An NLP Approach to Quantifying Bias Magnitude
W266 Final Report

Raj Jagannath, David Lee, Vincent Qu
{rjagan, dlee2455, vincentqu}@ischool.berkeley.edu

Abstract

Media bias in news articles can subtly influence public opinion, contributing to misinformation and societal polarization [1]. The rise of social media platforms has globally expanded the reach of news, enabling biased information to disseminate rapidly and widely, often before thorough fact-checking or analysis has occurred. This unchecked spread of bias can result in dangerous viewpoints on complex and important issues. To address these challenges, our project develops a Bias Score model that detects and quantifies bias in news articles using Natural Language Processing (NLP) techniques. We implement a logistic regression baseline, XGBoost model, a RoBERTa-based model enhanced with ConvBERT, a DistilBERT classifier, and an ELECTRA classifier. By analyzing sentiment, stance, and specific bias indicators within the text, these models are able to classify articles as biased or unbiased. Utilizing the Bias Annotations By Experts (BABE) dataset [2], we train and validate our models to enhance bias detection accuracy. Our main objective is to determine the presence of bias in articles, thereby empowering readers to critically assess the content they consume.

1. Introduction

In today's digital age, the landscape of news dissemination has undergone profound refinement and transformation primarily driven by the rise of social media platforms. These platforms have enabled people to share information at an unprecedented rate, allowing them to reach global audiences faster than ever before. While the democratization of information has numerous benefits, it also presents a plethora of challenges, particularly concerning the integrity and impartiality of the content being shared.

Media bias in online news articles is a critical issue that can subtly influence public perception, shaping opinions and contributing to the spread of misinformation. Biased reporting can distort facts, emphasize particular viewpoints, and purposely omit essential information, thereby influencing readers' perceptions and understanding of events. The rise in biased content, especially on platforms where information spreads quickly, exacerbates societal polarization and undermines informed public discourse.

To address these challenges, our project focuses on developing a Bias Score model that detects and quantifies bias in news articles using advanced Natural Language Processing (NLP) techniques. By implementing and comparing various machine learning algorithms– including logistic regression, XGBoost, RoBERTa-based model enhanced with ConvBERT, DistillBERT, and an ELECTRA classifier– with the goal of creating a robust system capable of accurately identifying bias in textual content. Our group aims to do this by leveraging the Bias Annotations By Experts (BABE) dataset [2]—carefully curated by domain experts for media bias research— our models analyze sentiment, stance, and specific bias indicators within the text to classify articles as biased or unbiased thereby empowering readers to critically evaluate the content they consume.

This report outlines our approach to bias detection, detailing the methodologies employed, the performance of baseline models, and the enhancements made to improve accuracy. Through this work, we aim to contribute to the field of media analysis by providing tools that can systematically identify bias, thereby supporting efforts to promote balanced and objective journalism.

2. Background

Media bias refers to the prejudice or inclination of journalists and news organizations in selecting and presenting information. This bias manifests itself through the selection of stories, the framing of issues, and the language used in reporting [3]. Such biases can influence public perception by highlighting certain aspects of a story while intentionally neglecting others, thereby shaping the narrative in a particular direction. Studies have shown that media bias can significantly affect public opinion and contribute to societal polarization; for instance, Gentzkow and Shapiro found that media bias significantly affects political polarization in the United States, with biased reporting contributing to diverging beliefs among people of different demographic groups [4]. This phenomenon is further amplified by social media platforms, where algorithm-driven content delivery often creates echo chambers, limiting diverse information exposure and opposing perspectives [9].

The significance of media bias has been measurable and noticeable especially in recent years. According to a Pew Research Survey, 62% of Americans believe that news organizations tend to favor one side politically, exacerbating partisan polarization. During the COVID-19 pandemic, biased reporting on vaccine efficacy and public health measures contributed to widespread misinformation and resistance to health guidelines, as documented by the World Health Organization [6]. Moreover, biased news coverage during the elections has shown to influence voter behavior and turnout, thus highlighting candidates in different ways inadvertently bringing awareness to media impartiality on democratic processes.

As a result, media bias detection has emerged as a crucial area of research within Natural Language Processing (NLP). Traditional approaches to bias detection often rely on manual annotation and subjective analysis, which are time-consuming and susceptible to human error. To overcome these limitations, researchers have explored various machine learning models, including both traditional classifiers and deep learning architectures.

