Leveraging Artificial Intelligence and Large Language Models for Fake Content Detection in Digital Media

Andriy Matviychuk[1,*,†], Vasyl Derbentsev[1,†], Vitalii Bezkorovainyi[1,†], Tetiana Kmytiuk[1,†] and Oleksii Hostryk[2,†]

[1] *Kyiv National Economic University named after Vadym Hetman, Beresteysky Ave. 54/1, 03057 Kyiv, Ukraine*
[2] *Odesa National Economic University, Preobrazhenskaya Str. 8, 65082 Odesa, Ukraine*

Abstract

The rapid proliferation of misinformation and fake news across online platforms has become a significant challenge, necessitating the development of advanced detection methods. This study explores the application of BERT-based models, including RoBERTa, DistilBERT, and XLM-RoBERTa, for the identification of fake news. Using diverse datasets (WELLFake, and PolitiFact) our approach involves fine-tuning these pre-trained models with minimal text preprocessing to preserve linguistic nuances. The models were evaluated based on their accuracy, F1-score, and computational efficiency, with experiments conducted on Google Colab using NVIDIA GPUs for acceleration. RoBERTa demonstrated the highest accuracy on the WELLFake dataset, while DistilBERT achieved the best performance on the more concise PolitiFact dataset, highlighting the importance of matching models to dataset characteristics. XLM-RoBERTa, with its multilingual capabilities, showed strong generalization on diverse data but faced challenges with domain-specific tasks. The results underscore that model selection should be tailored to the specifics of the dataset and available computational resources, offering valuable insights for deploying effective fake news detection systems.

Keywords

fake news detection, text classification, risk information, artificial intelligence, deep learning, natural language processing, BERT-like models

1. Introduction

The proliferation of fake content in electronic media has become a critical challenge in our increasingly digitized world. From misinformation and disinformation to sophisticated deepfakes, the spread of false or misleading content poses significant threats to social cohesion, democratic processes, and individual decision-making. As the volume and complexity of digital content continue to grow exponentially, traditional methods of fact-checking and content verification struggle to keep pace, necessitating the development of more advanced, automated approaches to identifying fake content [1, 2].

In recent years, the field of Natural Language Processing (NLP) has witnessed remarkable advancements, particularly in the domain of Large Language Models (LLMs). These sophisticated Artificial Intelligence (AI) systems, trained on vast corpora of text data, have demonstrated an unprecedented ability to understand and generate human-like text, making them promising candidates for tackling the fake content detection challenge. Models like CNN (Convolutional Neural Network), BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pre-

Information Technology and Implementation (IT&I-2024), November 20-21, 2024, Kyiv, Ukraine
* Corresponding author.
† These authors contributed equally.
matviychuk@kneu.edu.ua (A. Matviychuk); derbv@kneu.edu.ua (V. Derbentsev); retal.vs@kneu.edu.ua (V. Bezkorovainyi); kmytiuk.tetiana@kneu.edu.ua (T. Kmytiuk); alic@i.ua(O. Hostryk)
0000-0002-8911-5677 (A. Matviychuk); 0000-0002-8988-2526 (V. Derbentsev); 0000-0002-4998-8385 (V. Bezkorovainyi); 0000-0001-5262-856X (T. Kmytiuk); 0000-0001-6143-6797 (O. Hostryk)

trained Transformer), and their variants have set new benchmarks in various NLP tasks, including text classification, sentiment analysis, and question answering [3-6].

The potential of LLMs in identifying fake content lies in their capacity to capture subtle linguistic patterns, contextual nuances, and semantic relationships that may be indicative of fabricated or misleading information. By leveraging pre-trained models and fine-tuning them on specific datasets related to fake content, researchers and developers can create powerful tools for automated content authenticity verification [7]. This article explores the application of large language models, with a focus on BERT and its variations, in the detection of fake content across various electronic media platforms. We will delve into the process of adapting these pre-trained models to the specific task of fake content identification, discussing the methodology of retraining several last layers on custom datasets chosen for this purpose.

The approach of fine-tuning pre-trained models offers several advantages in the context of fake content detection. Firstly, it allows us to benefit from the rich language understanding already encoded in these models, which have been trained on diverse and extensive datasets. Secondly, it provides a more efficient and resource-effective method compared to training models from scratch, which is especially valuable when working with limited labelled data specific to fake content [8].

However, the application of LLMs in this domain is not without challenges. Issues such as model bias, the need for continual updating to keep pace with evolving disinformation tactics, and the ethical implications of automated content analysis must be carefully considered [9]. Moreover, the effectiveness of these models can vary depending on the type and source of fake content, necessitating a nuanced approach to model selection and fine-tuning.

Throughout this article, we will examine the architecture of the chosen pre-trained models, detail the process of dataset preparation and model fine-tuning, and present a comprehensive analysis of the results obtained. We will also discuss real-world applications, limitations of the current approach, and potential future developments in this rapidly evolving field.

By exploring the use of LLM and the detection of fake content, this study contributes to broader efforts to combat disinformation in the digital age. In an increasingly complex information environment, developing advanced AI-based tools for content verification is not only a technical challenge, but also an important measure to ensure the reliability and credibility of information.

