Classifying Fake News in Filipino Language using Improved DistilBERT Model

Jan Mar Ruel Espiritu
College of Engineering Architecture and Computing
Notre Dame of Marbel University
Koronadal City, Philippines
cassylum23@gmail.com

Ralph John E. Policarpio
College of Engineering Architecture and Computing
Notre Dame of Marbel University
Koronadal City, Philippines
ralphpolicarpio513@gmail.com

Aliah Chavy B. Sabado
College of Engineering Architecture and Computing
Notre Dame of Marbel University
Koronadal City, Philippines
aliahchavysabado@gmail.com

Vince Marc B. Sabado
College of Engineering, Architecture, and Computing
Notre Dame of Marbel University
Koronadal City, Philippines
vmbsabado@ndmu.edu.ph

Hajah T. Sueno
College of Engineering, Architecture, and Computing
Notre Dame of Marbel University
Koronadal City, Philippines
htsueno@ndmu.edu.ph

Abstract— **Detecting misinformation in the Filipino language poses unique challenges, especially when working with limited data resources. This study explored the use of transformer models, specifically pre-trained models like BERT, to construct a fake news classification model for Filipino. However, BERT's large size and computation hinder its deployment in resource-constrained environments. To address this, the study focuses on developing a lightweight fake news classifier using the DistilBERT architecture, fine-tuned for Filipinos. The model's performance was evaluated against BERT based on accuracy, precision, recall, F1 score, inference time, memory requirement, and parameter size. Results indicate that while BERT exhibits slightly better classification performance, DistilBERT offers significant advantages in efficiency with faster inference time and smaller resource requirements.**

Keywords— Fake news, BERT, Transformer, DistilBERT, Fine-tune, Classification model, Data augmentation

I. INTRODUCTION

Fake news, characterized by false information masquerading as legitimate news, undermines the credibility of traditional media by lacking essential editorial standards [1]. The rapid spread of misinformation through online channels and social media platforms has the potential to sway public sentiment and undermine trust in democratic institutions. To combat this growing challenge, artificial intelligence (AI) has emerged as a promising strategy, leveraging machine learning techniques to automatically detect patterns and indicators of fake news for more effective mitigation strategies.

In their research, [2] utilized transfer learning to identify misinformation in the Filipino language. However, the reliance on large models like BERT presented obstacles to widespread deployment due to their considerable size. This study aimed to address this issue by developing a more efficient model, utilizing a smaller pre-trained model specifically optimized for identifying misleading information. By incorporating DistilBERT, a streamlined variant of BERT introduced by [3], this approach seeks to balance robust fake news detection and practical considerations for deployment on everyday devices. This advancement held promise in providing accessible and effective solutions to combat the proliferation of fake news.

Detecting misinformation in the Filipino language poses challenges due to limited data resources. To address this, constructing a model relies on transformer models like BERT, pre-trained on multiple languages, and then fine-tuned for classifying fake news in Filipino. However, the significant hurdle lies in BERT's large size and speed, with configurations containing up to 340 million parameters, making deployment difficult in environments with limited resources.

This research aimed to develop an efficient and lightweight model for classifying fake news in Filipinos using an enhanced version of DistilBERT. The study sought to achieve this by fine-tuning the DistilBERT architecture for fake news classification in Filipino, applying data augmentation techniques to enhance performance. Furthermore, the research aimed to evaluate the efficacy of the improved DistilBERT model compared to both BERT and DistilBERT models, considering essential criteria such as accuracy, precision, recall, and F1-score. The classification performance was also compared based on parameters like parameter size, memory requirement, and inference time. Statistical analysis using the ANOVA test and Tukey's Honestly Significant Difference (HSD) test was conducted to assess the significance of accuracy among the three models, ultimately aiming to integrate the improved DistilBERT model into a web application.

II. REVIEW OF RELATED LITERATURE

A. Deep Learning

Artificial neural networks, a subset of machine learning called deep learning, extract insights by training on large datasets. This enables them to excel in language translation, computer vision, and natural language processing tasks, with designs tailored to specific challenges in different fields [4]. Convolutional Neural Networks (CNN) are commonly used for image recognition and computer vision tasks, whereas Recurrent Neural Networks (RNN) are useful for forecasting and handling time series problems [5].

