UNIVERSIDADE FEDERAL DO CEARÁ

# TECHNICAL REPORT NLP

Aluno: RUAN RODRIGUES SOUSA

## Uso de LLM Pipeline para Análise de Sentimentos e Comparação com Vetorização Tradicional

LINK PARA TER ACESSO AO NOTEBOOK NO COLAB: AP2_nlp.ipynb

### 1. Introdução

Neste estudo, foi utilizada a pipeline de Modelos de Linguagem de Grande Escala (LLMs) da biblioteca Transformers da Hugging Face, especificamente o modelo BERT para português, para a análise de sentimentos em um dataset de tweets. Essa abordagem moderna oferece uma alternativa sofisticada às técnicas tradicionais de vetorização, como CountVectorizer e TF-IDF.

### Carregamento do Modelo e Tokenizer

O modelo BERT e seu tokenizer foram carregados, permitindo que o modelo fosse adaptado para a tarefa de classificação binária, configurado para distinguir entre sentimentos positivos e negativos. O ajuste de parâmetros, como o dropout, contribui para a regularização do modelo durante o treinamento.

### Configuração da Pipeline de Análise de Sentimentos

A pipeline para análise de sentimentos foi criada, simplificando o uso do modelo BERT. Essa configuração automatiza o pré-processamento dos textos, a inferência e o pós-processamento dos resultados, permitindo que a operação ocorra de forma eficiente, especialmente se uma GPU estiver disponível.

A aplicação da pipeline com o modelo BERT para a análise de sentimentos em tweets envolve:

2. **Carregamento dos Dados**: O dataset contém tweets em português, classificados em sentimentos positivos e negativos.

3. **Pré-processamento Automático**: A pipeline realiza a tokenização, uma etapa que converte os textos dos tweets em um formato que o modelo consegue processar, preservando a ordem e o contexto das palavras.
4. **Inferência e Resultados**: A pipeline aplica o modelo BERT treinado em português para inferir o sentimento dos tweets, retornando uma classificação binária (positivo/negativo) com base no conteúdo do texto.

**Comparação com Técnicas Tradicionais de Vetorização**

Após a aplicação da pipeline com LLM, os resultados são comparados com técnicas mais tradicionais, como o **CountVectorizer** e o **TF-IDF**, que têm sido amplamente utilizadas na análise de textos.

**CountVectorizer e TF-IDF**

**CountVectorizer** e **TF-IDF** são abordagens baseadas na **frequência de palavras**. O **CountVectorizer** cria uma representação do texto em forma de uma matriz de contagem de palavras, enquanto o **TF-IDF** ajusta essa contagem levando em consideração a relevância das palavras, atribuindo pesos maiores às palavras menos comuns e mais específicas de cada documento.

Embora esses métodos sejam rápidos e computacionalmente leves, suas principais limitações incluem:

- **Falta de Compreensão Contextual**: Essas técnicas tratam as palavras de forma independente, ignorando o contexto no qual elas são usadas. Isso resulta em uma interpretação limitada do texto, o que pode comprometer a classificação de sentimentos em textos com nuances, como ironia ou negação.
- **Sensibilidade a Palavras Individuais**: No CountVectorizer e no TF-IDF, palavras frequentes podem dominar a interpretação, mesmo que, em alguns casos, essas palavras não carreguem o peso emocional ou semântico correto.

**BERT via Pipeline**

O **BERT**, por outro lado, oferece uma compreensão mais profunda do texto ao analisar o contexto completo de cada palavra dentro da frase. Isso permite que o modelo identifique a relação entre as palavras e interprete o significado subjacente, mesmo em textos curtos e informais, como os tweets. Alguns pontos de destaque incluem:

- **Análise de Contexto**: O BERT captura as interações entre as palavras em uma frase, entendendo como elas se influenciam mutuamente. Isso é particularmente útil em situações onde o sentimento não é explicitamente claro apenas pelo uso individual de palavras.
- **Melhor Performance em Textos Curto-Formais**: Em textos concisos e informais como tweets, o uso de um modelo contextual como o BERT supera a análise de frequência de palavras, uma vez que a linguagem usada em mídias sociais frequentemente possui elementos informais que as técnicas tradicionais não capturam adequadamente.