It should be noted that Russia's ongoing military aggression against Ukraine has provided a vivid illustration of the power and danger of fake content in modern warfare and international relations. The onset of the full-scale phase of the hostilities has been marked by unprecedented levels of information warfare, with a flow of fakes and disinformation flooding social media platforms, news feeds, and messaging apps.

The dissemination of fake content in this context ranges from fabricated stories about Ukrainian aggression to doctored videos purporting to show military actions that never occurred. This barrage of false information has not only complicated the international community's understanding of the situation, but has also affected public opinion, potentially influencing policy decisions and humanitarian aid efforts.

The situation highlights the critical need for reliable, rapid detection systems for fake content, as the consequences of unverified disinformation in such high-stakes geopolitical scenarios can be severe and far-reaching.

The objective of our study is to develop a set of fake news identification models based on pre-trained BERT models by fine-tuning the last few layers, and compare their performance.

2. Literature review

With advancements in Machine Learning (ML), Deep Learning (DL), and LLMs, researchers have developed various methods to identify false information accurately. Recent studies in this domain have explored different approaches, ranging from traditional machine learning models to advanced neural networks, hybrid models, and even explainable AI. Recently have been published several

overview of ML and DL approaches in the field of identification of fake content in the digital media [10-13].

Harris et al. [10] explore the emergence of information pollution and the infodemic resulting from the widespread use of digital technologies on online social networks, blogs, and websites. They highlight the negative consequences of the malicious broadcast of misleading content, including social unrest, economic impacts, and threats to national security and user safety. The authors critically evaluate existing fake news detection (FND) methods, emphasizing the lack of multidisciplinary approaches and theoretical considerations in current research. They argue for a more comprehensive analysis of FND through various fields such as linguistics, healthcare, and communication, while also examining the potential of pre-trained transformer models for multilingual, multidomain, and multimodal FND. The authors suggest future research directions that focus on large, diverse datasets and the integration of human cognitive abilities with AI to combat fake news and AI-generated content.

Hu et al. [11] provide a comprehensive overview of fake news detection by analyzing its diffusion process through three intrinsic characteristics: intentional creation, heteromorphic transmission, and controversial reception. The authors classify existing detection approaches based on these characteristics and discuss the technological trends that are shaping this research field. They highlight the importance of designing effective and explainable detection mechanisms and offer insights into future research directions, helping to advance the understanding and development of fake news detection strategies.

Alghamdi et al. [12] present a comparative study of different approaches to fake news detection. The authors evaluate the performance of traditional ML methods such as Support Vector Machines (SVMs) and Random Forests (RFs) alongside more advanced DL models like CNN and Long Short-Term Memory (LSTM) network. Their research highlights the superiority of deep learning models, particularly LSTM, in capturing the sequential nature of text data and achieving higher accuracy in fake news detection. The study also emphasizes the importance of feature selection and engineering in improving model performance, suggesting that a combination of content-based and metadata features can lead to more robust detection systems.

Hamed et al. [13] offer a comprehensive review of fake news detection approaches, focusing on the challenges associated with datasets, feature representation, and data fusion. The authors critically analyze existing studies, highlighting the limitations of current datasets, which often lack diversity and real-world applicability. They discuss various feature representation techniques, from traditional bag-of-words models to more sophisticated word embeddings and contextual representations. The paper also explores the potential of multi-modal approaches that combine textual, visual, and social context information for more accurate fake news detection. The authors conclude by identifying key research gaps and suggesting future directions, including the need for more robust and diverse datasets, improved feature extraction methods, and the integration of explainable AI techniques to enhance the interpretability of FND models.

For example, in the article [14], the explainability of decision-making in the field of text analysis is ensured by the use of semiotic AI tools, namely fuzzy logic. In the article [15], explainable AI was implemented based on an artificial neural network, which provided the rationale for the formation of logical inference. Additional advantages in the interpretability of artificial intelligence can be provided by combining both such approaches, based on semiotic and biological principles of constructing AI systems and implemented in neuro-fuzzy hybrid systems, as shown in [16, 17].

It is also worth noting the studies that have focused on comparing traditional ML and DL approaches for fake news detection. Thus, authors of the review [18] compare the performance of such ML algorithms as Naïve Bayes, Logistic Regression, SVM, and RNNs. They noted that SVM and Naïve Bayes outperform the other models in terms of classification efficiency. This approach addresses the growing issue of misinformation on social media, where users often perceive content as reliable without verification.

In contrast, DL techniques have gained attention for their ability to automatically extract features from text data. Nasir et al. [19] employed CNNs and RNNs for fake news detection, showing that CNNs excel at capturing local patterns in text, while RNNs, particularly LSTM models, are better at understanding sequential information. The combination of these two models led to superior results.

Tipper et al. [20] provide a comprehensive review of video deepfake detection techniques using hybrid CNN-LSTM models. The paper systematically investigates feature extraction approaches and widely used datasets, while evaluating model performance across various datasets and identifying factors influencing detection accuracy. The authors here also compare CNN-LSTM models with non-LSTM approaches, discuss implementation challenges, and propose future research directions for improving deepfake detection.

