Classificação de Sentimentos em Análises de Filmes
Utilizando BERT e Validação Cruzada

Pedro Henrique Gurski de Oliveira - 20224759
Ulisses Curvello Ferreira - 20223829

22 de junho de 2025

Sumário

1 Introdução

A análise de sentimentos é uma das tarefas mais importantes em processamento de linguagem natural (NLP), com aplicações que vão desde monitoramento de redes sociais até análise de feedback de produtos. Este trabalho apresenta a implementação de um modelo de classificação de sentimentos utilizando BERT (Bidirectional Encoder Representations from Transformers) aplicado ao dataset IMDB de análises de filmes.

1.1 Problema

O objetivo deste trabalho é desenvolver um classificador capaz de determinar se uma análise de filme possui sentimento positivo ou negativo. Esta tarefa de classificação binária é fundamental para compreender a opinião dos usuários sobre produtos cinematográficos e pode ser estendida para outras aplicações comerciais.

1.2 Dataset IMDB

O dataset utilizado contém 50.000 análises de filmes extraídas do Internet Movie Database (IMDB), distribuídas igualmente entre sentimentos positivos e negativos. Para fins de experimentação e viabilidade computacional, foram realizados dois experimentos com diferentes configurações: o primeiro utilizando uma amostra de 5.000 exemplos e o segundo com 6.500 exemplos.

Link: `https://www.kaggle.com/datasets/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews`

Características do dataset:

- **Tamanho original:** 50.000 análises
- **Amostras utilizadas:** 5.000 (Experimento 1) e 6.500 (Experimento 2) análises
- **Classes:** Positivo (1) e Negativo (0)
- **Distribuição:** Balanceada (50% para cada classe)
- **Tipo de texto:** Análises de filmes em inglês de tamanho variável

2 Metodologia

2.1 Arquitetura do Modelo

Foi utilizado o modelo BERT-base-uncased como base para a classificação de sentimentos. O BERT é um modelo transformer bidirecional pré-treinado que demonstrou excelente performance em diversas tarefas de NLP.

Especificações do modelo:

- **Modelo base:** bert-base-uncased (12 camadas, 768 dimensões)
- **Número de labels:** 2 (classificação binária)
- **Comprimento máximo:** 128 tokens
- **Tokenizer:** BertTokenizer do Hugging Face

2.2 Pré-processamento

O pipeline de pré-processamento incluiu as seguintes etapas:

1. **Carregamento dos dados:** Utilização do kagglehub para download automático do dataset
2. **Mapeamento de labels:** Conversão de "positive"/"negative"para 1/0
3. **Amostragem:** Redução para 5.000 (Experimento 1) e 6.500 (Experimento 2)
4. **Tokenização:** Aplicação do tokenizer BERT com:
    - Truncamento para 128 tokens
    - Padding para comprimento máximo
    - Retorno de tensores PyTorch

2.3 Validação Cruzada

Foi implementada uma validação cruzada com 3 folds (3-fold cross-validation) para garantir a robustez dos resultados:

- **Estratégia:** K-Fold com k=3
- **Distribuição Exp. 1:** Aproximadamente 3.333 exemplos para treino e 1.667 para validação por fold
- **Distribuição Exp. 2:** Aproximadamente 4.333 exemplos para treino e 2.167 para validação por fold
- **Shuffle:** Ativado com seed fixo para reprodutibilidade
- **Métricas:** Accuracy, Precision, Recall e F1-Score

2.4 Configurações de Treinamento

Os hiperparâmetros utilizados foram:

Tabela 1: Hiperparâmetros de Treinamento

Parâmetro	**Experimento 1**	**Experimento 2**
Épocas	1	2
Batch size (treino)	4	16
Batch size (validação)	4	32
Tamanho da amostra	5.000	6.500
Otimizador	AdamW (padrão)	AdamW (padrão)
Learning rate	5e-5 (padrão)	5e-5 (padrão)
Seed	42	42
Device	CUDA	CUDA

3 Resultados e Análise

3.1 Métricas de Performance

Os resultados da validação cruzada com 3 folds demonstraram performance consistente do modelo em ambos os experimentos:

3.1.1 Experimento 1 (5.000 exemplos, 1 época)

Tabela 2: Resultados da Validação Cruzada - Experimento 1

Fold	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Fold 1	0.8452	0.8203	0.8747	0.8466
Fold 2	0.8296	0.8045	0.8927	0.8463
Fold 3	0.7989	0.7714	0.8468	0.8074
Média	**0.8246**	**0.7987**	**0.8714**	**0.8334**
Desvio Padrão	**±0.0192**	**±0.0204**	**±0.0189**	**±0.0184**

