

Instructions for Authors of SBC Conferences

Papers and Abstracts

1

Abstract. *This meta-paper describes the style to be used in articles and short papers for SBC conferences. For papers in English, you should add just an abstract while for the papers in Portuguese, we also ask for an abstract in Portuguese (“resumo”). In both cases, abstracts should not have more than 10 lines and must be in the first page of the paper.*

Resumo. *Este meta-artigo descreve o estilo a ser usado na confecção de artigos e resumos de artigos para publicação nos anais das conferências organizadas pela SBC. É solicitada a escrita de resumo e abstract apenas para os artigos escritos em português. Artigos em inglês deverão apresentar apenas abstract. Nos dois casos, o autor deve tomar cuidado para que o resumo (e o abstract) não ultrapassem 10 linhas cada, sendo que ambos devem estar na primeira página do artigo.*

1. Introdução

2. Trabalhos Relacionados

3. Material e Métodos

A metodologia fundamenta-se na premissa de que o viés ideológico se manifesta em padrões discursivos e semântico recorrentes. A abordagem proposta processa o conteúdo textual de notícias utilizando modelos pré-treinados para a geração de *embeddings*, os quais são otimizados via *fine-tuning* com as funções de perda *Contrastive Loss* e *Triplet Loss*. Esse refinamento visa maximizar orientações políticas, servindo como entrada para um classificador de aprendizado de máquina.

Conforme ilustrado na Figura 1, o fluxo de trabalho compreende quatro etapas principais:

- **Conjunto de treino e teste:** Coleta baseada em conjunto de dados da literatura correlata;
- **Geração de *embeddings*:** *Fine-tuning* de modelos pré-treinados com aprendizagem métrica (*Contrastive* e *Triplet Loss*);
- **Classificação:** Treinamento do modelo de classificação sobre os vetores otimizados;
- **Avaliação da abordagem:** análise do desempenho do sistema e dos resultados obtidos.

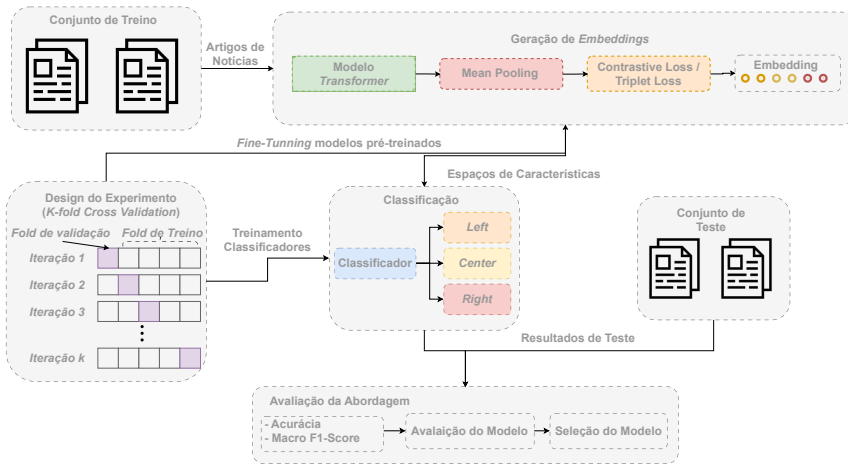


Figura 1. Visão geral do método de detecção de viés ideológico por meio do conteúdo textual de artigos de notícias.

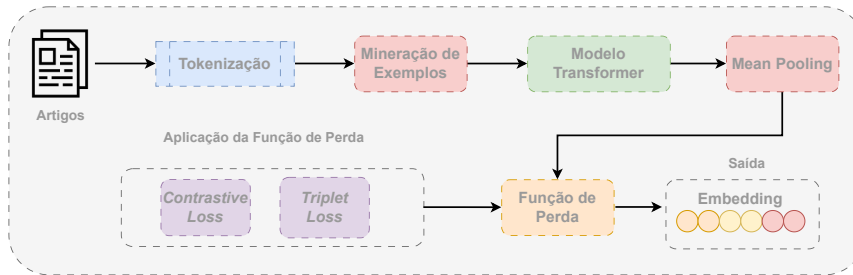


Figura 2. Fluxo de treinamento e geração de características para artigos de notícias.