Paka et al. [21] introduced Cross-SEAN, a semi-supervised neural attention model for detecting COVID-19 fake news on Twitter, leveraging both labelled and unlabelled data. Their approach, which incorporates external knowledge from trusted sources, achieved significant performance improvement over seven state-of-the-art models. Despite some limitations, such as potential biases in external knowledge, the model shows promising results, particularly with its real-time application in the Chrome-SEAN extension, designed to label fake tweets and collect user feedback for continuous improvement.

Recent studies show that the use of LLMs such as GPT and BERT has led to more refined approaches in fake news detection [22-25]. These models enhance the ability to understand the context, semantics, and intricate relationships within news articles, which are essential for distinguishing between truthful and deceptive content. By leveraging deep learning techniques, LLMs have significantly improved the accuracy and effectiveness of fake news detection systems, making them more robust in combating misinformation in the digital landscape.

For instance, Radhi et al. [22] examine the application of DL methods, including transformer-based models like BERT, to detect fake news. Their research highlights the growing impact of misleading content on social media platforms such as Facebook, Twitter, Instagram, and WhatsApp, and emphasizes the urgency of addressing the problem of fake news, particularly in the context of psychological warfare and revenue-driven clickbait.

Kaliyar et al. [23] propose FakeBERT, a BERT-based model that combines BERT with a CNN to handle ambiguity in news content. This model achieves a remarkable accuracy of 98.90%, outperforming existing models by using bidirectional training to capture semantic and long-distance dependencies, thus improving classification performance.

Similarly, Alnabhan and Branco [24] present BERTGuard, a multi-domain fake news detection system that employs a two-tiered approach for domain classification and domain-specific news validity verification. This system demonstrates its effectiveness through rigorous testing on various datasets and incorporates strategies to mitigate class imbalance, enhancing its reliability and generalizability.

Dhiman et al. [25] propose a novel framework called GBERT, combining GPT and BERT to tackle the problem of fake news detection. The model's high performance, achieving 95.30% accuracy and a 96.23% F1 score, underscores its potential to address the challenges posed by fake news in the digital era.

Overall, these diverse approaches underline the evolving nature of research in fake news detection. While traditional ML models still provide a foundation, the rapid advancements in deep learning, LLMs, and hybrid methods have expanded the capabilities to combat disinformation. The integration of explainable AI and adversarial training techniques ensures that these models remain both transparent and robust, helping to build trust in automated fake news detection systems.

3. Methodology

3.1. BERT-like models

In our study, the methodology focuses on leveraging a pre-trained BERT (introduced by Devlin et al. in 2018 [6]) and its modifications for fake news detection. BERT is a powerful transformer-based model known for its deep bidirectional nature, which allows it to understand the context of words in a text by looking both to the left and right of a given token.

Due to its ability to encode rich semantic information from large text corpora, BERT has been a popular choice for various NLP tasks, including text classification, sentiment analysis, and fake news detection. Here, the goal is to fine-tune BERT for classifying news articles into "real" or "fake" categories, aiming for accurate detection of misleading information.

At its core, BERT utilises a multi-layer bidirectional Transformer encoder. This bidirectional approach enables the model to consider context from both directions simultaneously, which is in stark contrast to traditional left-to-right language models. The standard BERT model comprises (BERT base) 12 transformer layers (encoders), 12 attention heads, and about 110 million parameters. The larger variant (BERT large) has 24 layers, 16 attention heads, and 340 million parameters (Fig. 1).

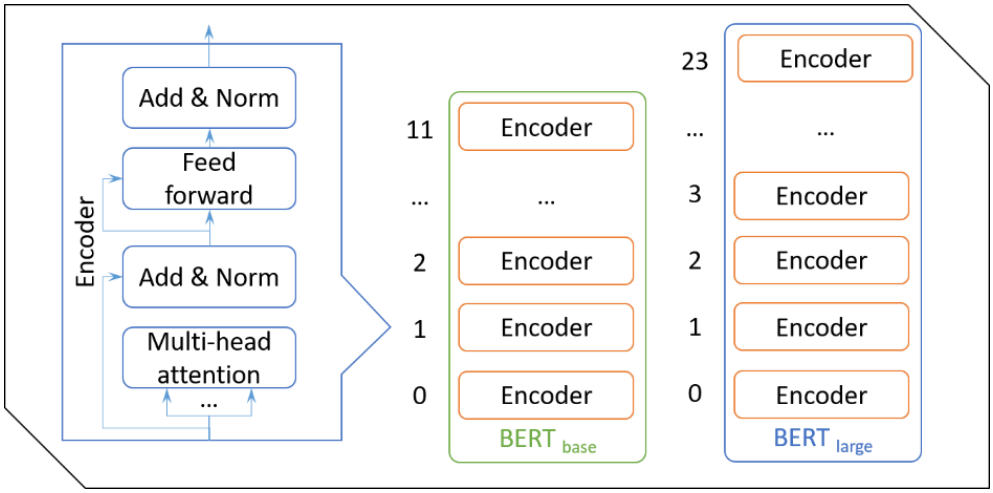

Figure 1: Simplified architecture of the BERT model (designed by the authors based on [6])

BERT consists of the following main components:

1. *Tokenizer.* BERT uses a WordPiece tokenizer that splits text into tokens, including subwords, to effectively handle rare words. This helps the model manage vocabulary more efficiently and capture the meaning of morphologically complex words.
2. *Embeddings.* BERT embeddings include:
   - Token embeddings: vectors that represent individual tokens.
   - Positional embeddings: encodes the position of each token in the sequence to capture word order.
   - Segment embeddings: differentiate between segments (sentences) within the input sequence, enabling the model to distinguish sentences in tasks like question answering.
3. *Encoders.* BERT consists of multiple layers of encoders, each containing:
   - Multi-head Self-Attention: this mechanism allows the model to focus on different parts of the input sequence, capturing dependencies between all tokens regardless of their distance from each other.
   - Feed-Forward Networks (FFN): each attention layer is followed by an FFN that applies non-linear transformations, enhancing the model's capacity to learn complex patterns.

