de coerência, coesão, registro formal e estrutura retórica que um avaliador humano ou um modelo de linguagem mais complexo consegue inferir.

A subjetividade intrínseca da avaliação humana, somada aos possíveis defeitos na transcrição dos dados originais para o meio digital, também representou um desafio, dificultando a correta aprendizagem dos padrões de correção. As técnicas de PLN baseadas em características superficiais podem não levar em conta o contexto necessário para corrigir uma redação de ensino fundamental, o que pode levar a divergências em relação às notas de professores com escalas de correção bem definidas.

2.2.4 Aplicação do Modelo

Para a fase de modelagem, foi empregado o RandomForestRegressor, um algoritmo de aprendizado de máquina baseado em árvores de decisão. A escolha do Random Forest deve-se à sua robustez e boa performance com conjuntos de dados que possuem um número razoável de características explicativas, além de sua relativa simplicidade de implementação. Um modelo independente foi treinado para cada uma das quatro competências de avaliação das redações. Os parâmetros do modelo, como o número de estimadores e a profundidade máxima das árvores, foram ajustados para otimizar o desempenho.

Após o treinamento, o modelo foi utilizado para gerar previsões das notas nos conjuntos de validação e teste. Essas previsões, que inicialmente são valores contínuos (de regressão), foram então arredondadas para o número inteiro mais próximo e ajustadas para se encaixarem no intervalo de notas permitido (de 1 a 5). Esse arredondamento é crucial para converter a saída de regressão em uma classificação inteira, que se alinha com o formato das notas reais.

3. Resultados

3.1 Avaliação quantitativa

A avaliação quantitativa do desempenho dos modelos Random Forest e BERTimbau foi realizada por meio de um conjunto de métricas projetadas para refletir a eficácia em tarefas de regressão quanto para classificação inteira. A natureza das notas, que variam discretamente de 1 a 5, exigiu a aplicação de métricas que pudessem capturar a concordância e a acurácia de forma robusta. As principais métricas utilizadas foram o Mean Absolute Error (MAE), que mede a média dos erros absolutos entre as previsões e os valores reais , o Quadratic Weighted Kappa (QWK), uma métrica de concordância para classificação ordinal que penaliza erros de forma proporcional à sua distância e a Correlação de Postos de Spearman, que avalia a monotonicidade entre as ordenações das notas verdadeiras e preditas.

3.1.1 Análise Comparativa dos Resultados

Abaixo, apresentamos uma tabela consolidando as métricas de desempenho para cada modelo e para cada uma das quatro competências avaliativas. Os valores refletem os resultados obtidos nos respectivos experimentos.

Critério	Modelo	QWK	Spearman
Registro Formal	BERTimbau	0,5900	0,6426
	Random Forest	0,1022	0,4577
Coerência Temática	BERTimbau	0,7667	0,8296
	Random Forest	0,0000	0,0933
Estrutura da Retórica Narrativa	BERTimbau	0,4511	0,4066

Critério	Modelo	QWK	Spearman
	Random Forest	0,3037	0,3850
Coesão	BERTimbau	0,5291	0,5339
	Random Forest	0,1677	0,4069

Observações dos Resultados

- **Quadratic Weighted Kappa (QWK)**: O BERTimbau consistentemente obteve valores de QWK significativamente mais altos para todas as competências. Isso é crucial, pois o QWK é a métrica mais relevante para classificação ordinal, indicando uma concordância muito maior entre as notas preditas pelo BERTimbau e as notas reais atribuídas por avaliadores humanos. O QWK do Random Forest manteve-se muito baixo (próximo de 0), especialmente para "Coerência Temática" (0.0000), sinalizando um alinhamento nulo com os padrões de correção.
- **Correlação de Spearman**: A correlação de Spearman, que mede a monotonicidade entre as ordenações, também favoreceu o BERTimbau em critérios como Registro Formal e Coerência Temática, indicando que o modelo acertou mais a tendência das notas.
- **Random Forest como Baseline**: O Random Forest serviu como uma boa linha de base, mas suas limitações são evidentes. Seu QWK permaneceu consideravelmente inferior ao do BERTimbau (0.3037 contra 0.4511, respectivamente) para estrutura retórica. A incapacidade do Random Forest de extrair e processar características semânticas profundas do texto, dependendo apenas de características lexicais e estruturais mais simples, resultou em uma performance inferior para a tarefa de avaliação de redações.

Em síntese, a avaliação quantitativa valida a maior eficácia do BERTimbau na avaliação automática de textos, devido à sua capacidade avançada de compreensão linguística, em contraste com o desempenho limitado do Random Forest para esta tarefa específica, especialmente no que tange à captura da subjetividade e nuances da avaliação humana.

Entretanto, apesar do BERTimbau ter tido melhores resultados, para todas as competências, a precisão de acerto de notas muito frequentes foi boa (70% de precisão em média) porém para notas pouco frequentes obteve precisão ruim (0% de precisão em vários casos), como pode ser conferido nas saídas do código fonte.