## 5. Observações

Apesar de o modelo **BERT** via pipeline ser amplamente reconhecido por sua capacidade de compreender o contexto das palavras de maneira profunda e precisa, o desempenho observado neste caso específico não atingiu o seu potencial máximo. Existem dois fatores principais que contribuíram para esse resultado:

**Recursos Computacionais Insuficientes**:

**Modelos LLMs**, como o BERT, são computacionalmente intensivos e requerem um grande poder de processamento, especialmente ao lidar com dados em tempo real ou quando aplicados a conjuntos de dados substanciais. A falta de um ambiente de execução com hardware adequado, como GPUs de alta performance, pode limitar o desempenho do modelo. O processo de inferência e treinamento se torna mais lento, e isso afeta diretamente a capacidade de ajustar hiperparâmetros ou realizar várias iterações para melhorar o modelo.

Além disso, o aumento do **dropout** para 0.5 no modelo para mitigar o overfitting pode reduzir a precisão final, especialmente se o modelo não tiver tempo suficiente ou recursos computacionais adequados para treinar com várias repetições.

**Tamanho Limitado do Dataset**:

Embora o dataset de tweets usado contenha uma boa amostra de dados, o fato de ser reduzido para apenas 20% do total disponível significa que o modelo não tem acesso a um volume amplo de exemplos para generalizar melhor. **Modelos LLMs** são particularmente eficazes quando treinados com grandes quantidades de dados, pois isso lhes permite capturar padrões sutis e variações de linguagem.

Além disso, a variação limitada de exemplos pode levar o modelo a uma performance subótima, com dificuldades em capturar todas as nuances presentes nos tweets, como sarcasmo e ironias mais complexas.

Essas duas restrições—falta de poder computacional adequado e a redução do tamanho do dataset—ajudam a explicar por que o resultado final da aplicação do **BERT** via pipeline, embora promissor, não foi o melhor possível. Para uma aplicação otimizada e uma análise mais profunda do potencial do LLM, seria ideal contar com um **maior volume de dados** e **maior capacidade de processamento**.

## 6. Resultados e discussão

**ANALISANDO LLM (BERT) EM CONJUNTO COM A PIPELINE "sentiment-analysis"**

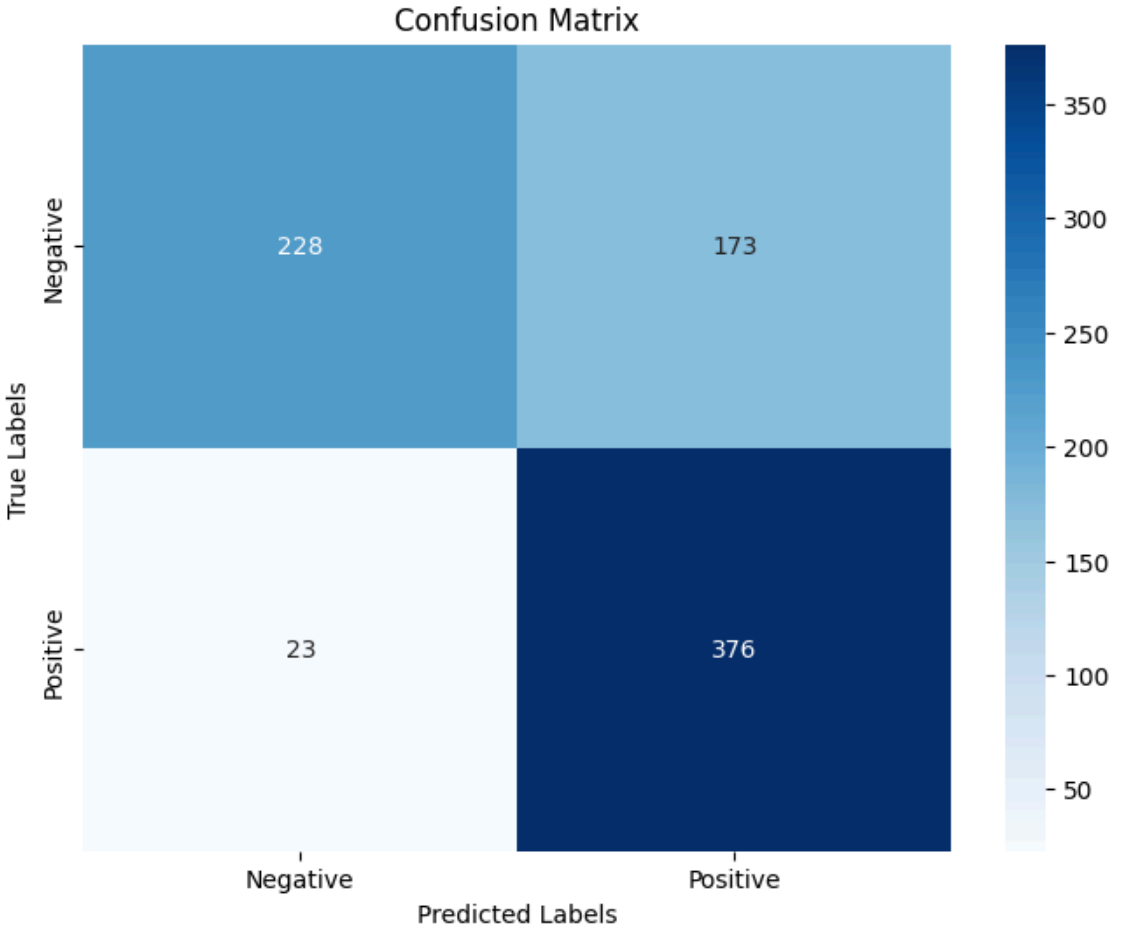
LLM (BERT)