- Residual connections and layer normalization: these are used to stabilize training and improve gradient flow through the network.

The input representation in BERT is a combination of three embeddings: token embeddings, segment embeddings, and position embeddings. Token embeddings represent individual words or subwords, segment embeddings differentiate between pairs of sentences, and position embeddings provide information about the token's position in the sequence. These embeddings are summed to produce the final input representation.

BERT's transformer layers consist of multi-head self-attention mechanisms and feed-forward neural networks. The self-attention mechanism allows the model to assign varying importance to different words in the input when processing each word, capturing complex relationships within the text. The feed-forward networks then refine this processed information, applying non-linear transformations that enhance the model's capacity to recognize and learn complex patterns.

BERT's pre-training process involves two novel unsupervised tasks. The first is Masked Language Modeling (MLM), where the model attempts to predict randomly masked tokens in the input sequence (Fig. 2).

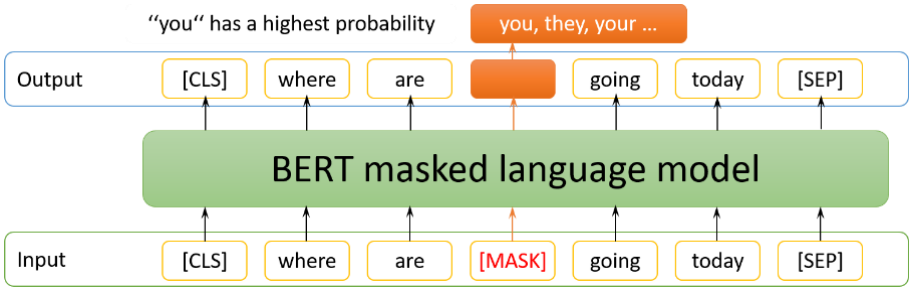

Figure 2: BERT masked language model predictions (designed by the authors based on [6])

This task forces the model to consider context from both directions, enhancing its bidirectional understanding. The second task is Next Sentence Prediction (NSP), where the model learns to predict whether two sentences naturally follow each other, fostering a grasp of the relationships between sentences.

One of BERT's key strengths is its ability to generate contextualised word embeddings. Unlike static word embeddings, BERT's representations for a given word can vary depending on the surrounding context, capturing nuanced word usage and polysemy effectively.

The fine-tuning process allows BERT to be adapted to a wide range of downstream tasks. By adding task-specific layers to the pre-trained BERT model and fine-tuning on task-specific data, researchers can achieve state-of-the-art results on various NLP tasks, including question answering, sentiment analysis, text classification, summarisation, and named entity recognition.

BERT's impact extends beyond its architecture. It has sparked a new paradigm in NLP, demonstrating the power of unsupervised pre-training on large corpora followed by supervised fine-tuning. This approach has led to the development of numerous BERT variants and inspired new research directions in contextual language modeling.

Since its release, various modifications and improvements have been introduced to address specific limitations and further enhance the model's performance on a range of NLP tasks. Some of the notable modifications include RoBERTa, DistilBERT, and XLM-RoBERTa, each designed with unique features to optimize BERT's efficiency, scalability, and multilingual capabilities.

RoBERTa (a Robustly Optimized BERT pretraining Approach by Liu et al. [26]) was developed to address some of the original training challenges in BERT. RoBERTa builds upon the BERT architecture by using more training data and a larger number of training steps, along with other optimizations like removing the Next Sentence Prediction objective.

Instead of focusing on the relationships between sentence pairs, RoBERTa concentrates purely on the Masked Language Modeling objective, which has shown to be more effective for a wide range of downstream NLP tasks. Additionally, RoBERTa utilizes dynamic masking, which allows for different masked tokens during each epoch, offering a more diverse learning experience. As a result, RoBERTa has consistently outperformed BERT on various benchmarks, making it a preferred choice for tasks like text classification and sentiment analysis.

DistilBERT (by Sanh et al. [27]) is another significant modification aimed at making BERT lighter and faster while retaining most of its performance capabilities. Developed using a technique called knowledge distillation, DistilBERT is approximately 60% of the size of BERT, making it faster during both training and inference. In knowledge distillation, a smaller model (the student model) is trained to reproduce the behavior of a larger pre-trained model (the teacher model).

This process enables the student model, DistilBERT in this case, to learn a more compact representation of the language while preserving 97% of BERT's language understanding abilities. DistilBERT's smaller size makes it particularly suitable for scenarios where computational resources are limited or where real-time performance is critical, such as in mobile or edge computing applications.