3.2 Avaliação qualitativa

Considerando uma avaliação qualitativa dos resultados, podemos considerar alguns textos em específico para entender como que os modelos avaliaram e se os resultados são adequados em casos intermediários e extremos.

Considerando a redação abaixo, ela é uma avaliação intermediária, a qual segue a distribuição da maioria dos dados na maioria das notas, portanto a tabela a seguir apresenta as notas originais atribuídas e as notas atribuídas por ambos os modelos.

Texto: “[T] Os potes de tintas

[P] Depois a minha mãe lipou a parede e depois meu irmãozinho pegou os potes de tintas para pintar os desenhos dele e jogou minhas roupas no chão e saio depois quando eu terminei de arrumar minhas roupas fui no quarto dele tava tudo melado de tinta e tinha tinta para todo lugar nas parede no chão mais os desenhos que ele pintou estava linado mais eu estava um pouco triste porque ele pegou as minhas tintas eu resolvir sair mais quando eu voutei para casa eu fiquei muito feliz porque tinha muitos podes de tintas fiz muitos desenhos passei a tarde na Escola com minhas amiguinhas a catharina a Karol a Raquel e a ana júlia e fiquei muito feliz.”

	formal_register	**thematic_coherence**	**narrative_rhetorical_structure**	**cohesion**
Original	3	3	4	3
BERTimbau	3	3	4	3
Random Forest	3	2	4	3

Agora considerando um caso mais extremo, onde o modelo deveria perceber as nuances e defeitos do texto para atribuir notas mais baixas, temos o seguinte texto:

Texto: “[T] (a Menina Da tinta [S] >

— um dia eu e minha prima ulece turu umlunto na rua e loyo eu u nu alguete

— Outro poty ele tintu e eu falei — olhei primeiro tinha umas Pegur minha Prima falou:

— Umas [X] [X] pegou [?] tinta e fomos para casa ai chegamos em casa ai agente penjou

— ogo vamos [?]ger com as tintas penjumos no no outro dia a mãe da minha primo ci

— Chegou em casa ai agente perguntou para ela o que agente podia

—que agente podia pintar o quarto da gente et[?] agente tetou a agente pintou e [?] ”

	formal_register	thematic_coherence	narrative_rhetorical_structure	cohesion
Original	1	2	3	2
BERTimbau	3	3	4	3
Random Forest	3	2	4	3

Em ambos os casos, percebe-se que o modelo BERTimbau, assim como o modelo do Random Forest, tiveram um desempenho bom ao identificar o caso intermediário e ruim para discriminar o caso extremo, o que mostra as limitações dos modelos, que apenas aprenderam os padrões mais comuns das notas e está tentando aplicar para todos os casos, sem aprender de fato os erros ortográficos e de estrutura que são relevantes para corrigir um texto.

4. Considerações finais

Considerando o que foi explorado neste trabalho, é possível concluir que a tarefa de correção automática de redações é significativamente mais complexa que a de classificação simples a partir de dados tabulares.

A complexidade das entradas e a natureza subjetiva da correção não se traduz totalmente para os dados e, utilizando o dataset escolhido, apresenta-se um problema de desbalanceamento do dataset. Estratégias de oversampling ou undersampling foram consideradas, porém no caso de aumentar os dados, não é possível criar redações novas que se encaixem no contexto de forma tradicional, enquanto diminuir as classes majoritárias diminuiria muito a quantidade de dados, gerando overfitting dos modelos aos dados.

Como forma de melhorar os resultados obtidos neste experimento, pode-se aplicar uma abordagem para cada modelo: Com o modelo de Random Forest, a mudança crítica a ser feita é considerar extrair características mais complexas e que consigam captar mais riqueza do texto. Por outro lado, o modelo BERT seria beneficiado com um *dataset* mais equilibrado e uniforme, com amostras abundantes para todas as notas de cada competência.

5. Referências

[1] Processamento de Linguagem Natural: Conceitos, Técnicas e Aplicações em Português (3ª Edição). Disponível em: <https://brasileiraspln.com/livro-pln/3a-edicao/Livro_PLN_BPLN_3ed.pdf>.

[2] Lima, T. B., Silva, I. L. A., Freitas, E. L. S. X., & Mello, R. F. (2023). Avaliação Automática de Redação: Uma revisão sistemática. Revista Brasileira de Informática na Educação – RBIE, 31, 205-221. https://doi.org/10.5753/rbie.2023.2869

[3] MOESIOF. Brazilian Portuguese Narrative Essays Dataset. Disponível em: <https://www.kaggle.com/datasets/moesiof/portuguese-narrative-essays/data>.

[4] Text classification. Disponível em: <https://huggingface.co/docs/transformers/en/tasks/sequence_classification>.

[5] WIKIPEDIA CONTRIBUTORS. Spearman's rank correlation coefficient. Disponível em: <https://en.wikipedia.org/wiki/Spearman%27s_rank_correlation_coefficient>.

[6] ARORAAMAN. Quadratic Kappa Metric explained in 5 simple steps. Disponível em: <https://www.kaggle.com/code/aroraaman/quadratic-kappa-metric-explained-in-5-simple-steps>.