B. BERT

Recognized as Bidirectional Encoder Representations from Transformers, BERT stands out among the latest models for language representations, designed for thorough, bidirectional training of unannotated text, considering context on both sides at every layer. This unique configuration empowers BERT's pre-training, showcasing excellence in diverse tasks like language inference and question answering, often requiring minimal adjustments to task-specific structures, typically involving just a single output layer [6]. Despite BERT's benefits in distinguishing between sentences and its extensive pre-training, its classification as a large-scale model with 340 million parameters poses computational challenges for practical use in downstream tasks [7].

C. DistilBERT

DistilBERT, a lighter and faster iteration of BERT, demonstrates impressive versatility and performance in various language tasks. This is achieved through knowledge distillation during pre-training, resulting in a significant reduction in size. Despite its smaller size, DistilBERT maintains 97% of BERT's language understanding capabilities while offering a notable speed improvement, making it well-suited for edge applications where smaller and faster models are preferred [3].

According to [8], fine-tuning DistilBERT for text classification involves using the sigmoid activation function to generate probability distributions across potential labels for input text. During this process, DistilBERT is trained to predict accurate labels for each provision in the training dataset, minimizing the difference between predicted and actual labels using binary cross-entropy as the optimization criterion.DistilBERT.

D. Transformer

The Transformer, a neural network architecture centered on the self-attention mechanism, has revolutionized various applications in natural language processing (NLP), showcasing its effectiveness with prominent models like BERT and GPT-3 [9]. Unlike traditional neural networks like LSTM and RNNs, transformers excel in handling complex relationships within input sequences, sparking significant interest among AI researchers and becoming popular in domains like computer vision and NLP [10].

E. Fine-tuning

Fine-tuning in deep learning involves adjusting pre-trained models with new data to refine performance. It is akin to updating existing knowledge based on fresh information, leveraging knowledge from a large dataset to adapt to a specific task or domain. This method, particularly useful in transfer learning scenarios, allows for faster convergence and requires less labeled data than training from scratch, as the pre-trained model is a starting point for learning a new task [11]. Training a model for a fixed number of epochs, like three, helps mitigate overfitting by striking a balance between learning from training data and avoiding memorization, capturing underlying patterns and features more likely to generalize to unseen data [12]. Choosing a batch size, such as 8, impacts model performance by balancing noise regularization and training efficiency, with smaller batch sizes introducing more noise for regularization and larger batch sizes providing more data for effective learning, ultimately enhancing accuracy [13].

F. Data Augmentation

Data augmentation in natural language processing (NLP) involves generating synthetic training data to address data scarcity, especially in deep learning models, aiming to enhance performance by increasing data diversity and quantity. Particularly in sentiment analysis, augmenting training data with diverse examples enriches the dataset, enabling models to capture language complexities and nuances better, leading to improved sentiment analysis across languages and datasets [14]. Random deletion, a technique within the Easy Data Augmentation (EDA) framework, improves text classification by randomly removing words to augment the dataset, introducing noise as a regularization method to prevent overfitting and enhance model robustness, potentially capturing more salient text features [15].

G. Memory Requirement

Determining the memory requirement for a large language model (LLM) hinges on the size of the model's parameters. Memory needs during training are typically 3 to 4 times greater than during inference due to factors like backpropagation and Transformer architectures' complexity. A conservative estimate suggests multiplying each parameter's size by 4 bytes to calculate the memory requirement. This provides an approximate measure for storing the model's parameters during inference, as exemplified by Qwen-7B requiring an estimated 28 GB for inference [16].

H. Fake News

False information, resembling media presentations but lacking authentic editorial standards, pertains to deceptive content that does not adhere to credible news outlets' rigorous fact-checking processes [1]. According to [17], identifying misinformation on social platforms is challenging due to its varied manifestations, complex storytelling, lack of proper references, abundant negative emotional language, and longer length, requiring diverse solutions to address its dissemination effectively.