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.91 | 0.57 | 0.70 | 401 |
| 1 | 0.68 | 0.94 | 0.79 | 399 |
| accuracy | | | 0.76 | 800 |
| macro avg | 0.80 | 0.76 | 0.75 | 800 |
| weighted avg | 0.80 | 0.76 | 0.75 | 800 |

Os resultados obtidos mostram um desempenho equilibrado do modelo, com uma acurácia de 0.7550. A precisão de 0.7966 indica eficácia na identificação de sentimentos positivos, mas o recall de 0.7555 sugere que o modelo pode ter perdido alguns sentimentos positivos. O F1-Score de 0.7463 reflete um equilíbrio entre precisão e recall, apontando espaço para melhorias, especialmente na detecção de sentimentos negativos.

O classification report revela que o modelo teve alta precisão (0.91) na identificação de sentimentos negativos, mas um recall mais baixo (0.57), indicando que muitos sentimentos positivos foram erroneamente classificados como negativos. Para sentimentos positivos, o recall foi elevado (0.94), enquanto a precisão foi de 0.68.

A matriz de confusão complementa essa análise, mostrando 228 verdadeiros negativos e 376 verdadeiros positivos, mas com 173 sentimentos positivos classificados incorretamente como negativos e 23 negativos identificados como positivos. Essa discrepância sugere a necessidade de ajustes para melhorar a sensibilidade do modelo em relação aos sentimentos negativos, talvez através de um melhor balanceamento nos dados de treinamento.




**LLM (BERT) X VETORIZAÇÃO COUNTVECTORIZER somente c/pré-processamento**

LLM (BERT)

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.91 | 0.57 | 0.70 | 401 |
| 1 | 0.68 | 0.94 | 0.79 | 399 |
| accuracy | | | 0.76 | 800 |
| macro avg | 0.80 | 0.76 | 0.75 | 800 |
| weighted avg | 0.80 | 0.76 | 0.75 | 800 |

Vetorização CountVectorizer

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.72 | 0.81 | 0.75 | 401 |
| 1 | 0.77 | 0.69 | 0.72 | 399 |
| accuracy | | | 0.73 | 800 |
| macro avg | 0.74 | 0.74 | 0.74 | 800 |
| weighted avg | 0.74 | 0.74 | 0.74 | 800 |

## Análise Comparativa

1. **Pré-processamento**:
   - O modelo CountVectorizer foi aprimorado com **pré-processamento**, o que contribuiu para suas métricas de desempenho. Por outro lado, a pipeline LLM não exige um pré-processamento extensivo devido à sua capacidade de lidar com texto de forma mais robusta.
2. **Acurácia**:
   - O modelo LLM apresentou uma acurácia de **0.76**, ligeiramente superior à **0.73** do CountVectorizer.
3. **Precisão**:
   - O BERT teve uma precisão excepcional para a classe negativa (**0.91**), enquanto o CountVectorizer ficou em **0.72**.