XLM-RoBERTa (Cross-lingual Language Model) is an extension of the BERT architecture designed for multilingual tasks, building on the success of both RoBERTa and the earlier XLM. XLM-RoBERTa is pre-trained on a large-scale multilingual corpus covering over 100 languages, making it capable of handling cross-lingual understanding and translation tasks more effectively.

The model learns representations that are common across languages, which allows it to perform well on tasks involving low-resource languages by transferring knowledge from high-resource languages.

Proposed by Lan et al. [28], ALBERT (A Lite BERT) addresses BERT's limitations of model size and training time. It introduces parameter-reduction techniques like factorized embedding parameterization and cross-layer parameter sharing. Despite having fewer parameters, ALBERT achieves state-of-the-art results on several benchmarks while being more efficient.

Developed by Clark et al. [29], ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) introduces a new pre-training task where the model learns to distinguish between real input tokens and fake tokens generated by a small masked language model. This approach is more sample-efficient than BERT's masked language modeling, allowing ELECTRA to achieve strong performance with less computation.

Each of these modifications brings unique strengths to the BERT family of models. RoBERTa's focus on robust training has made it highly accurate, but it comes with increased computational requirements due to the larger dataset and training time. DistilBERT addresses the issue of computational expense by providing a smaller, faster alternative, making it a practical option for deployment in environments where resources are constrained. XLM-RoBERTa, meanwhile, opens the door to advanced multilingual applications, offering a model that can understand and process a variety of languages effectively.

In this paper we used both BERT base model and its modifications (RoBERTa, DistilBERT, and XLM-RoBERTa).

3.2. BERT-based classification pipeline

The pipeline starts with data collection and pre-processing, where text data is cleaned and tokenized. This involves removing any special characters, URLs, and unnecessary whitespace. Tokenization is done using the BERT tokenizer, which converts the text into a format that the BERT model can handle, specifically by converting words into tokens, adding special tokens like [CLS] and [SEP], and creating attention masks that help the model focus on relevant parts of the input data.

The data is then split into training, validation, and test sets to ensure that the model can be properly evaluated. The pre-trained BERT-like model is fine-tuned on the training dataset. Fine-

tuning involves using the general knowledge gained during BERT-like initial training on a large corpus and tailoring it to the specific task of detecting fake news. In this study, we freeze all layers of the model except the last few encoders and the soft max classifier, which are retrained on our dataset (Fig. 3).

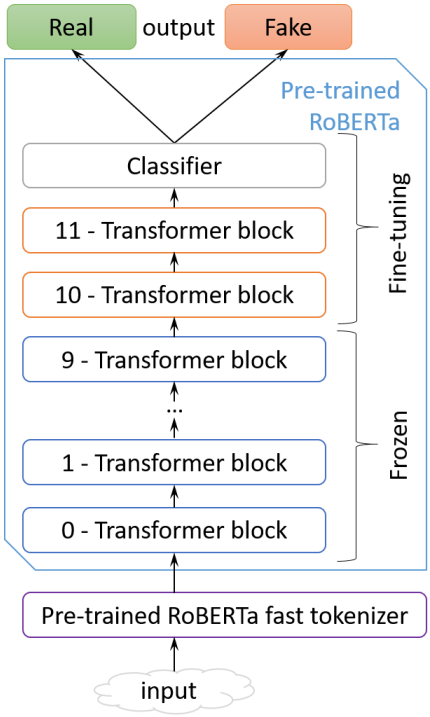

Figure 3: RoBERTa fine-tuning model

The model is trained on the labelled dataset, adjusting its weights using the cross-entropy loss by the Adam optimizer. During training, the learning rate is carefully controlled using a scheduler, starting from a small value to ensure stable updates and avoid overfitting.

Our study applies early stopping based on the validation loss to avoid training for too many epochs, which can lead to overfitting. This way, the model is more likely to generalize well on unknown data. The evaluation of the fine-tuned model is performed using metrics such as accuracy and F1-score.

These metrics provide a comprehensive understanding of the model's performance in detecting fake news. The F1-score, as the harmonic mean of precision and recall, provides a balanced measure of the model's performance. Confusion matrices are also used to visualize the performance, offering insight into the types of errors the model may make.

Since BERT was pre-trained on a large corpus, it can leverage the linguistic patterns learned during this general training, thereby requiring less labelled data for the specific task of fake news detection. This is especially advantageous given the scarcity of high-quality labelled fake news datasets. BERT's fine-tuning allows it to adapt to the nuances of fake news language without starting from scratch, making this approach efficient and effective.

4. Datasets and software implementation

4.1. Datasets

To effectively train and test BERT-type models for fake news detection, we utilized a variety of datasets that are widely recognized in the field. These datasets (FakeNewsNet, particular, PolitiFact subset, and WELFake Dataset) offer diverse contexts and sources of fake news, allowing for robust model evaluation. Each dataset has unique characteristics that contribute to a comprehensive

training and testing process, helping to ensure that the models generalize well across different types of misinformation.

FakeNewsNet (PolitiFact) [30] is a well-established dataset that combines news content with social context to facilitate the study of fake news detection. It includes news articles verified by the PolitiFact fact-checking website, where each article is classified as either "fake" or "real" based on professional verification. This dataset not only includes the text of the news articles but also metadata such as user engagement and social media activity around each news item.