III. CONCEPT OF THE STUDY

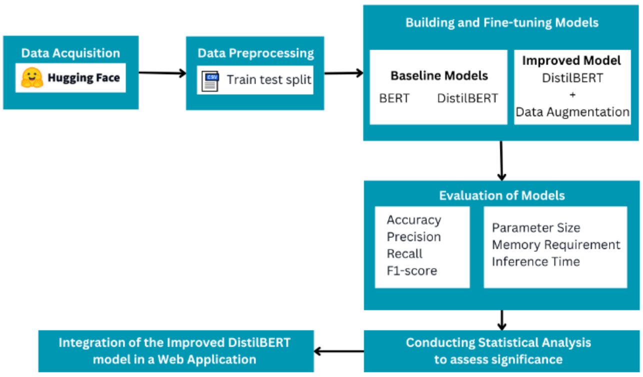


Fig. 1. Conceptual Framework of the Study

A. Data Acquisition

For this study, the researchers utilize the Fake News Filipino Dataset from the HuggingFace website "https://huggingface.co/datasets/fake_news_filipino". This dataset comprises Filipino news articles, each labeled as fake or real, with labels 0 and 1, respectively. The dataset was evenly balanced, with 50% of instances labeled as fake news

(0) and the remaining 50% as real news (1), totaling 3,210 rows, each representing an instance.

B. Data Preprocessing

The dataset was divided into a train-test split, with 70% allocated for training the fine-tuning of DistilBERT and 30% reserved for evaluating the model's accuracy. This method ensures adequate training while allowing separate evaluation of the model's performance on unseen data, providing a robust assessment of its effectiveness in classifying Filipino fake news articles.

C. Building and Fine-tuning Models

During this stage, the researchers fine-tuned two pre-trained models, BERT and DistilBERT, for the Fake News Classification task, initially training both models for 3 epochs with a batch size of 8. The improved DistilBERT model was designed by integrating advanced fine-tuning methods and data augmentation techniques, such as random deletion, to enhance its ability to generalize from limited data. This approach involved modifications to the original DistilBERT architecture, optimizing its performance for the fake news classification task. The improved DistilBERT model, with its enhanced design and augmentation strategy, served as the baseline for evaluating performance and provided valuable insights into the effectiveness of these enhancements in classification capabilities.

D. Evaluation of Models

After fine-tuning the Fake News Filipino Dataset, the evaluation aimed to gauge the model's ability to distinguish between fabricated and genuine news texts in Filipino. Five metrics, including accuracy, precision, recall, and F1-score, were utilized to assess classification performance comprehensively, providing detailed insights. Furthermore, the evaluation considered the model's parameter size, inference time, and memory requirements to understand its efficiency, scalability, and practical applicability for real-world deployment in detecting fake news in Filipino.

E. Conducting Statistical Analysis to Assess Significance

Statistical analysis for accuracy involved utilizing ANOVA followed by the Tukey HSD test to determine significant differences among multiple models. ANOVA assesses accuracy variations between models, while the Tukey HSD test identifies specific models with significant differences in accuracy. These tests offer insights into model performance, helping researchers make informed decisions for their study or application.

F. Integration of the Improved DistilBERT model in a Web Application

After fine-tuning and improving the DistilBERT model, the researchers downloaded it and imported it to their local computer. Subsequently, they deployed the model as an API, enabling the web application to connect to it and load text data to classify text as real or fake news. This deployment strategy allows for seamless model integration into various front-end interfaces, enabling users to leverage the model's capabilities for text classification easily.

IV. METHODOLOGY

A. Materials

1) Hardware

For this study, the utilized hardware consists of a laptop computer featuring an Intel Core i5-10300H @ 2.5GHz with RTX 2060, 24GB of DDR4 memory in 2933 MHz, and a 64-bit Windows 11 Home environment.

2) Software

a) Google Colab - significantly accelerated the fine-tuning of the model by utilizing powerful GPUs, enabling seamless execution of crucial Python scripts and offering cost-effective, collaborative features, thereby greatly improving the efficiency and effectiveness of the study.

b) PyTorch

The researchers employed the PyTorch library to train transformer models. They initially loaded the training dataset from a CSV file using pandas, followed by tokenizing the text data and conducting essential preprocessing steps. They then created a custom PyTorch dataset class to organize the tokenized data with corresponding labels and ultimately trained the model using PyTorch's Trainer class with specified training arguments.