4. **Recall**:

- O CountVectorizer superou o BERT em recall para a classe negativa (**0.81** contra **0.57**), indicando que foi mais eficaz na identificação de sentimentos negativos.

5. **F1-Score**:
   - O BERT obteve um F1-Score maior para a classe positiva (**0.79**), enquanto o CountVectorizer alcançou **0.72**.

**LLM (BERT) X VETORIZAÇÃO TD-IDF somente c/pré-processamento**

LLM (BERT)

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.91 | 0.57 | 0.70 | 401 |
| 1 | 0.68 | 0.94 | 0.79 | 399 |
| accuracy | | | 0.76 | 800 |
| macro avg | 0.80 | 0.76 | 0.75 | 800 |
| weighted avg | 0.80 | 0.76 | 0.75 | 800 |

*Vetorização TF-IDF*

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.76 | 0.73 | 0.75 | 401 |
| 1 | 0.74 | 0.77 | 0.76 | 399 |
| accuracy | | | 0.75 | 800 |
| macro avg | 0.75 | 0.75 | 0.75 | 800 |
| weighted avg | 0.75 | 0.75 | 0.75 | 800 |

### Análise Comparativa

1. **Pré-processamento**:
   - O modelo **TF-IDF** se beneficiou do **pré-processamento**, que envolveu técnicas como remoção de stop words e normalização dos dados. Isso melhorou a qualidade da entrada para o modelo, refletindo-se em métricas mais equilibradas.

   - Em contraste, o **BERT** através da pipeline não requer um pré-processamento tão extensivo, pois já é projetado para lidar com texto em formato bruto.
2. **Acurácia**:
   - Ambas as abordagens apresentaram uma acurácia competitiva, com o modelo TF-IDF em **0.75** e o BERT ligeiramente melhor, com **0.76**.
3. **Precisão**:
   - O BERT teve uma precisão significativamente maior para a classe negativa (**0.91**), enquanto o TF-IDF foi mais equilibrado, com **0.76** e **0.74** nas duas classes.
4. **Recall**:
   - O TF-IDF superou o BERT em recall para a classe negativa (**0.73** contra **0.57**), mostrando que foi mais eficaz em capturar sentimentos negativos. Para a classe positiva, o BERT teve um recall superior (**0.94**), refletindo sua eficácia em detectar sentimentos positivos.
5. **F1-Score**:
   - O F1-Score do TF-IDF é mais equilibrado entre as classes, com ambos em torno de **0.75**, enquanto o BERT teve um desempenho melhor para a classe positiva (**0.79**) e inferior para a classe negativa (**0.70**).

**TD-IDF COM STEMMING X LMM (BERT)**

Por fim, iremos comparar a abordagem de modelagem de linguagem LLM (BERT) com a vetorização TF-IDF combinada com stemming. Essa técnica de vetorização se destacou na análise anterior (AV1), apresentando um desempenho superior em relação à lemmatização. Essa comparação permitirá avaliar como as abordagens tradicionais de vetorização se posicionam frente à capacidade contextualizada do modelo BERT, fornecendo insights valiosos sobre a escolha da metodologia mais eficaz para a análise de sentimentos no conjunto de dados.

LLM (BERT)

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.91 | 0.57 | 0.70 | 401 |
| 1 | 0.68 | 0.94 | 0.79 | 399 |
| accuracy | | | 0.76 | 800 |
| macro avg | 0.80 | 0.76 | 0.75 | 800 |
| weighted avg | 0.80 | 0.76 | 0.75 | 800 |

*TF-IDF com Stemming:*

| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| 0 | 0.75 | 0.72 | 0.74 | 401 |
| 1 | 0.73 | 0.75 | 0.74 | 399 |
| accuracy | | | 0.73 | 800 |
| macro avg | 0.74 | 0.74 | 0.74 | 800 |
| weighted avg | 0.74 | 0.74 | 0.74 | 800 |

## Análise Comparativa

1. **Acurácia**:
    - O modelo LLM apresentou uma acurácia superior de **0.76**, em comparação à **0.73** do TF-IDF com stemming, sugerindo uma maior eficácia geral na classificação.