3.1. Dados Experimentais

3.2. Tarefa de Geração de *Embeddings*

Para a execução da tarefa, a Figura 2 ilustra o fluxo de processamento adotado. O processo inicia-se com a *tokenização* dos artigos de notícias, seguida pela mineração de exemplos. Esta etapa é fundamental para selecionar amostras informativas que otimizam a convergência e o aprendizado do modelo.

Conforme delineado na arquitetura apresentada, empregaram-se dois modelos fundamentados em *Bidirectional Encoder Representations from Transformers (BERT)* [Devlin2018], reconhecidos pela eficácia na modelagem de dependências de longo alcance e na extração de relações semânticas granulares [Gao2021]. A seleção recaiu sobre o DistilBERT [Sanh2019] e o DistilRoBERTa [Liu2019], variantes destiladas que preservam a robustez das arquiteturas originais, contudo, apresentam reduções substanciais no custo computacional e nos requisitos de memória.

O modelo *Transformer* processa as sequências de entrada, seguido por uma camada de *Mean Pooling* que consolida as representações em um vetor único. O (*fine-tuning*) é regido por estratégias de aprendizagem métrica (*metric learning*¹), utilizando as funções de perda *Contrastive Loss* ou *Triplet Loss*. Tal abordagem assegura que os *em-*

¹aprendizagem métrica

beddings gerados na saída posicionem instâncias contextualmente similares em regiões próximas do espaço de representação, otimizando a discriminação entre as classes.

No que se refere ao pré-processamento, as *stopwords* foram preservadas, visto que a arquitetura BERT demonstra eficácia na extração de nuances contextuais a partir desses elementos. Por fim, o treinamento foi estabelecido com um limite de 100 épocas, utilizando o otimizador Adam com taxa de aprendizado de 0,0001 e *batch size* de 16. Para mitigar o *overfitting* e assegurar a capacidade de generalização dos modelos, aplicou-se a técnica de *Early Stopping* com paciência de 30 ciclos, monitorando-se a convergência da função de perda no conjunto de validação.

3.2.1. Mecanismos de Aproximação e Distanciamento

Nesta abordagem, os codificadores (DistilBERT e DistilRoBERTa) ajustam os pesos de suas camadas para otimizar a qualidade dos *embeddings* via *Contrastive Loss* e *Triplet Loss*. O objetivo é o aprendizado de representações vetoriais onde instâncias semanticamente similares converjam no espaço de representação, enquanto exemplos dissimilares sejam repelidos.

A *Contrastive Loss* é aplicada utilizando a distância Euclidiana sobre pares de exemplos, conforme definido na Equação 1:

$$L = \frac{1}{2}(1 - y)D^2 + \frac{1}{2}y\{\max(0, m - D)\}^2 \quad (1)$$

Onde y representa o rótulo binário (0 para similar, 1 para dissimilar), D denota a distância entre as representações e m é a margem de separação.

Complementarmente, a *Triplet Loss* utiliza triplas compostas por uma âncora (a), um exemplo positivo (p) e um negativo (n). O objetivo, expresso na Equação 2, assegura que a distância entre a âncora e o positivo seja inferior à distância entre a âncora e o negativo por uma margem m :

$$L = \max(0, D(a, p) - D(a, n) + m) \quad (2)$$

Para otimizar o aprendizado, empregou-se o *mining* de negativos *semi-hard*. Esses exemplos, que satisfazem a condição $D(a, p) < D(a, n) + m$, fornecem gradientes mais informativos e mitigam o *overfitting* em comparação a negativos *hard* [kertez,2021]. Esse processo refina a capacidade discriminatória do modelo, permitindo que os *embeddings* capturem relações semânticas profundas, como a ideologia de uma notícia, independentemente da fonte de publicação.

3.3. Tarefa de Classificação: Modelos e Parametrização

Após o mapeamento dos *embeddings*, onde a proximidade entre os vetores reflete a similaridade ideológica das notícias. A classificação dos artigos foi realizada por meio de três algoritmos: *K-Nearest Neighbors (KNN)*, *K-Means* e *Multilayer Perceptron (MLP)*. O *KNN* e o *K-Means* foram utilizados para explorar a organização dos dados por vizinhança e agrupamento, respectivamente. Para o *KNN*, aplicou-se um *grid search* sistemático para

otimização de hiperparâmetros, variando o número de vizinhos (k) entre 5, 10, 15, 20, 25 e 30. Já o *K-Means* foi configurado com o número de *clusters* equivalente às classes presentes no conjunto de dados ABP.