3.1.2 Experimento 2 (6.500 exemplos, 2 épocas)

Tabela 3: Resultados da Validação Cruzada - Experimento 2

Fold	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Fold 1	0.8629	0.8331	0.9114	0.8705
Fold 2	0.8763	0.8749	0.8781	0.8765
Fold 3	0.8657	0.8690	0.8666	0.8678
Média	**0.8683**	**0.8590**	**0.8854**	**0.8716**
Desvio Padrão	**±0.0058**	**±0.0185**	**±0.0190**	**±0.0036**

3.1.3 Comparação entre Experimentos

Tabela 4: Comparação de Performance entre Experimentos

Experimento	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Exp. 1 (5k, 1 época)	0.8246 ± 0.0192	0.7987 ± 0.0204	0.8714 ± 0.0189	0.8334 ± 0.0184
Exp. 2 (6.5k, 2 épocas)	0.8683 ± 0.0058	0.8590 ± 0.0185	0.8854 ± 0.0190	0.8716 ± 0.0036
Melhoria	**+4.37%**	**+6.03%**	**+1.40%**	**+3.82%**

3.2 Análise das Matrizes de Confusão - Experimento 1

As matrizes de confusão de cada fold do primeiro experimento revelam padrões importantes no comportamento do modelo:

3.2.1 Fold 1

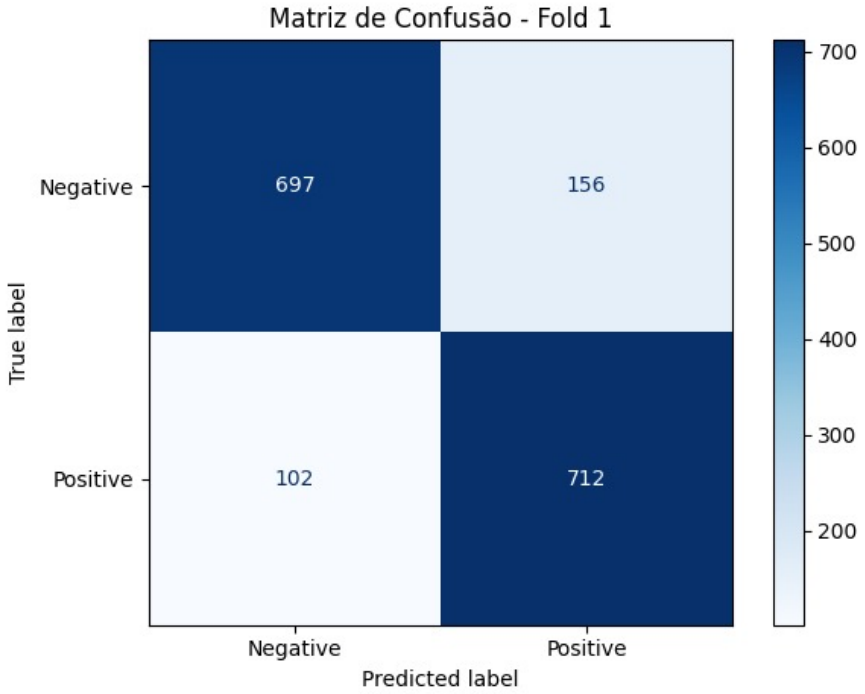

Figura 1: Matriz Confusão Fold 1 - Experimento 1

- **Verdadeiros Negativos:** 697
- **Falsos Positivos:** 156
- **Falsos Negativos:** 102
- **Verdadeiros Positivos:** 712
- **Taxa de Erro:** 15.48%