The social context allows models to capture the diffusion patterns of fake news, which is crucial for understanding how misinformation spreads online. By training BERT-type models on the textual content supplemented with metadata, we aimed to enhance the models' ability to detect fake news by considering both linguistic features and social spread patterns.

We used PolitiFact subset, which contains around 1,200 records, with the average length of the text being about 15 words, offering a moderate level of detail for each news item. This shorter length allows the BERT models to focus on concise stylistic and contextual indicators of fake news.

WELFake (Web Evaluated Fake News) [31] is significantly larger, with over 70,000 news articles. Its structure is simpler, focusing primarily on the text, title of the articles and a label field that classifies each article as "fake" or "real". This minimal structure makes WELFake an ideal dataset for large-scale training, enabling models to learn from a vast variety of textual examples. Despite its large size, the dataset is not entirely balanced, with a higher number of fake news records compared to real news. This imbalance requires careful consideration during model training, such as using class weighting or oversampling techniques to prevent the model from overfitting to the majority class. The average length of texts in WELFake is around 540 words, which provides enough data for models to learn linguistic patterns while ensuring efficient training time due to shorter text sequences. This shorter length allows the BERT models to focus on concise stylistic and contextual indicators of fake news.

By using these datasets, we were able to train BERT-type models with a diverse range of textual inputs and associated features. This diversity ensures that the models are not only capable of recognizing the typical writing styles and topics of fake news but also understand how false information is often framed within a broader social context. Moreover, combining datasets with large-scale examples like WELFake and more specific examples like those in PolitiFact ensures a balance between the volume of data and the richness of context. This approach helps improve the generalization abilities of the models, making them better suited to real-world applications where misinformation can take on many different forms and reach audiences through various channels.

Thus, in the PolitiFact dataset after removing duplicates there are a few short records (about 1,000 with an average length of 15 tokens), and in the WELLFake dataset there are about 50,000 with an average length of 540 tokens. Such diversification will allow us to test the performance of BERT-similar models in fundamentally different conditions.

4.2. Software

In our study, we utilized a variety of software tools and libraries to train, test, and analyze the BERT-type models in identifying fake news. These tools facilitated the entire pipeline from data pre-processing and model training to evaluation and visualization of results. The combination of these software solutions allowed us to leverage state-of-the-art techniques and streamline our workflow.

PyTorch served as the core library for building, training, and fine-tuning our deep learning models. As an open-source deep learning framework, PyTorch offers dynamic computational graphs and an intuitive API, making it suitable for implementing complex BERT-like models. It also provided robust support for GPU acceleration, which was crucial for training large language models efficiently on the substantial datasets we used. The ease of integrating PyTorch with pre-trained models through the Hugging Face Transformers library allowed us to fine-tune these models specifically for the task of fake news detection.

Pandas played a critical role in managing and pre-processing our datasets. Given the size and complexity of datasets, Pandas' capabilities for data manipulation and analysis were invaluable. We used it to load, filter, and clean data, ensuring that the text fields were formatted properly for input into the models. Pandas also allowed us to explore dataset characteristics, such as class distribution and text length, which helped guide our approach to model training and evaluation. Its versatility in handling various data formats, including CSV and JSON, streamlined the process of preparing our data.

Python 3.8 served as the primary programming language for this project, owing to its simplicity, versatility, and extensive ecosystem of libraries. Python's flexibility enabled us to integrate diverse tools seamlessly, from data pre-processing with Pandas to model training with PyTorch. Additionally, Python's wide range of libraries for data visualization, like Matplotlib and Seaborn, made it easier to conduct analysis of findings and interpret model performance. Python's extensive community support and documentation further facilitated the smooth implementation of cutting-edge methods.

Scikit-Learn was used for a range of pre-processing and evaluation tasks. This included splitting datasets into training and testing samples, calculating various performance metrics like accuracy, precision, recall, F1-score, and generating confusion matrices for deeper insights into model predictions. Scikit-Learn's easy-to-use API allowed us to quickly compare different models and pre-processing strategies, ensuring that we could iteratively refine our approach to achieve the best results.

Seaborn and Matplotlib were essential for data visualization throughout the project. Seaborn, with its high-level interface, was used to create aesthetically pleasing and informative plots, such as histograms of text lengths and confusion matrices, which helped us understand the distribution of data and model performance at a glance. Matplotlib provided additional customization capabilities, allowing us to tailor visualizations to our specific needs, such as adjusting axis scales or highlighting specific data points.

Additionally, we leveraged the *Hugging Face Transformers library* to access pre-trained BERT-type models and adapt them for our task. This library enabled us to import and fine-tune models with minimal efforts, allowing us to focus on the nuances of the fake news detection problem rather than the complexities of implementing models from scratch. The ease of integrating these models with PyTorch through the Transformers library made it possible to quickly experiment with different architectures and configurations.

We leveraged *Google Colab* as the primary development environment for training and evaluating our models. Google Colab is a cloud-based platform that offers a Jupiter notebook interface, providing a powerful and convenient setting for executing Python code and running deep learning experiments. One of the key advantages of Google Colab is its access to free GPUs and TPUs, which significantly accelerated the training process for our BERT-type models.

