Distributed Fake News Detection

Danial Beg *Adithya Bhattiprolu*

1 Introduction

The proliferation of fake news across social media, digital platforms, and even legacy media has escalated into a pressing global concern. Ranging from harmless pranks to far-reaching consequences causing both physical and emotional harm, these deliberate misinformation campaigns have spurred the need for vigilant and up-to-date fake-news detection models. Given the extensive diversification of information on social media platforms, it has become imperative to continuously retrain and update fake-news detection models to ensure they remain aligned with the latest trends. Deploying a distributed training approach and utilizing a cluster allows for the efficient retraining and redeployment of the model without significant delays, as opposed to retraining the model on single system. By embracing this agile approach, we aim to ensure that our fake-news detection system remains highly effective in identifying and combating misinformation.

The surging demand for computational resources, attributed to the exponential growth of data used to train increasingly intricate models, has prompted the development of a distributed fake news detection model. This innovative model represents a distributed solution to the burgeoning issue of fake news and misinformation. By harnessing the power of a diverse array of nodes, the model is trained in a distributed manner, ensuring efficiency in training and liberating computational resources. This approach mirrors a real-world deployment scenario, where the model would operate across various servers in distinct data centers, catering to a global audience.

Our strategy entails training multiple pre-trained transformer models tailored to this task, drawing insights from an ensemble of datasets to craft a resilient fake news classifier, poised to rival existing systems. Leveraging MLflow, we intend to meticulously track and document all experiments and model configurations, thereby unraveling the impact of various hyperparameters on the model's performance. Furthermore, the data insights gleaned from MLflow will facilitate the implementation of a comprehensive error analysis framework, enabling the identification of recurring patterns or prevalent themes in misclassified instances. Lastly, MLflow allows for better model explainability as all runs for a model are logged with hyperparameters and loss for example being stored for future reference. Such a system will lead into better model explainability in the future to help keep the system transparent and also allow for ease of use and access for various disjoint teams. This analysis promises valuable insights into the model's vulnerabilities, guiding future enhancements in the detection system.

Following an exhaustive literature review, no existing work was found that addressed the development of a fake news detection system trained in a distributed fashion. Hence, the novelty of our solution lies in the holistic approach, encompassing distributed training, fine-tuning, containerization, and the potential for future deployment of a robust and scalable fake news classification model.

2 Motivation

The pervasive nature of fabricated or "fake" news has escalated with the rise of social media and the transition from print to digital media. According to a Pew Research report, approximately 64% of Americans acknowledge that fake news often distorts their understanding of current events and critical issues [2]. This staggering statistic underscores the crucial impact of fake news on American society. In recent times, the U.S. government has voiced apprehension regarding the role of social media platforms, including TikTok, in spreading misinformation orchestrated by foreign entities to deceive the American public. These concerns have underscored the urgent necessity for automated news classification systems to distinguish fabricated information, thereby preventing malicious actors from sowing panic or influencing public opinion based on falsehoods. President Joseph Biden has made combating misinformation a priority of his administration, demonstrating a commitment to leveraging machine learning to automate and streamline the detection process [13].

In parallel, our endeavor to develop a distributed classification process aims to address the escalating data volumes, where optimizing computational resources has become imperative. Our focus is on exploring methodologies that minimize computational requirements while ensuring the deployment of a robust and efficient model. Additionally, our integration of MLFlow into the workflow will enable comprehensive experiment tracking and model logging, enhancing the transparency of our approach. By promoting algorithmic explainability and engendering public trust, this transparent approach will allow collaborators to work more effectively and facilitate the exploration of previous model versions when necessary.

3 Related Work

Fake news, a pervasive issue in today's information landscape, has been extensively studied and classified into various categories. Notably, Zhou et al. [21] provide a comprehensive analysis of the term "fake news," distinguishing between different types such as "deceptive news," "clickbait," and "satirical news." Given the significant role of social media

platforms in the dissemination of fake news, several studies ([18] and [1]) have focused on detecting and classifying misinformation on prominent platforms like Facebook and Twitter. Kaur et al. [12] utilize machine learning techniques, including Naïve Bayes, Neural Network, and Support Vector Machine (SVM), to detect fake news. They emphasize the critical role of data normalization in the pre-processing stage, reporting a high accuracy rate of 96.08% for Naïve Bayes, and 99.90% for the advanced methods, neural networks, and SVM.