3.2.2 Fold 2

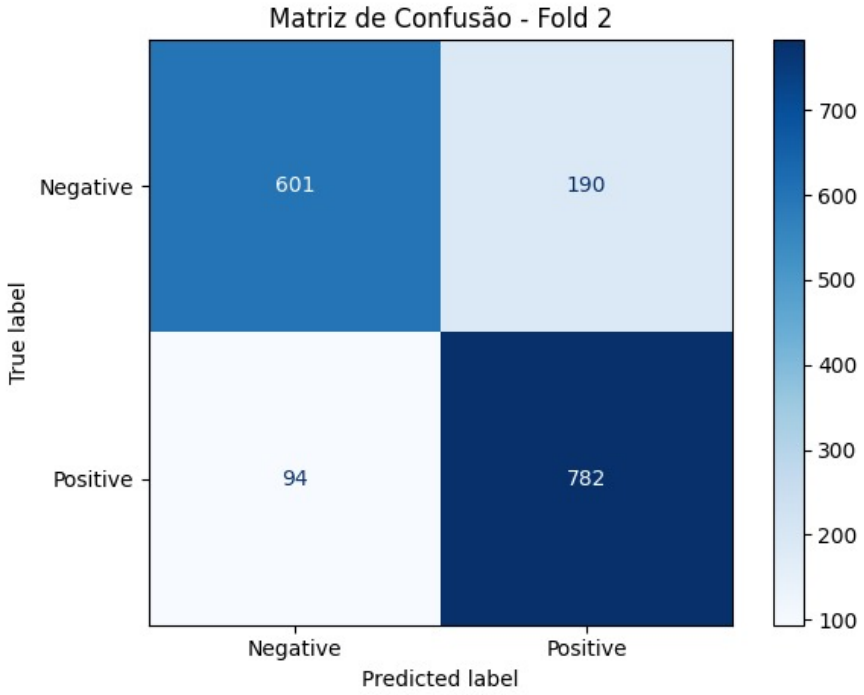

Figura 2: Matriz Confusão Fold 2 - Experimento 1

- **Verdadeiros Negativos:** 601
- **Falsos Positivos:** 190
- **Falsos Negativos:** 94
- **Verdadeiros Positivos:** 782
- **Taxa de Erro:** 17.04%

3.2.3 Fold 3

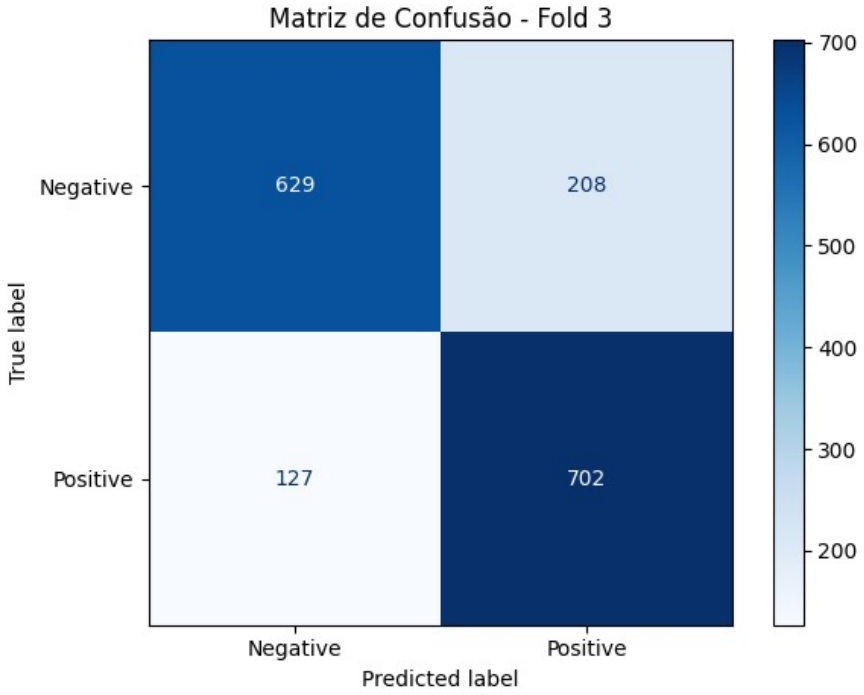

Figura 3: Matriz Confusão Fold 3 - Experimento 1

- **Verdadeiros Negativos:** 629
- **Falsos Positivos:** 208
- **Falsos Negativos:** 127
- **Verdadeiros Positivos:** 702
- **Taxa de Erro:** 20.11%

3.3 Análise do Training Loss - Experimento 2

O segundo experimento permitiu monitorar a evolução do training loss durante o treinamento:

3.3.1 Fold 1

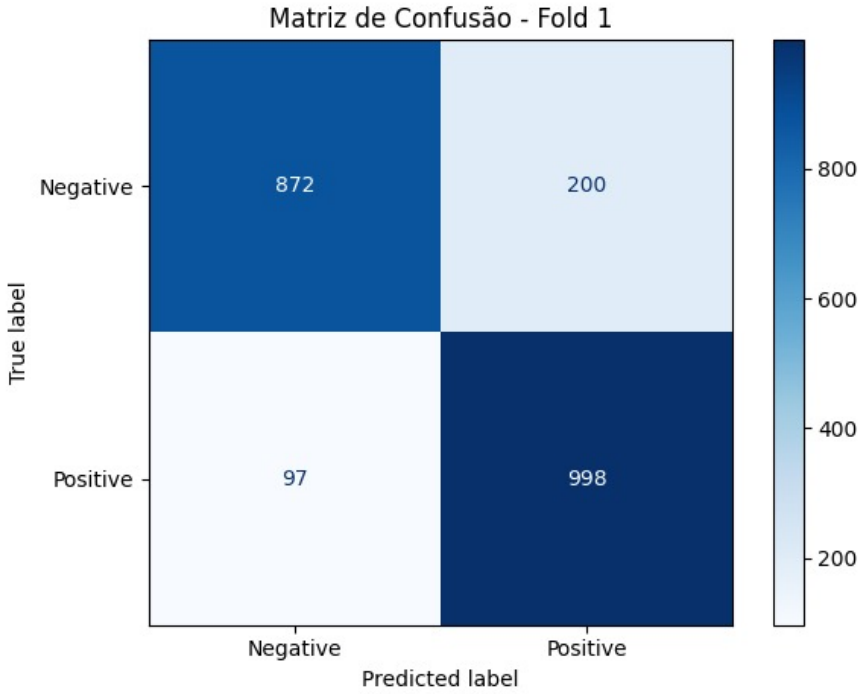

Figura 4: Matriz Confusão Fold 1 - Experimento 2

- **Verdadeiros Negativos:** 872
- **Falsos Positivos:** 200
- **Falsos Negativos:** 97
- **Verdadeiros Positivos:** 998
- **Taxa de Erro:** 13.71%

3.3.2 Fold 2

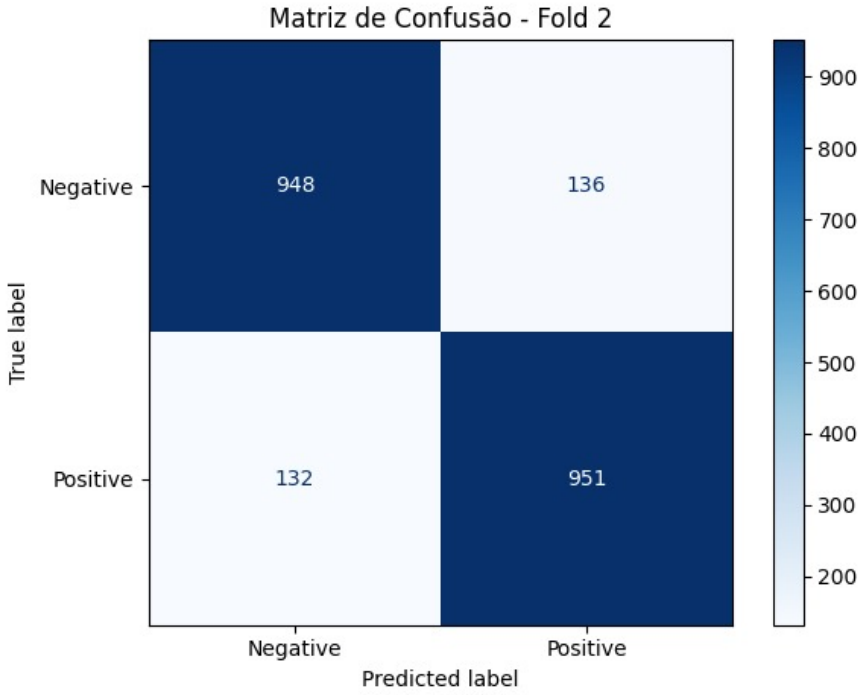

Figura 5: Matriz Confusão Fold 2 - Experimento 2

- **Verdadeiros Negativos:** 948
- **Falsos Positivos:** 136
- **Falsos Negativos:** 132
- **Verdadeiros Positivos:** 951
- **Taxa de Erro:** 12.37%