5. Experimental setup

5.1. Final hyperparameter settings

The final hyperparameter settings are presented in Table 1. These settings describe our setup for fine-tuning pre-trained BERT-type models. A batch size of 16 provides a balance between memory usage and training speed that is suitable for most GPUs. Input sequences are limited to 128 (64 for PolitiFact) tokens, which is sufficient for our datasets. The model is trained for 5 epochs, allowing it to train on the data multiple times without overfitting.

The learning rate is small, allowing careful adjustment of the pre-trained weights. For optimization, the AdamW optimizer is used, an improved version of Adam that properly implements weight decay. CrossEntropyLoss serves as a loss function that is standard for many classification tasks in natural language processing.

Table 1
Final hyperparameter settings

Hyperparameters	Description	Value
Batch Size	Number of samples processed per batch	16
Max Sequence Length	Maximum length of the input sequence	64 (128)
Number of Epochs	Number of complete passes through the training dataset	5
Learning Rate	Learning rate used by the optimizer	$2{,}00 \cdot 10^{-5}$
Optimizer	Optimization algorithm	AdamW
Loss Function	Loss function used for training	CrossEntropyLoss
Device	Computational device used	GPU (NVIDIA L4)
Training Environment	Platform used for model training	Google Colab
Validation Split	Proportion of data used for validation	20%

To prevent overfitting, a dropout rate of 0.3 is applied, randomly deactivating 30% of neurons during training. The optimization process takes advantage of GPU acceleration, in particular NVIDIA L4, which significantly speeds up the computation compared to CPU. Google Colab serves as a training environment, offering free access to GPUs for model development.

5.2. Text preprocessing

Text preprocessing for our fake identification task was intentionally minimal, leveraging the robust capabilities of BERT-type models. These models are pre-trained on vast corpora of unrefined text, allowing them to handle raw input effectively. This approach preserves the natural structure and nuances of the text, which can be crucial for detecting subtle indicators of fake content.

The primary preprocessing step was performed by the tokenizer specific to each BERT model variant. These tokenizers are designed to break down text into subword units, handling out-of-vocabulary words and maintaining semantic relationships. The tokenization effectively translates raw text into a format that BERT models can process, without losing important linguistic information.

Our preprocessing pipeline focused mainly on preparing the data structure for input into the model. This included binary encoding of labels, transforming the classification targets into a format suitable for machine learning. In addition, duplicates and data with gaps were removed.

We also concatenated various fields related to each piece of content, such as the author's name, the main text of the article, its title, and the URL. This concatenation allows the model to consider all relevant information simultaneously, potentially capturing relationships between different aspects of the content that might indicate its authenticity or lack thereof.

By keeping preprocessing minimal, we aimed to reveal the sophisticated language understanding capabilities of BERT models. This approach allows the models to work with text that closely resembles what it encountered during the pre-training phase, potentially improving its ability to detect nuanced signals of fake content across various writing styles and formats.

5.3. Evaluation metrics

To compare forecasting performance of the proposed models we used Accuracy metric and F1-score. Accuracy characterizes the share of correct answers of the classifier and can be calculated as

$$Accuracy = \frac{TP + TN}{P + N} \times 100\%,$$

where *TP* and *TN* are the number of correctly estimated positive (articles with fake news) and negative (articles without fakes) classes, respectively; *P* and *N* are the actual number of representatives of each class, respectively.

The F1-score provides a balanced measure of a model's performance, particularly when the dataset is imbalanced, i.e. when the number of positive and negative instances is significantly different. F1-score is calculated as:

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall},$$

where

$$Precision = \frac{TP}{TP + FP}, Recall = \frac{TP}{TP + FN},$$

and *FP, FN* are false positive (predicting fake news when there is none) and false negative (assessing news as real when it is fake) classes, respectively.

Note that since we are primarily interested in the correct identification of fakes, we have chosen them as a positive class (label 1).

We also calculated Confusion Matrix, which provides a comprehensive view of the model's performance, allowing for the calculation of various metrics and providing insights into the types of errors the model is making. It's particularly useful for understanding the compromises between different types of misclassifications and for fine-tuning the model to meet specific performance criteria.

5.4. Empirical results

The results presented in Tables 2 and 3 highlight the performance of various BERT-type models on the WELLFake and PolitiFact datasets, providing insights into their strengths and weaknesses in the context of fake news detection. The metrics of accuracy and F1-score, along with training time and the number of trainable parameters, provide a basis for a comprehensive comparison of these models.

Table 2
Classification performance for WELLFake dataset

Model	Number of trainable parameters	Training time, sec	Accuracy	F1-score
BERT base	7,088,641	4733	0.985	0.985
RoBERTa	7,679,233	**2025**	**0.998**	**0.994**
DistilBERT	7,680,002	4410	0.985	0.985
XLMRoBERTa	7,680,002	2048	0.994	0.991

RoBERTa emerges as the best classifier with an accuracy of 0.998 and an F1-score of 0.994 on the WELLFake dataset. This suggests that RoBERTa is particularly adept at capturing the nuances of the language used in fake news, allowing it to make very accurate predictions. The relatively short training time of 2025 seconds further highlights the effectiveness of RoBERTa, indicating that it can quickly process and adapt to the dataset, making it a strong candidate for real-world applications where both accuracy and speed are important.