B. Procedures

The study proceeds with the following steps:

1) Data Acquisition

The first step in developing the light transformer model task involved data acquisition, utilizing a dataset comprising 3,210 samples evenly split between fake and real news text, with labels categorized as 1 and 0, where 0 represents real news and 1 represents fake news. The dataset, sourced from "https://huggingface.co/datasets/fake_news_filipino," was used to fine-tune transformer models for fake news classification, aligning with a previous study by Cruz et al. (2019) titled "Localization of Fake News Detection via Multitask Transfer Learning," where it was also employed for detecting fake news in Filipino.

2) Data Pre-processing

This phase occurred before the training and testing of the data, with all texts comprising a combination of Tagalog and English. The dataset was divided into two groups, with a significant portion allocated for training, representing 70% of the total dataset size, while the remaining 30% was reserved for testing purposes.

3) Building the Model

In the initial stage of model construction, pre-trained models were sourced from specific websites and configured for a binary classification task with two labels, ensuring accurate tokenization of Tagalog language inputs. Subsequently, both the training and testing datasets undergo tokenization using the tokenizer object, with settings like 'truncation=True' and 'padding=True' facilitating uniform input length for efficient batch processing. To optimize the DistilBERT model's performance, data augmentation techniques such as random deletion are applied, creating varied sentence versions to enhance the model's generalization on unseen data. Following this, training arguments, including hyperparameters like epochs, batch size, and learning rate scheduler settings, are defined to fine-tune the model for classifying Tagalog news articles as real or fake, carefully considering computational resources and model convergence. Finally, the Trainer class encapsulates the model, training arguments, and dataset, with the 'train' method iteratively optimizing the model's weights to minimize loss and enhance performance on the classification task.

4) Evaluating the Model

To assess and validate the efficacy of a classification model, it was crucial to utilize classification techniques and performance metrics such as Accuracy, F1-score, Precision, Recall, inference time, parameter size, and memory requirements, offering a comprehensive assessment of the model's suitability for real-world deployment.

Accuracy in deep learning is a metric used to measure the proportion of correctly classified instances from the total instances in a dataset. It is calculated as the number of correct predictions divided by the total number of predictions made [18].

$$Accuracy = \frac{(TP+TN)}{((TP+FP)+(TN+FN))} \quad (1)$$

Precision is a classification metric that measures the proportion of true positive predictions (correctly predicted positive instances) out of all positive predictions (true positives and false positives). It is calculated as the number of true positives divided by the sum of true and false positives [18].

$$Precison = \frac{TP}{(TP+FP)} \quad (2)$$

Recall, synonymous with sensitivity or true positive rate, assesses the proportion of accurately predicted positive instances among all actual positives, calculated by dividing true positives by the sum of true positives and false negatives [18].

$$Recall = \frac{TP}{(TP+FN)} \quad (3)$$

The F1-score, a classification metric, harmoniously integrates precision and recall, serving as their balanced measure calculated by multiplying their product by 2 and dividing by their sum [18].

$$F1 - score = \frac{2*Precision*Recall}{(Precision+Recall)} \quad (4)$$

where TP = True Positive, TN = True Negative, FP = False Positive, and is FN = False Negative.

Parameter Size and Memory Requirement were evaluated by summing the number of elements in each parameter tensor to determine the model's capacity and resource requirements. The total memory requirement was calculated by multiplying the total number of parameters by the size of each parameter element. Inference time, crucial for assessing model efficiency, was computed by measuring the duration of model inference for each batch of evaluation data and averaging the inference time per batch. These assessments collectively informed the model's effectiveness and practicality for real-world applications, ensuring it met the necessary criteria for deployment.

5) Conducting Statistical Analysis to Assess Significance

The statistical analysis using the ANOVA test followed by the Tukey HSD test compared the accuracy of multiple groups or models in a study, determining whether statistically significant differences existed among them. The ANOVA test identifies overall differences in accuracy. In contrast, the Tukey HSD test offers a detailed examination by pinpointing specific groups that differ from each other, aiding researchers in selecting the most effective approach for their research objectives.

6) Integration of the Improved DistilBERT Model in a Web Application

After fine-tuning and enhancing the DistilBERT model, the researchers saved the model and tokenizer to a specified directory. They then developed an API using Flask to deploy the model. The Flask application initializes and loads the pre-trained DistilBERT model and tokenizer from the saved directory. This API allows the web application to send text data for classification by connecting to the Flask server. When the web application sends a request, the Flask API processes the text through the DistilBERT model, classifying it as real or fake news, and returns the results to the web application.