A rede *MLP* foi estruturada com duas camadas densas (512 e 256 neurônios). Adotou-se a função de ativação *ReLU* para garantir um treinamento mais rápido e estável [REFERENCIA], enquanto a camada de saída utilizou a *softmax* para a classificação final. O modelo otimizado com o algoritmo *Adam* [REFERENCIA] e a função de perda *Categorical Cross-Entropy*, escolhas consolidadas na literatura para problemas multiclasse [REFERENCIA GOODFELLOW].

A confiabilidade do experimento foi assegurada pela validacruzada estratificada (5-fold). Esse procedimento garante que a proporção das classes seja mantida em todas as etapas, evitando resultados enviesados e permitindo medir com precisão a capacidade do modelo em classificar novos dados [REFERENCIA]

3.4. Avaliação de Desempenho

O desempenho dos modelos de classificação propostos será avaliado quantitativamente através das métricas de Acurácia e *Macro F1-score*. A escolha de tais métricas fundamenta-se na necessidade de uma análise robusta: enquanto a acurácia fornece uma medida geral da taxa de acerto do modelo (Equação 3), o *Macro F1-score* permite uma avaliação criteriosa da capacidade preditiva em todas as classes, mitigando distorções causadas por eventuais desbalanceamentos no conjunto de dados (Equação 4).

As métricas são formalmente definidas como segue:

$$\text{Acurácia} = \frac{1}{|C|} \sum_{c \in C} \frac{TP_c + TN_c}{TP_c + TN_c + FP_c + FN_c} \quad (3)$$

$$\text{Macro } F1 = \frac{1}{N} \sum_{i=1}^N F1_i \quad (4)$$

Onde o $F1$ de cada classe é a média harmônica entre a Precisão e a Revocação, definidas nas Equações 5 e 6:

$$\text{Precisão} = \frac{TP}{TP + FP}, \quad \text{Revocação} = \frac{TP}{TP + FN} \quad (5)$$

$$F1 = \frac{2 \times (\text{Precisão} \times \text{Revocação})}{\text{Precisão} + \text{Revocação}} \quad (6)$$

Neste contexto, TP , TN , FP e FN representam, respectivamente, os verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. N indica o número total de categorias ideológicas.

A validação da hipótese de pesquisa — de que o discurso textual reflete o viés ideológico — dar-se-á mediante a obtenção de altos índices em ambas as métricas. Espera-se que valores elevados de *Macro F1-score* confirmem que o modelo possui alta

capacidade discriminatória entre as diferentes vertentes ideológicas, garantindo que o desempenho não seja fruto de uma tendência majoritária no conjunto de dados.

4. Resultados e Discussão

A linguagem Python, com as bibliotecas Numpy, Pandas, Scikit-Learn e PyTorch, foi a ferramenta primária para implementação e avaliação dos modelos. Os experimentos ocorreram em um servidor com processador Intel Xeon W-2235, 128 GB de RAM e GPU NVIDIA RTX 8000 (48 GB VRAM), visando a aceleração em hardware. Para garantir a reprodutibilidade, o código-fonte, hiperparâmetros e scripts de pré-processamento estão disponíveis em: <https://github.com/jailsonpj/detecting-ideological-bias>.

5. Considerações Finais

6. References

Bibliographic references must be unambiguous and uniform. We recommend giving the author names references in brackets, e.g. [Knuth 1984], [Boulic and Renault 1991], and [Smith and Jones 1999].

The references must be listed using 12 point font size, with 6 points of space before each reference. The first line of each reference should not be indented, while the subsequent should be indented by 0.5 cm.

Referências

Boulic, R. and Renault, O. (1991). 3d hierarchies for animation. In Magnenat-Thalmann, N. and Thalmann, D., editors, *New Trends in Animation and Visualization*. John Wiley & Sons Ltd.

Knuth, D. E. (1984). *The T_EX Book*. Addison-Wesley, 15th edition.

Smith, A. and Jones, B. (1999). On the complexity of computing. In Smith-Jones, A. B., editor, *Advances in Computer Science*, pages 555–566. Publishing Press.