Models such as logistic regression and Support Vector Machines (SVM) have been employed for bias detection by using handcrafted features derived from the text [8]. While these models can achieve reasonable performance, their effectiveness is limited by the quality and comprehensiveness of the feature engineering process. Deep learning models on the other hand, like BERT (Bidirectional Encoder Representations from Transformers) and RoBERTa (A Robustly Optimized BERT Pretraining Approach), have shown significant promise in capturing contextual nuances and complex language patterns, leading to improved accuracy in bias detection tasks [8]. However, these models require substantial computational resources and large annotated datasets for effective training. Despite advancements, existing models often struggle with contextual understanding and the subtlety of biased language, resulting in inconsistent accuracy rates. The scarcity of high-quality, annotated datasets for media bias hampers the development of more accurate and reliable models.

Several studies have laid the groundwork for automated media bias detection using NLP techniques, providing both methodological innovations and valuable datasets that have greatly influenced subsequent research in this domain. Entman introduced the concept of framing bias, emphasizing how media outlets can influence public perception by selecting and highlighting specific aspects of a story [3]. Gentzkow and Shapiro provided empirical evidence linking media bias to political polarization, illustrating the societal impact of biased reporting [4]. In the realm of NLP, Chung et al. conducted a comprehensive survey on bias and fairness in machine learning, highlighting the potential of NLP techniques in mitigating bias [7]. Allcott and Gentzkow examined the role of social media in spreading fake news, underscoring the need for a robust bias detection mechanism [9]. Spinde et al. emphasize the critical role word choice plays in bias detection, demonstrating how subtle linguistic variations can significantly influence the perceived bias of a text [2]. Leveraging distant supervision techniques, Spinde et al. developed a neural network based model that effectively identifies biased language patterns, achieving superior performance compared to traditional machine learning approaches.

Overall, these studies collectively underscore the importance of developing advanced models that can accurately detect and quantify media bias, paving the way for our project's focus on creating a robust Bias Score model using enhanced NLP techniques.

3. Methods

3.1 Dataset and Data Processing

The primary dataset employed in this project is sourced from the Neural Media Bias Detection Using Distant Supervision With BABE - Bias Annotations By Experts repository [2]. Specifically, the SG2 subset, comprising 3,700 sentences annotated for media bias curated by domain experts. The dataset comprises a diverse collection of news articles annotated for various bias indicators, including sentiment, stance, and specific linguistic cues. BABE provides a robust foundation for developing machine learning models capable of accurately identifying and quantifying bias in textual content.

A general outline of how the dataset was collected and created is highlighted in Figure 1. The study collected 3,700 sentences from news articles covering 12 controversial topics, with a focus on US media as they found articles surrounding the topic had increasing political polarization. The sentences were extracted from left-wing, center, and right-wing news outlets selected based on the media bias chart provided by Allsides. The collection process involved defining keywords for each topic, specifying the news outlets and time frame, and manually inspecting the list of articles to extract sentences based on media bias annotation guidelines.

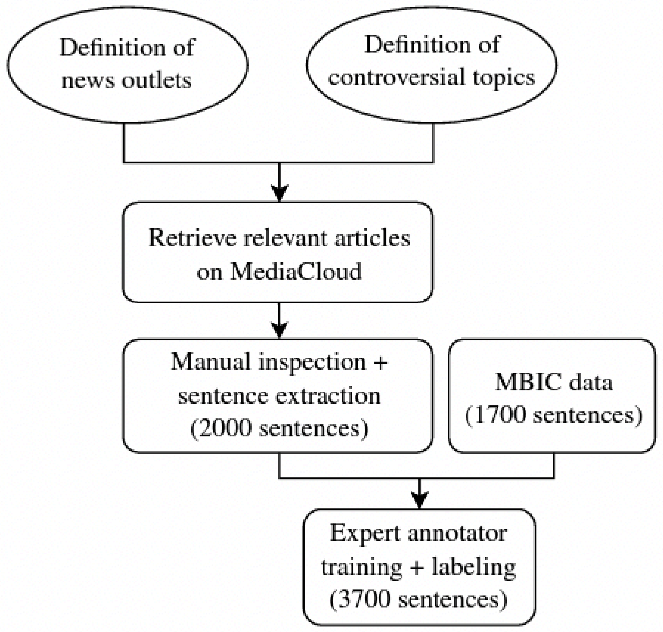
Definition of news outlets
Definition of controversial topics
Retrieve relevant articles on MediaCloud
Manual inspection + sentence extraction (2000 sentences)
MBIC data (1700 sentences)
Expert annotator training + labeling (3700 sentences)