On the front end, users input text through the web interface, which sends this data to the Flask API. The API processes the text and returns the classification results, which are then displayed to the users. This integration approach using Flask ensures that the model is efficiently and seamlessly incorporated into the web application, providing users with a straightforward and effective means to classify text in real-time

V. RESULTS AND DISCUSSIONS

The models' performance was evaluated using standard classification metrics: Accuracy, Precision, Recall, and F1-score. The dataset consisted of 3,210 Filipino articles categorized into fake and non-fake news, each comprising 50% of the data. The dataset was split into training and testing sets using a 70-30 train-test split to train and evaluate the models. The DistilBERT and BERT models were each trained five times, and their evaluation results were recorded. Additionally, DistilBERT with data augmentation was trained five times and evaluated. The two models without data augmentation served as baselines for comparing the performance of DistilBERT with data augmentation.

TABLE I. RESULT OF THE AVERAGE PERFORMANCE EVALUATION OF THREE MODELS

Comparison	**BERT**	**DistiBERT**	**Improved DistilBERT**
Accuracy	0.926	0.916	0.947
Precision	0.933	0.906	0.954
Recall	0.916	0.928	0.938
F1 - Score	0.924	0.917	0.946

Table 1 displays the average performance metrics of three models: BERT, DistilBERT, and an enhanced version of DistilBERT. The improved DistilBERT consistently outperforms both BERT and the original DistilBERT across all metrics. Notably, the improved DistilBERT achieves the highest average accuracy, precision, recall, and F1 score, demonstrating its superior capability in accurately predicting and classifying data compared to the other models.

TABLE II. RESULT OF THE MODEL SIZE AND SPEED COMPARISON

Comparison	BERT	DistiBERT	Improved DistilBERT
Parameter Size	109,160,450 Parameters	66,631,682 Parameters	66,631,682 Parameters
Estimated Memory Requirement	416.41 MB	254.18 MB	254.18 MB
Inference Time	0.02402 seconds	0.00955 seconds	0.00862 seconds

Table 2 compares three models: BERT, DistilBERT, and an enhanced version of DistilBERT, focusing on parameter size, memory requirement, and inference time. BERT has the largest parameter size and memory requirement, while both versions of DistilBERT have smaller parameter sizes and memory requirements. Regarding inference time, BERT has the longest. At the same time, DistilBERT demonstrates improvement, with the enhanced DistilBERT achieving the shortest inference time, indicating its superior efficiency in processing compared to the other models.

TABLE III. ANOVA TEST RESULT

Anova: Single Factor

SUMMARY

Groups	*Count*	*Sum*	*Average*	*Variance*
BERT	5	4.63	0.926	1.54074E-32
DistilBERT	5	4.582	0.9164	9.8E-06
Improved DistilBERT	5	4.736	0.9472	3.2E-06

ANOVA

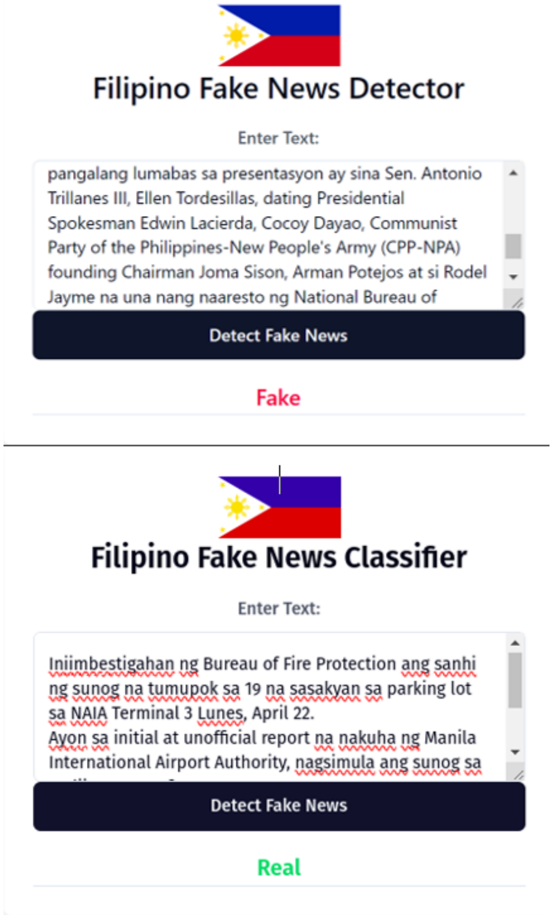
Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	0.002483733	2	0.001241867	286.5846154	7.43701E-11	3.885293835
Within Groups	5.2E-05	12	4.33333E-06			
Total	0.002535733	14				