In a different approach, Riedel et al. [14] propose a stance detection system that labels articles as "agree," "disagree," "discuss," or "unrelated" based on the alignment between the article's headline and its main text. They leverage linguistic properties such as term frequency (TF) and term frequency-inverse document frequency (TF-IDF) as feature sets, achieving an impressive overall accuracy of 88.46%. Furthermore, Zhang et al. [20] employ explicit and latent textual features to classify news articles, while Kaliyar et al. [11] utilize basic Deep Convolutional Neural Networks (CNNs) to extract contextual information features for identifying fake news articles. Wani et al. [19] focus on identifying COVID-19 related fake news, demonstrating that pretraining transformer-based language models on subject-specific corpora and fine-tuning the model for the specific task yields the best accuracy. In a bid to improve upon existing models, Jwa et al. [9] propose "BAKE," an automatic fake news detection model that addresses the data imbalance issue associated with BERT. Their model outperforms all other existing models, demonstrating promising potential in the field of fake news detection. In our work we plan to take inspiration from the "BAKE" model by training a Roberta based language model for the fake-news detection task in a distributed fashion.

4 Overview and Approach

Our distributed fake news detection system comprises three distinct components:

1. Database
2. Model Training and Evaluation
3. Hyperparameter Tuning and Retraining

4.1 Database

In this study, we harnessed the power of three diverse datasets to train and evaluate the effectiveness of our models. These datasets include the Information Security and Object Technology (ISOT) [3], WELFake [17], and Getting Real About Fake News datasets [10], each dedicated to news data. Table 1 provides information on the dataset distribution.

The provided data encompasses textual information alongside corresponding labels indicating whether the news is classified as fake or real, which we will employ for analysis. We use the "Getting Real about Fake news" dataset as a test dataset and evaluate the performance of models that have been trained on either of the two remaining datasets as well as the the zero shot models. Presently, the datasets are sufficiently compact to be accommodated entirely in memory, and as such, they are stored locally. Nevertheless, the system is meticulously structured to facilitate seamless adaptability, allowing for the effortless retrieval of data from HDFS files or cloud storage with minimal modifications. It's important to note that opting for data retrieval from these alternative sources may result in an increase in training time due to the additional time required for data acquisition.

Dataset	True Instances	Fake Instances
Getting Real About Fake News	1,245	761
WELFake	35,028	37,106
ISOT	21,418	23,482
Politifact + Gossipcop	17,445	5,757

Table 1: Distribution of True and Fake News Instances in Datasets

4.2 Model Training

For our experiments, we employed four popular language models (bert-based model, roberta based fake news classfier, distilbert based model and a different roberta classifier) retrieved from the Hugging Face Model Hub [4, 5, 7, 8] for the task of fake news classification. Among these models, three are based on the RoBERTa architecture, while one is a DistilBERT-based classifier, all pre-trained specifically for fake news classification. We conducted an evaluation of the performance of all zero-shot models in our analysis.

4.3 Evaluation

In this module, the models undergo distributed training by leveraging multiple GPUs through an all-reduce approach, utilizing a dataset sourced from our database. Post-training, a diverse set of models undergo assessment using the designated test dataset. Throughout the evaluation phase, meticulous records of instances with incorrect predictions are meticulously kept and subsequently transmitted to the statistical analysis component for in-depth examination. To gauge the system's performance, we employ validation accuracy and F-1 score as key metrics

4.4 EDA

Once the model has completed training, we gather performance data through MLFlow logging. Analyzing the model's behavior on both the test and training datasets allows us to discern its tendencies. Armed with this insight, we undertake model retraining to enhance overall performance. Our scrutiny extends to factors such as learning rate, batch size, text simplicity, and text sentiment, offering a comprehensive evaluation of the model's efficacy. This iterative process not only provides a clearer understanding of the model's performance but also illuminates potential areas for future refinement.