Figure 1: Data collection and annotation pipeline

For annotations, experts are defined as someone with at least six months of experience in the media bias domain and sufficient training to identify biased wording, distinguish between bias and plain polarizing language, and maintain a politically neutral viewpoint while annotating. Annotators were given refined annotation guidelines to provide a clear criteria. The annotators had to provide basic reasoning about their annotation decisions during weekly discussion sessions, and their annotations were evaluated and discussed as a group. Annotations of one annotator were discarded based on this method.

The resulting dataset includes the sample sentence, article URL, news outlet, topic, ideology, bias or not bias label, label opinion, and biased words.

3.2 Logistic Regression (Baseline)

Establishing a robust baseline model is integral for evaluating the effectiveness for more complex algorithms in media bias detection. In this study, Logistic Regression (LR) is leveraged as the baseline classifier due to its recognition as a fundamental machine learning algorithm widely used in classification tasks due to its simplicity, interpretability, and effectiveness [10]. These characteristics make Logistic Regression an ideal starting point, allowing for straightforward implementation and providing an achievable benchmark against which the performance of advanced models, such as transformer-based architectures, can be compared.

In the context of media bias detection, LR facilitates the identification of key linguistic features that contribute to bias classification, thereby offering insights into the underlying patterns present in biased versus non-biased content. By leveraging features such as sentiment scores, stance indicators, and specific bias-related linguistic cures, the LR model establishes a reference performance level. This foundation is crucial for ensuring consistency with established methodologies and enabling meaningful performance comparisons with more sophisticated models like the ones outlined below.

The performance of the LR baseline will be evaluated using standard classification metrics including accuracy, accuracy, precision, recall, and F1-score. These metrics provide a comprehensive understanding of the model's ability to correctly identify based on unbiased articles, forming a critical reference point for assessing the effectiveness of advanced NLP techniques employed in subsequent models.

3.4 Data Preprocessing and Feature Extraction

To ensure the model's ability to detect nuanced linguistic patterns that may indicate bias, effective data preprocessing was performed. The preprocessing pipeline outlined below is specifically designed to optimize the performance of the baseline Logistic Regression model.

The SG2 subset from the BABE repository [2] consists of 3,700 sentences extracted from news articles and annotated for media bias by five expert annotators. Parameters were configured to ensure that delimiters and parsing inconsistencies were handled appropriately ensuring data integrity.

An initial analysis revealed missing entries in several columns, including news_link, type, label_bias, and label_opinion. Specifically, there are 35 missing entries in news_link, 997 in type, and 3 each in label_bias and label_opinion. To address these, rows with missing label_bias or label_opinion are retained and encoded as a separate class (None) to preserve dataset completeness. Given the substantial number of missing entries in the type column, this feature is excluded from the feature set to prevent introducing noise or bias into the model.

The categorical target variable label_bias was transformed into numerical values using LabelEncoder, facilitating the application of machine learning algorithms that require numerical input. The encoding scheme maps 'Biased' to 0, 'No agreement' to 1, 'Non-biased' to 2, and 'None' to 3. A series of processing steps were conducted to enhance the feature quality of the data; special characters were removed to reduce noise, all text was converted to lowercase to ensure uniformity, excessive whitespace was eliminated to standardize text formatting, and common English stopwords were removed to retain meaningful words that contribute to bias detection. The preprocessing was implemented through a custom 'preprocess_text' function applied to the text column, resulting in a new cleaned_text column.

For feature extraction, Term Frequency-Inverse Document Frequency (TF-IDF) vectorization is employed to convert the cleaned text into numerical features. TF-IDF effectively captures the importance of words within the corpus relative to their frequency, allowing the model to focus on significant terms that contribute to bias classification. There were a maximum of 5,000 features selected to balance computational efficiency and model performance.