Table 3 compares the performance of BERT, DistilBERT, and Improved DistilBERT models using ANOVA. It shows that Improved DistilBERT outperformed both BERT and DistilBERT with a significantly higher average score. The analysis indicates substantial differences in performance among the models, highlighting the superiority of Improved DistilBERT and providing insights for model selection in different contexts.

TABLE IV. TUKEY'S HONESTLY SIGNIFICANT DIFFERENCE (HSD) TEST RESULT

	Mean	Difference	HSD	Significant
BERT VS. DistilBERT	0.926 - 0.916	0.01	0.00351	Yes
Improved DistilBERT VS. BERT	0.947 - 0.926	0.021	0.00351	Yes
Improved DistilBERT VS. DistilBERT	0.947 - 0.916	0.031	0.00351	Yes

Table 4 displays the Tukey's Honestly Significant Difference (HSD) test results for accuracy scores, revealing significant differences among the performance of BERT, DistilBERT, and Improved DistilBERT models. While BERT shows slightly higher accuracy than DistilBERT by 0.01, both BERT and Improved DistilBERT notably surpass DistilBERT in accuracy, with mean differences of 0.021 and 0.031, respectively, indicating improved accuracy with architectural enhancements in DistilBERT..



Filipino Fake News Detector

Enter Text:

pangalang lumabas sa presentasyon ay sina Sen. Antonio Trillanes III, Ellen Tordesillas, dating Presidential Spokesman Edwin Lacierda, Cocoy Dayao, Communist Party of the Philippines-New People's Army (CPP-NPA) founding Chairman Joma Sison, Arman Potejos at si Rodel Jayme na una nang naaresto ng National Bureau of

Fake

Filipino Fake News Classifier

Enter Text:

Iniimbestigahan ng Bureau of Fire Protection ang sanhi ng sunog na tumupok sa 19 na sasakyan sa parking lot sa NAIA Terminal 3 Lunes, April 22.
Ayon sa initial at unofficial report na nakuha ng Manila International Airport Authority, nagsimula ang sunog sa

Real

Fig. 2. Filipino Fake News Classifier Web App

Figure 2 presents the web application implementing the improved DistilBERT model. This application features an input box where users can submit text. Upon clicking the “Detect Fake News” button, the model provides a prediction regarding the authenticity of the input text, labeling it as either fake or real news.

VI. CONCLUSION AND RECOMMENDATION

Based on the study findings, several conclusions were drawn regarding the performance and suitability of BERT and DistilBERT for fake news classification tasks. DistilBERT demonstrated competitive performance comparable to BERT in certain scenarios, highlighting its potential as a viable alternative. Moreover, the implementation of data augmentation notably enhances DistilBERT's performance, surpassing both BERT and the baseline DistilBERT model in accuracy and other metrics.

DistilBERT's smaller parameter size and lower memory requirement make it a more efficient choice, particularly in resource-constrained environments. The reduction in inference time further highlighted its improved efficiency, positioning it as an attractive option for applications demanding fast and accurate processing. DistilBERT's efficiency and effectiveness underscore its promise for various natural language processing tasks, especially where computational resources are limited.

Considering these findings, future researchers were recommended to explore alternative fine-tuning strategies to optimize DistilBERT's performance further. Additionally, larger and more diverse datasets could enhance the model's generalizability and effectiveness in detecting deceptive content. Investigating newer transformer architectures for binary text classification and exploring different approaches to measure the model's speed are also suggested avenues for future research to advance the field of fake news detection and classification.