2. **Precisão**:
    - O BERT teve uma precisão significativamente melhor para a classe negativa (**0.91**), enquanto o TF-IDF ficou em **0.75**. Para a classe positiva, o TF-IDF manteve uma precisão similar de **0.73**.

3. **Recall**:
    - O TF-IDF foi mais eficaz em identificar sentimentos negativos, apresentando um recall de **0.72**, enquanto o BERT alcançou apenas **0.57**

para essa classe. Para a classe positiva, o BERT teve um recall muito superior (**0.94**).

4. **F1-Score**:
    - O TF-IDF manteve um F1-Score equilibrado de **0.74** para ambas as classes, enquanto o BERT teve um desempenho melhor na classe positiva (**0.79**) e inferior na negativa (**0.70**).

## 7. Conclusões

- **LMM(BERT) X COUNTVECTORIZER somente c/pré-processamento**

A comparação entre LLM (BERT) e CountVectorizer revela diferenças importantes em suas capacidades de detecção de sentimentos. O BERT se destacou em precisão e na identificação de sentimentos positivos, aproveitando seu treinamento prévio para entender contextos complexos e nuances linguísticas. No entanto, devido à sua robustez computacional, não conseguimos utilizar 100% do potencial do BERT, o que pode limitar sua eficácia em alguns cenários.

Por outro lado, o CountVectorizer, especialmente quando combinado com pré-processamento, mostrou-se mais eficaz na identificação de sentimentos negativos. Sua abordagem baseada em palavras-chave e frequências permite capturar diretamente a essência dos sentimentos negativos, resultando em um recall superior para essa classe. Enquanto o BERT pode enfrentar dificuldades ao interpretar ironias ou sutilezas, o CountVectorizer se beneficia de uma representação mais explícita dos dados.

- **LMM(BERT) X TD-IDF somente c/pré-processamento**

A comparação entre o modelo LLM (BERT) e o TF-IDF com pré-processamento revela que, embora o BERT seja mais eficaz na identificação de sentimentos positivos, o TF-IDF, quando combinado com um pré-processamento cuidadoso, se destaca na detecção de sentimentos negativos. Essa diferença de desempenho sugere que a escolha do método deve ser orientada pela natureza do problema em questão e pelas características específicas do conjunto de dados utilizado.

É importante notar que, assim como no caso do CountVectorizer, não conseguimos utilizar 100% do potencial do BERT devido às suas exigências computacionais, o que pode limitar sua eficácia em alguns contextos. Portanto, para análises que priorizam a identificação de sentimentos negativos, o TF-IDF pode ser uma escolha mais apropriada. Já para contextos que exigem uma compreensão mais profunda das sutilezas linguísticas, especialmente na detecção de sentimentos positivos, o BERT pode oferecer vantagens significativas, mesmo que não seja plenamente utilizado. Essa consideração é fundamental para otimizar a abordagem de análise de sentimentos de acordo com os objetivos específicos do estudo.

- **LMM(BERT) X TD-IDF COM STEMMING**

A comparação demonstra que, embora o LLM (BERT) seja mais eficaz na identificação de sentimentos positivos, o TF-IDF combinado com stemming se destaca na detecção de sentimentos negativos. Cada abordagem apresenta suas próprias vantagens, e a decisão entre elas deve levar em conta os objetivos da análise e as características específicas do conjunto de dados.

Essas considerações finais enfatizam a importância de selecionar a técnica mais adequada, visando otimizar os resultados da análise de sentimentos. Em última análise, a escolha deve refletir não apenas a natureza dos dados, mas também as necessidades específicas do projeto, garantindo assim uma compreensão mais robusta e precisa dos sentimentos expressos nas interações.

## 8. Próximos passos

Após a conclusão e da análise dos resultados, os próximos passos deste projeto devem se concentrar no aprimoramento da abordagem LLM (BERT). Sugere-se a realização de experimentos adicionais para ajustar hiperparâmetros e explorar outros modelos de LLM, como RoBERTa ou DistilBERT, que podem oferecer vantagens em termos de desempenho e eficiência computacional. A aplicação de validação cruzada é fundamental para garantir que os resultados obtidos sejam robustos e representativos. Além disso, a análise de erros deve ser aprofundada para compreender as limitações do BERT e identificar oportunidades de melhoria. A integração de técnicas de ensemble pode ser considerada para potencializar a precisão dos resultados