4.5 Implementation

Our objective is to investigate the impact of the number of GPUs on both training time and overall accuracy. Furthermore, we aim to assess how text pre-processing influences model performance and to scrutinize the effects of hyperparameters such as learning rate. To facilitate distributed training, we executed the code on UCI's High-Performance Computing (HPC) cluster, utilizing a varying number of GPUs to simulate distributed nodes for dataset training. Our experimental setup ranged from a singular GPU as the smallest configuration to four GPUs, representing the maximum available resources for this endeavor.

5 Results

The results section is structured into three distinct components. Initially, we delve into the examination of model performance under distributed training systems. Subsequently, our attention shifts to the second component, where we concentrate on hyperparameter tuning to enhance overall results. Finally, in the third segment dedicated to Exploratory Data Analysis (EDA), we conduct a thorough analysis of the model's tendencies.

5.1 Effect of Number of GPUs

The impact of the number of GPUs on model performance is presented in Table 2. As evident from the table, the training time decreases as expected with an increase in the number of GPUs used for evaluation. Similar trends were observed when training the model with the WELFake and ISOT datasets. Notably, for smaller datasets, there was minimal alteration in the observed training time. Furthermore, when the model was trained on a subset of the original data, it was observed that the time required to train the model reached a saturation point with an increasing training data size, resulting in a graph reminiscent of an exponential decay. We believe that increasing the number of GPUs beyond this point does not yield significant changes in training time, with accuracy and F1 score remaining relatively consistent. To prevent job cancellation on the cluster all future results are obtained by training the model using 2 GPUs in a distributed fashion.

Table 2: Politifact and Gossipcop Results

GPUs	Time	Val. Acc.
1	3h 20m 47s	0.850
2	2h 56m 28s	0.848
3	2h 44m 40s	0.846
4	2h 35m 4s	0.840

5.2 HyperParameter Effect

To determine suitable hyperparameters that lead to improved performance we systematically varied the number of epochs, batch size, and learning rate to gauge their respective impacts on model performance.

1. **Batch Size -** We explored variations in batch size (ranging from 2 to 64) during the training of transformer models on the WELFake dataset. As anticipated, there was a decrease in the time taken for training; however, no noticeable change in model performance was observed.

2. **Learning Rate -** We varied the learning rate (5 e-8 to 5e-5 in magnitudes of 10) used to train the transformer models on the WELFake dataset and observed a negative correlation, i.e., as the learning rate increased the model's F1 score began to steadily drop. So for all future runs the learning rate was fixed at 5e-8.

3. **Epochs -** Finally, we varied the number of epochs (ranging from 0 to 5) during the training of transformer models on the WELFake dataset. We observed a performance increase with an upsurge in the number of epochs. However, not all models behaved the same. For instance, the performance of the Fake-news BERT model plateaued, while the DistilBERT model exhibited a decline. Our suspicion is that, in the case of DistilBERT, the model may have overfit the WELFake dataset (as indicated by a training accuracy of 99.87%), resulting in compromised generalization capability. Figure 1 visually depicts the performance trend of the fake-news detector model as the number of epochs increases.

   An intriguing insight emerged during our analysis: as the number of epochs increased, test accuracy exhibited a decline, while the F1 score demonstrated an opposite trend, presenting conflicting results. Subsequent examination of the confusion matrix shed light on this phenomenon. Figure 2 showcases the confusion matrix of the Fake-news BERT model before and after fine-tuning. Notably, an initial class imbalance in the test dataset led the model to predominantly predict everything as fake news, consequently achieving a higher accuracy. However, this

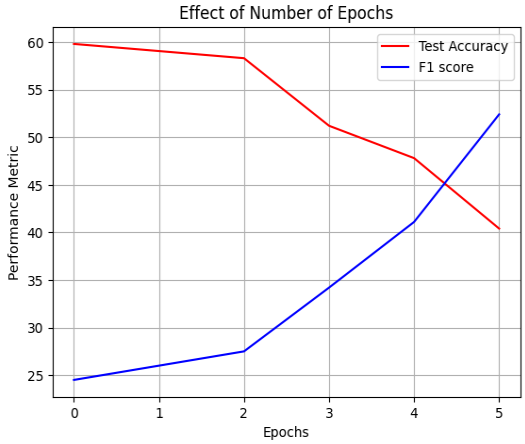

Figure 1: Variation in test accuracy and F1 score as the number of epochs vary for Fake-News-Bert

majority prediction affected precision and recall, resulting in a lower F1 score. As the model refined itself, sample distribution across the matrix became more even, contributing to an improved F1 score at the expense of validation accuracy.