3.5 Model Training and Evaluation

The Logistic Regression model was trained and evaluated through a systematic approach tailored for this baseline classifier. The dataset was partitioned into training and testing subsets using a 80-20 split, ensuring that the model is trained on a substantial portion of the data while retaining an unbiased evaluation set. A fixed random state was used to guarantee reproducibility. Additionally, a Logistic Regression classifier was instantiated with a maximum iteration parameter of 1000 to ensure convergence, especially given the high dimensionality of the TF-IDF feature space. The model was trained to learn the relationship between the TF-IDF features and the encoded bias labels.

Although Logistic Regression has relatively few hyperparameters, optimizing these can enhance model performance. In this study, the regularization strength and penalty type were tuned using grid search with cross-validation. The grid search was performed over a predefined parameter grid to identify the combination that yields the best cross-validated performance. The trained model generates the bias predictions on the test subset and the performance was assessed using accuracy and a classification report, which provides the precision, recall, and F1-score for each bias class. These evaluation metrics establish the baseline performance, facilitating comparative analyses with more advanced models.

To gain more insight in the factors influencing performance, a Random Forest (RF) classifier is employed to assess feature importance. The top 10 most significant features were identified and visualized, highlighting key terms that contributed to bias detection. The analysis complements the Logistic Regression baseline by identifying specific linguistic cues that are more indicative of media bias.

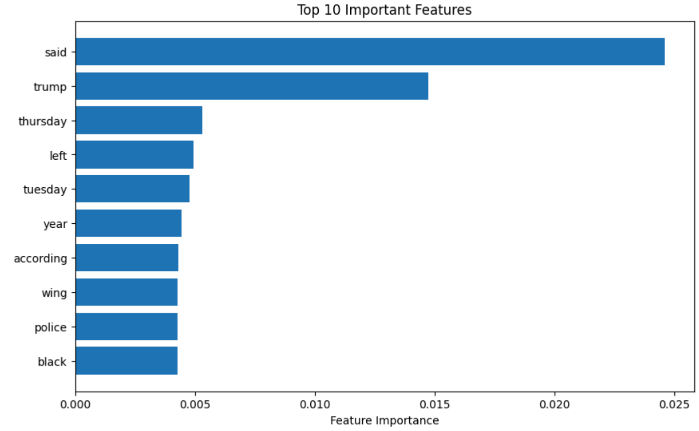

Figure 2: Top 10 Important Features Identified by the Random Forest Classifier

3.6 Baseline Results

The Logistic Regression baseline model achieved an accuracy of 75%, indicating that the model correctly classified 75% of the instances in the testing set. The classification report provides a detailed breakdown of the model's performance across different bias categories:

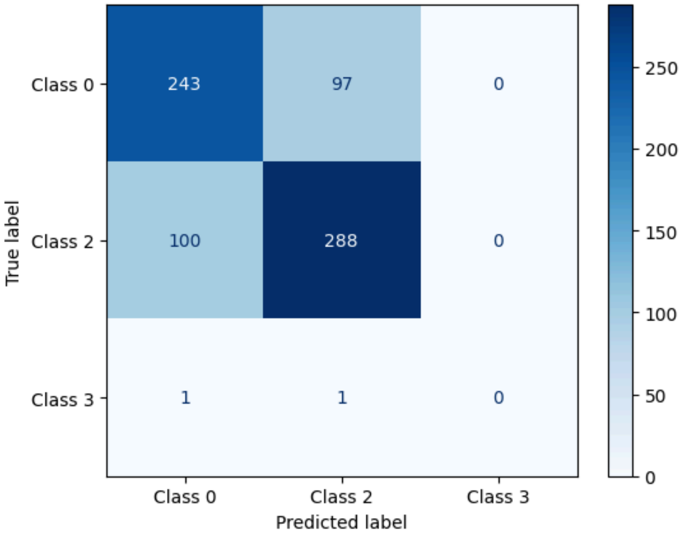

Figure 3: Top 10 Important Features Identified by the Random Forest Classifier

3.7 XGBoost

Extreme Gradient Boosting otherwise known as XGBoost is highly efficient and a scalable implementation of gradient boosting framework, which is acknowledged for superior performance in classification tasks [5]. In our study, we utilized XGBoost to classify texts as Non-Biased or Biased, aiming to leverage its ability to handle large feature spaces and prevent overfitting through regularization. To optimize the model's performance, we conducted hyperparameter tuning using Grid Search with cross-validation, adjusting several parameters. Training with a max depth of 3, and learning rate of 0.3 the optimized model yielded an accuracy of approximately 71.29% on the test set. The classification report indicated a balanced performance across both classes, with a precision of 0.74 and recall of 0.75 for the Biased category, and a precision of 0.69 and recall of 0.68 for the Non-biased category. These results demonstrate XGBoost's effectiveness in accurately identifying media bias, offering a robust alternative to traditional logistic regression models.