3.3.3 Fold 3

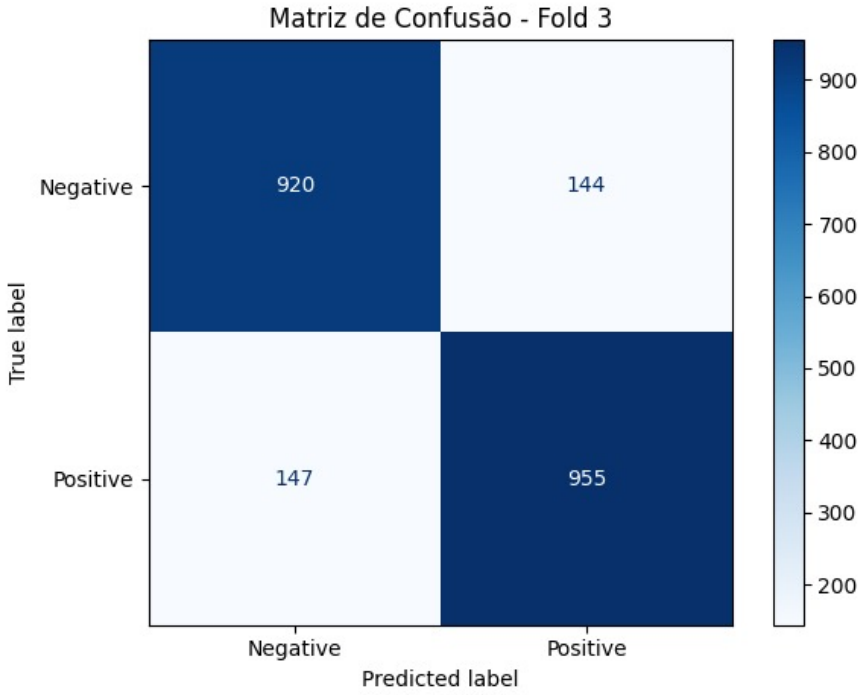

Figura 6: Matriz Confusão Fold 3 - Experimento 2

- **Verdadeiros Negativos:** 920
- **Falsos Positivos:** 144
- **Falsos Negativos:** 147
- **Verdadeiros Positivos:** 955
- **Taxa de Erro:** 13.43%

3.4 Interpretação dos Resultados

3.4.1 Performance Geral

O segundo experimento demonstrou melhoria significativa em todas as métricas comparado ao primeiro:

- **Accuracy:** Aumento de 82.46% para 86.83% (+4.37%)
- **F1-Score:** Melhoria de 83.34% para 87.16% (+3.82%)
- **Precision:** Elevação de 79.87% para 85.90% (+6.03%)
- **Estabilidade:** Redução significativa na variância entre folds
- **Tempo:** 5h 30min (Experimento 1) e 12h 40min (Experimento 2).

3.4.2 Impacto das Modificações

- **Mais dados (6.5k vs 5k):** Contribuiu para melhor generalização
- **Mais épocas (2 vs 1):** Permitiu convergência mais completa
- **Batch size maior:** Melhorou a estabilidade do treinamento
- **Training loss:** Demonstrou convergência adequada sem overfitting

3.4.3 Análise de Recall e Precision

- **Experimento 1:** Recall alto (87.14%) mas precision mais baixa (79.87%)
- **Experimento 2:** Melhor balanceamento entre recall (88.54%) e precision (85.90%)
- **Viés positivo:** Reduzido no segundo experimento devido ao maior treinamento

3.4.4 Consistência entre Folds

O segundo experimento mostrou maior estabilidade com desvios padrão menores:

- **Accuracy:** ± 0.0058 vs ± 0.0192 (melhoria de 70%)
- **F1-Score:** ± 0.0036 vs ± 0.0184 (melhoria de 80%)

4 Discussão

Durante o desenvolvimento deste trabalho, diversos desafios técnicos foram enfrentados, levando à realização de dois experimentos com configurações distintas para avaliar o impacto das modificações implementadas.

4.1 Desafios Técnicos Enfrentados

As limitações computacionais exigiram adaptações significativas no primeiro experimento, como a redução do batch size para 4 devido à restrição de memória da GPU, e o uso de apenas 5.000 exemplos do dataset original. O tempo de treinamento prolongado (mais de 5 horas) e as interrupções recorrentes no Google Colab dificultaram experimentos mais longos do que o segundo experimento (mais de 13 horas).

Outro ponto crítico envolveu a compatibilidade entre bibliotecas, sendo necessário adaptar o código para versões anteriores do pacote transformers, desativar o WandB e ajustar argumentos de treinamento. A instabilidade do ambiente, exigindo reinstalação de bibliotecas a cada reinicialização, resultou em perda de produtividade.

5 Conclusão

Este trabalho demonstrou a aplicação bem-sucedida do modelo BERT para classificação de sentimentos no dataset IMDB, com dois experimentos que evidenciaram o impacto positivo de ajustes metodológicos e computacionais.

Experimentos futuros poderiam explorar classificações mais granulares de sentimentos, uso de sequências mais longas (até 512 tokens), técnicas de ensemble, aumento de dados via back-translation, e estratégias de regularização como dropout e weight decay. A implementação de fine-tuning progressivo também poderia melhorar ainda mais os resultados.