Henceforth, we opt to utilize F1 score as the primary metric for evaluating model performance to mitigate the potential misinterpretation induced by class imbalance in validation accuracy.

5.3 EDA Effect

To gain deeper insights into the tendencies of transformer models, we undertook a comprehensive analysis of their behaviors across diverse datasets. We primarily address concerns related to class imbalance, token length, and sentiment in our evaluation.

5.3.1 Understanding Performance

Table 3 presents the outcomes derived from training four transformer models in a distributed manner on the WELFake dataset, followed by an evaluation of their performance on the "Getting Real With Fake News" dataset.

uilding upon the insights from the preceding subsection, where F1 score was established as the primary metric, we discern that the optimally performing model is the fine-tuned BERT model. Remarkably, fine-tuning enhances the performance of three models, underscoring its efficacy. However, for the RoBERTa-based model, the performance declines post-finetuning, suggesting that the original RoBERTa-based classifier exhibited superior generalization capabilities. Subsequently, for the subsequent exploratory analysis, we leverage the best-performing model, namely the fine-tuned BERT model.

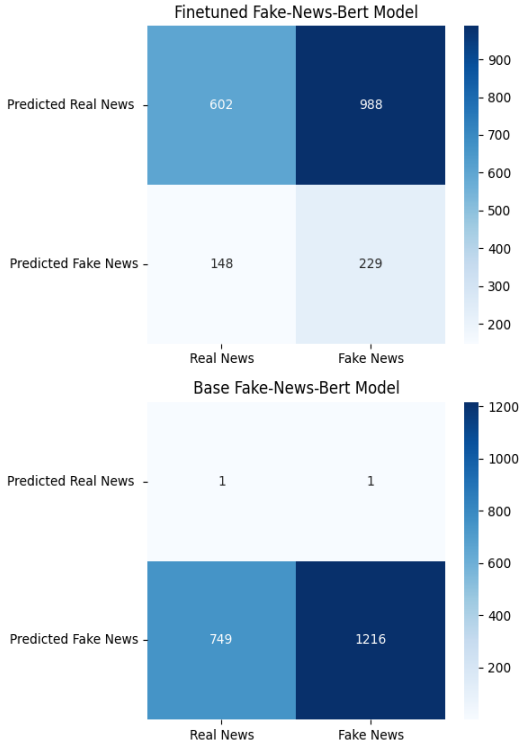

Figure 2: Confusion matrix of finetuned and default Bert based transformer model

5.3.2 Effect Of Class Imbalance

Upon scrutinizing the confusion matrix during model evaluation, a noteworthy tendency surfaced— the model exhibited a propensity to misclassify news as fake even when it was not (resulting in false positives). Given the dataset's inherent class imbalance, with 2000 more fake news samples than real news samples, our experiment sought to explore the impact of this imbalance on model performance.

To address this, we randomly selected 4000 fake news articles and 4000 real news articles from the WELFake dataset, creating a balanced dataset for training. The performance of the model trained on this balanced dataset was then evaluated on the "Getting Real with Fake News" dataset. Table 4 illustrates that by rectifying the class imbalance, we achieved a model with enhanced performance. Notably, a similar performance boost was observed when the dataset's size was further increased from 4000 to 6000 samples for each class.

5.3.3 Effect Of Summarization

Our investigation sought to address another research question: whether the length of a text article introduces bias in the model's behavior, specifically whether the model exhibits a higher inclination to classify longer articles as fake. To explore this potential bias and assess the model's robustness,

Model	Approach	Accuracy (%)	Precision (%)	Recall (%)	F-1 Score (%)
Fake News (Roberta)	Original	**61.9**	0.5	0.001	0.003
	Finetune	42.2	37.9	80.3	51.5
bert	Original	59.8	43.1	17.1	24.5
	Finetune	40.4	37.7	**86.1**	**52.4**
distilbert	Original	59.2	30.9	0.05	0.09
	Finetune	59.2	33.1	0.06	0.11
roberta	Original	60.2	46.9	32.4	38.3
	Finetune	61.9	**50.3**	12.7	20.2

Table 3: Performance metrics for different models and approaches.