3.8 RoBERTa

RoBERTa is a powerful tool for NLP tasks, including sentiment analysis and text classification [14]. In the context of understanding bias in news article sentences, we felt the RoBERTa model would do well at analyzing the language used in the sentences and identify patterns or features that may indicate bias.

We wanted to leverage RoBERTa's contextualized embeddings capability to better understand the representation of each word in a sentence taking into account the context in which it appears. Bias in political news articles may have more subtle nuances in the language choice. In addition, RoBERTa was pre-trained on a large vocabulary which we felt was important for news articles where the author may pick highly contextualized words, sometimes even made up, to get their point across. This combination led us to believe that the RoBERTa model would be a good fit to identify bias that could be further tuned on the dataset we provide.

Although our dataset was on the smaller side given the difficulty of the task, our limited resources when it comes to computing power led us to tweaking the parameters to optimize for model efficiency. Specifically, we set the batch size to 32 and trained the model for 2 epochs. Learning rate was set to 5e-5 and weight decay of 0.01. These hyperparameters were chosen based on a combination of grid search and random search, and were found to provide the best trade-off between model performance and computational efficiency.

3.9 ConvBERT

ConvBERT (Convolutional BERT) is a modification of the BERT architecture which adds convolutional layers and introduces a mixed-attention block in place of the self-attention block used in BERT. The convolutional layer is much more computationally and memory efficient than the self-attention mechanism of BERT. Experiments have shown that ConvBERT achieves improved performance over BERT while training more quickly and with better efficiency [11].

In contrast with the RoBERTa model, due to the relative ease in training the ConvBERT model, we were able to use a smaller batch size of 4 and trained the model for 2 epochs using the same learning rate of 5e-5 and weight decay of 0.01. We found these parameters to be a good tradeoff between performance and efficiency.

3.10 DistilBERT

DistilBERT (Distilled BERT) is a lighter and faster version of BERT. It uses knowledge distillation, where a smaller model is trained to mimic the larger model. As a result, it is able to perform similarly to BERT while using fewer parameters and less computational power. Experiments have shown that while using 60% of the parameters of BERT, it achieves 97% of BERT's performance [12].

Because it is cheaper and easier to train than BERT, we were able to use a batch size of 4 and trained the model for 2 epochs and used the same learning rate of 5e-5 and weight decay of 0.01.

3.11 ELECTRA

ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) is a Transformer model that uses replaced token detection instead of the masked language model used in BERT. The model is pre-trained using a generator model to replace some tokens with alternatives and then a discriminator model to predict whether each original token was replaced or not. Experiments have shown that this pre-training task is more efficient than the masked language model because the task is defined over all input tokens rather than just the small subset that was masked out [13]. The result is that the model trains much more quickly and efficiently than BERT and provides similar performance to RoBERTa with 25% of the compute power.

The increased efficiency in comparison to BERT allowed us to use a batch size of 4 and train the model for 2 epochs with a learning rate of 5e-5 and weight decay of 0.01.

4. Results and Discussion

4.1 Model Summary Table

Model	Accuracy	Recall	F1 Score
Logistic Regression	72.7%	0.73	0.73
XGBoost	72.1%	0.72	0.72
RoBERTa	81.7%	0.81	0.81
ConvBERT	81.1%	0.81	0.81
DistilBERT	79.3%	0.79	0.79
ELECTRA	81.3%	0.81	0.81

4.2 Results Summary

The logistic regression and XGBoost models performed similarly and resulted in accuracies of about 72% and recall and F1 scores of about 0.72. The transformer-based models RoBERTa, ConvBERT, DistilBERT, and ELECTRA performed similarly to each other and provided a superior accuracy of about 80%, recall, and F1 score of about 0.80 compared to the baseline.

We were not surprised that the four transformer-based models performed so similarly, since we realized that due to the relatively small dataset and limited compute capacity that we had, it was likely that the models would have such close results. It was reasonable that of the four, DistilBERT would perform slightly worse than the other three due to the tradeoff of its lower performance and faster training compared to BERT.