REFERENCES

[1] Lazer, D. M. J., Baum, M. A., Benkler, Y., Berinsky, A. J., Greenhill, K. M., Menczer, F., Metzger, M. J., Nyhan, B., Pennycook, G., Rothschild, D., Schudson, M., Sloman, S. A., Sunstein, C. R., Thorson, E. A., Watts, D. J., & Zittrain, J. L. (2018). The science of fake news. Science, 359(6380), 1094–1096. https://doi.org/10.1126/science.aao2998

[2] Cruz, J. C. B., Tan, J. A., & Cheng, C. (2019). Localization of fake news detection via multitask transfer learning. ArXiv Preprint ArXiv:1910.09295.

[3] 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.

[4] Sarker, I. H. (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions. SN Computer Science, 2(6), 420. https://doi.org/10.1007/s42979-021-00815-1

[5] Shrestha, A., & Mahmood, A. (2019). Review of Deep Learning Algorithms and Architectures. IEEE Access, 7, 53040–53065. https://doi.org/10.1109/ACCESS.2019.2912200

[6] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. ArXiv Preprint ArXiv:1810.04805.

[7] Arase, Y., & Tsujii, J. (2019a). Transfer fine-tuning: A BERT case study. ArXiv Preprint ArXiv:1909.00931.

[8] Tuggener, D., von Däniken, P., Peetz, T., & Cieliebak, M. (2020). LEDGAR: A Large-Scale Multi-label Corpus for Text Classification of Legal Provisions in Contracts. In N. Calzolari, F. Béchet, P. Blache, K. Choukri, C. Cieri, T. Declerck, S. Goggi, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, & S. Piperidis (Eds.), Proceedings of the Twelfth Language Resources and Evaluation Conference (pp. 1235–1241). European Language Resources Association. https://aclanthology.org/2020.lrec-1.155

[9] Han, K., Xiao, A., Wu, E., Guo, J., XU, C., & Wang, Y. (2021). Transformer in Transformer. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, & J. W. Vaughan (Eds.), Advances in Neural Information Processing Systems (Vol. 34, pp. 15908–15919). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2021/file/854d9fca60b4bd07f9bb215d59ef5561-Paper.pdf

[10] Islam, S., Elmekki, H., Elsebai, A., Bentahar, J., Drawel, N., Rjoub, G., & Pedrycz, W. (2023). A Comprehensive Survey on Applications of Transformers for Deep Learning Tasks. ArXiv Preprint ArXiv:2306.07303.

[11] Mishra, P. (2023). Fine-Tuning Deep Learning Models Using PyTorch. In PyTorch Recipes (pp. 157–170). Apress. https://doi.org/10.1007/978-1-4842-8925-9_6

[12] M.R.Narasinga Rao, D., Venkatesh Prasad, V., Sai Teja, P., Zindavali, M., & Phanindra Reddy, O. (2018). A Survey on Prevention of Overfitting in Convolution Neural Networks Using Machine Learning Techniques. International Journal of Engineering & Technology, 7(2.32), 177. https://doi.org/10.14419/ijet.v7i2.32.15399

[13] Yong, H., Huang, J., Meng, D., Hua, X., & Zhang, L. (2020). Momentum Batch Normalization for Deep Learning with Small Batch Size (pp. 224–240). https://doi.org/10.1007/978-3-030-58610-2_14

[14] Omran, T., Sharef, B., Grosan, C., & Li, Y. (2023). The Impact of Data Augmentation on Sentiment Analysis of Translated Textual Data. 2023 International Conference on IT Innovation and Knowledge Discovery (ITIKD), 1–4. https://doi.org/10.1109/ITIKD56332.2023.10099851

[15] Wei, J., & Zou, K. (2019). EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 6381–6387. https://doi.org/10.18653/v1/D19-1670

[16] Xiao, B. (2024). Some basic knowledge of LLM: Parameters and Memory Estimation. https://medium.com/@baicenxiao/some-basic-knowledge-of-llm-parameters-and-memory-estimation-b25c713c3bd8

[17] Abdullah-All-Tanvir, Mahir, E. M., Huda, S. M. A., & Barua, S. (2020). A Hybrid Approach for Identifying Authentic News Using Deep Learning Methods on Popular Twitter Threads. 2020 International Conference on Artificial Intelligence and Signal Processing (AISP), 1–6. https://doi.org/10.1109/AISP48273.2020.9073583

[18] Karimi, Z. (2021). *Confusion Matrix*. 9073583