Model	F1 score
Bert Based Original	24.5%
Bert Based (FT)	52.40%
Bert Based (Balanced data)	**54.9%**

Table 4: Effect of Class imbalance

we devised an experiment to examine the effect of reducing input size and paraphrasing on the model's performance.

To achieve this, we employed a well-regarded Hugging Face transformer [6] for text summarization. Utilizing this model, we paraphrased the 8000 randomly selected samples from the previous experiment, creating a simplified and balanced dataset that was then fed to the BERT-based model.

Notably, this experiment required a substantially larger number of epochs for model training. We systematically increased the number of epochs from 1 to 25 in increments of 5 to observe the model's performance. Figure 3 illustrates the F1 score as a function of the number of epochs. Initially, the test F1 score experiences an upward trend, reaching 46.4% after 10 epochs, making it the third-best performing model compared to the results obtained in Table 3. However, beyond 10 epochs, the test F1 score starts to decline, despite ongoing improvements in training performance.

5.3.4 Effect Of Sentiment

In our final exploration, we sought to ascertain whether the sentiment of a news article had any discernible impact on model performance. For this investigation, we utilized the 4000 randomly sampled balanced dataset generated in earlier experiments along with the bert based model. It's worth noting that despite achieving class balance in terms of fake and real news, the sentiment distribution within this dataset remained heavily skewed. Approximately 73.72% of the samples exhibited positive sentiment, with the remaining samples expressing negative sentiment. Our hypothesis was that this sentiment difference causes a bias in the models behavior.

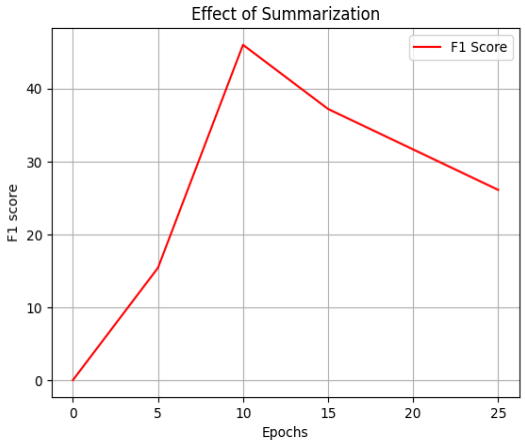

Figure 3: Effect of Epochs using the Summarized dataset.

However, upon evaluating the model's performance on this sentiment-skewed dataset, we found no significant correlation between sentiment and model performance. Both the original and fine-tuned models demonstrated similar classification accuracies on samples with positive and negative sentiment, suggesting that sentiment had limited influence on the model's classification outcomes.

6 Discussion

Based on the results we obtained in the previous section. we opted to split the discussion section into two distinct parts: one focused on the challenges we encountered, and the other dedicated to drawing conclusions and outlining directions for future work.

6.1 Issues

In our initial observations, we noted an abrupt surge in training time corresponding to an increase in the number of GPUs utilized. We hypothesize that this phenomenon may be attributed to our utilization of a free High-Performance Com-

puting (HPC) account lacking priority execution privileges within the cluster. This might have led to the cancellation of our training jobs when a larger number of GPUs were requested, resulting in an unexpected extension of training time. This hypothesis can be readily validated by rerunning the models with datasets configured for priority execution.

We initially planned on using the Flesch-Kincaid Reading Ease [15] for obtaining a simplicity score and then using simplification models to paraphrase the text, but then decided not to because of the ease with which it can be manipulated as described in [16].

We faced issues in sentiment analysis, wherein we expected the bias in dataset to have an effect on the model performance. However, after exhaustive testing by varying models, datasets and hyperparameters we were unable to obtain a conclusive relation between the sentiment of the dataset and its affect on the model. Given the time constraints in this aspect, we deferred the implementation of topic modeling, leaving it as a prospect for future work.