It is possible that given better compute capacity, we could have improved the performance of the transformer models by training for more epochs, and we experimented with different hyperparameters, including batch size, learning rate, and epoch size before reaching our final results.

5. Conclusion

The results of our experiments demonstrate a striking similarity in performance among various HuggingFace transformers and other NLP models. This phenomenon can likely be attributed to the fact that all these models are designed to capture the underlying patterns and relationships in text. The transformer-based models, such as BERT and RoBERTa, have been pre-trained on large corpora of text and fine-tuned on our dataset, allowing them to learn contextualized representations of words and phrases. It is also possible that the sample data we used to fine-tune the models was small and limited the models' ability to learn and generalize to new, unseen data. Furthermore, this result highlights the importance of pre-training and fine-tuning in achieving high performance in NLP tasks, even with limited sample data. Our findings may suggest that the choice of model may not be as crucial as previously thought, and that a range of models can achieve similar performance on NLP tasks, even with limited sample data.

Limitations:

- Dataset Size: The study was constrained by a relatively small dataset, which may limit the generalizability of the findings. Larger datasets could provide more robust evaluations and model performance.
- Class Imbalance: Although techniques like class weighting, and oversampling was employed, inherent class imbalances in the dataset may have still affected model performance and bias detection accuracy.

In subsequent iterations of this study, we intend to incorporate the evaluation of diverse datasets, acknowledging the resource-intensive nature of our current dataset. This approach may offer enhanced insights into the influence of labeling quality on model performance and assess whether certain models exhibit improved performance with larger datasets, despite the presence of marginally lower-quality annotations. The findings underscore the necessity to refine our research scope. Considering the complexities inherent in language and the potential for bias originating from annotators, authors, and even topic-dependent factors, concentrating model development on bias detection for specific use cases may lead to more actionable and precise conclusions.

7. References

1. Entman, R. M. (2007). *Media framing biases and political power: Towards a taxonomy of frames*. In J. T. Cacioppo, L. G. Tassinary, & G. G. Berntson (Eds.), *Handbook of Neuroscience for the Behavioral Sciences* (pp. 691-707). Elsevier.
2. Spinde, T., Plank, M., Krieger, J.-D., Ruas, T., Gipp, B., & Aizawa, A. (2021). Neural Media Bias Detection Using Distant Supervision With BABE - Bias Annotations By Experts. *Findings of the Association for Computational Linguistics: EMNLP 2021*. Retrieved from https://aclanthology.org/2021.findings-emnlp.101/
3. Entman, R. M. (2007). Framing Bias: Media in the Distribution of Power. *Journal of Communication, 57*(1), 163-173.
4. Gentzkow, M., & Shapiro, J. M. (2011). Ideological segregation online and offline. *The Quarterly Journal of Economics, 126*(4), 1799-1839.
5. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825-2830.
6. Pew Research Center. (2022, September 28). *Americans see bias in media coverage, leaders*. Retrieved from https://www.pewresearch.org/fact-tank/2022/09/28/americans-see-bias-in-media-coverage-leaders/
7. World Health Organization (WHO). (2021). *Managing the COVID-19 infodemic: Promoting healthy behaviours and mitigating the harm from misinformation and disinformation*. Retrieved from https://apps.who.int/iris/handle/10665/342887
8. Chung, H., Gummadi, K. P., & Chetty, M. (2020). A Survey on Bias and Fairness in Machine Learning. *ACM Computing Surveys (CSUR)*, 54(6), 1-35.
9. Allcott, H., & Gentzkow, M. (2017). Social Media and Fake News in the 2016 Election. *Journal of Economic Perspectives, 31*(2), 211-236.
10. Hosmer, D. W., & Lemeshow, S. (2000). *Applied Logistic Regression*. New York: Wiley.
11. Mao, Y., Peng, H., Li, D., et al. (2020). ConvBERT: Improving BERT with Span-based Dynamic Convolutions. *arXiv preprint arXiv:2008.02496*. Retrieved from https://arxiv.org/pdf/2008.02496
12. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*. Retrieved from https://arxiv.org/pdf/1910.01108
13. Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *arXiv preprint arXiv:2003.10555*. Retrieved from https://openreview.net/pdf?id=r1xMH1BtvB
14. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L.,& Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692.* Retrieved from https://arxiv.org/abs/1907.11692