6.2 Conclusion and Future Work

In this study, we conducted an extensive evaluation of four language models trained in a distributed fashion for the fake news classification task, leveraging five distinct datasets. Subsequently, we devised a comprehensive pipeline to scrutinize model tendencies under diverse scenarios, aiming to inform training strategies and enhance overall performance. Furthermore, this Exploratory Data Analysis (EDA) pipeline can be expanded to encompass additional textual properties such as topic modeling (e.g., LDA) and named entity recognition (NER) to gain deeper insights into model tendencies.

Despite achieving performance improvements, the F1 score of the best-performing model remains relatively modest, hovering around 54%. To address this limitation, future research endeavors should explore the capabilities of larger models. From a systems perspective, techniques like model parallelism could be employed to train substantially larger models for the same task, thereby investigating potential performance enhancements.

References

[1] Hunt Allcott and Matthew Gentzkow. Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–236, 2017.

[2] Pew Research Center. Many americans believe fake news is sowing confusion, 2016.

[3] Martins Samuel Dogo, Deepak P, and Anna Jurek-Loughrey. Exploring thematic coherence in fake news, 2020.

[4] HuggingFace. distilbert. https://huggingface.co/hamzab/roberta-fake-news-classification.

[5] HuggingFace. Fake-news-bert-detect. https://huggingface.co/jy46604790/Fake-News-Bert-Detect.

[6] HuggingFace. falconsai/textsummary. https://huggingface.co/Falconsai/text_summarization.

[7] HuggingFace. roberta-fake-news. https://huggingface.co/ghanashyamvtatti/roberta-fake-news.

[8] HuggingFace. roberta-fake-news-classification. https://huggingface.co/hamzab/roberta-fake-news-classification.

[9] Heejung Jwa, Dongsuk Oh, Kinam Park, Jang Mook Kang, and Heuiseok Lim. exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences*, 9(19):4062, 2019.

[10] Kaggle. Getting real about fake news. https://www.kaggle.com/datasets/ruchi798/source-based-news-classification. Accessed on YYYY-MM-DD.

[11] Rohit Kumar Kaliyar, Anurag Goswami, Pratik Narang, and Soumendu Sinha. Fndnet – a deep convolutional neural network for fake news detection. *Cognitive Systems Research*, 61:32–44, 2020.

[12] Prabhjot Kaur, Rajdavinder Singh Boparai, and Dilbag Singh. Hybrid text classification method for fake news detection. *International Journal of Engineering and Advanced Technology (IJEAT)*, 8(5):2388–2392, 2019.

[13] Politico. Biden's social media misinfo fight. https://www.politico.com/news/2023/09/19/bidens-social-media-misinfo-fight-00116721, 2023.

[14] Benjamin Riedel, Isabelle Augenstein, Georgios P. Spithourakis, and Sebastian Riedel. A simple but tough-to-beat baseline for the fake news challenge stance detection task, 2018.

[15] Marina Solnyshkina, Radif Zamaletdinov, L.A. Gorodetskaya, and A.I. Gabitov. Evaluating text complexity and flesch-kincaid grade level. *Journal of Social Studies Education Research*, 8:238–248, 11 2017.

[16] Teerapaun Tanprasert and David Kauchak. Flesch-kincaid is not a text simplification evaluation metric. In *Proceedings of the 1st Workshop on Natural Language*

Generation, Evaluation, and Metrics (GEM 2021), pages 1–14, Online, August 2021. Association for Computational Linguistics.

[17] Pawan Kumar Verma, Prateek Agrawal, Ivone Amorim, and Radu Prodan. Welfake: Word embedding over linguistic features for fake news detection. *IEEE Transactions on Computational Social Systems*, 8(4):881–893, 2021.

[18] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018.

[19] Apurva Wani, Isha Joshi, Snehal Khandve, Vedangi Wagh, and Raviraj Joshi. Evaluating deep learning approaches for covid19 fake news detection. In *Combating Online Hostile Posts in Regional Languages during Emergency Situation*, pages 153–163. Springer International Publishing, 2021.

[20] Jiawei Zhang, Bowen Dong, and Philip S. Yu. Fakedetector: Effective fake news detection with deep diffusive neural network, 2019.

[21] Xinyi Zhou and Reza Zafarani. A survey of fake news. *ACM Computing Surveys*, 53(5):1–40, sep 2020.