XLM-RoBERTa also demonstrated strong performance with an accuracy of 0.994 and an F1-score of 0.991. XLM-RoBERTa's performance indicates that its multilingual pre-training allows it to effectively handle the diverse linguistic features present in the WELLFake dataset. However, despite the high accuracy, it took slightly longer to train (2048 seconds) compared to RoBERTa.

BERT base and DistilBERT achieved an accuracy of 0.985 with a corresponding F1-score. While they performed well, their accuracy did not reach the accuracy of RoBERTa or XLM-RoBERTa. This suggests that while the basic BERT architecture can effectively classify fake news, the additional fine-tuning and optimization present in RoBERTa and XLM-RoBERTa provide a noticeable advantage. Moreover, these two models took almost twice as long to train. It should be noted that

the lighter DistilBERT architecture did not contribute to significant reduction in training time (it took 4410 seconds compared to 4733 for BERT base), which does not make it an efficient model in terms of computing resources.

On the PolitiFact dataset, the performance landscape shifts, as shown in Table 3. Here, DistilBERT outperforms the other models, achieving an accuracy of 0.917 and an F1-score of 0.931. This result is particularly noteworthy because it demonstrates that DistilBERT, despite being a lighter and more compact version of BERT, can achieve higher accuracy on smaller datasets. Its reduced the number of trainable parameters makes it easier to train and adapt, especially when computational resources are a constraint. The relatively short training time of 9.8 seconds further underscores its efficiency.

Table 3
Classification performance for PolitiFact dataset

Model	Number of trainable parameters	Training time, sec	Accuracy	F1-score
BERT base	7,088,641	12.6	0.901	0.912
RoBERTa	7,679,233	12.2	0.891	0.891
DistilBERT	7,680,002	**9.8**	**0.917**	**0.931**
XLMRoBERTa	7,680,002	11.1	0.872	0.883

BERT base followed DistilBERT with an accuracy of 0.901 and an F1-score of 0.912. This performance suggests that the original BERT architecture remains highly effective for fake news detection, particularly when fine-tuned on a specific dataset like PolitiFact. However, BERT's longer training time of 12.6 seconds compared to DistilBERT reflects the additional computational demands of its more complex architecture.

RoBERTa, which excelled on the WELLFake dataset, achieved an accuracy of 0.891 and an F1-score of 0.891 on PolitiFact. This indicates that while RoBERTa is highly effective with larger datasets like WELLFake, it may not generalize as well to smaller datasets like PolitiFact without further fine-tuning. Its training time was slightly lower than BERT, at 12.2 seconds, suggesting some computational efficiency, but it also had lower accuracy.

XLM-RoBERTa achieved the lowest accuracy on the PolitiFact dataset at 0.872, with an F1-score of 0.883. This could be due to its design, which is optimized for multilingual tasks rather than domain-specific datasets like PolitiFact. Although it is highly versatile across different languages and contexts, this versatility may result in decreased performance when applied to a narrower task. The training time for XLM-RoBERTa was also substantial at 11.1 seconds, indicating that it is not the most efficient choice for this particular dataset.

Figures 4, 5 show the loss and accuracy graphs for the best models for the datasets we used, and Figures 6, 7 present the confusion matrices for these models.

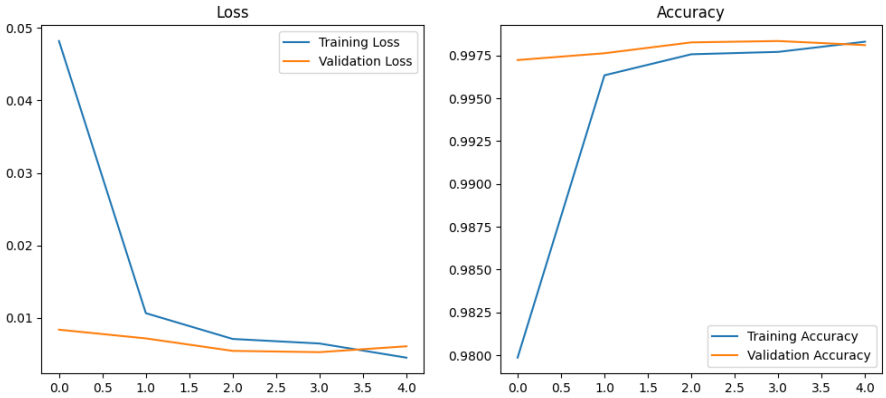

Figure 4: RoBERTa Loss and Accuracy graphs for WELLFake dataset

The graphs in Figure 4 illustrate the RoBERTa model's learning progress over the course of training epochs. The loss curve shows a steady decline, indicating that the model is successfully minimizing classification error as training proceeds. Simultaneously, the accuracy graph rises, reflecting the model's increasing ability to correctly classify fake news instances. The convergence of the loss and accuracy curves suggests that the model has effectively learned on the training data without significant signs of overfitting. The smoothness of the curves highlights the stability of RoBERTa during training, which contributes to its high performance on the WELLFake dataset.

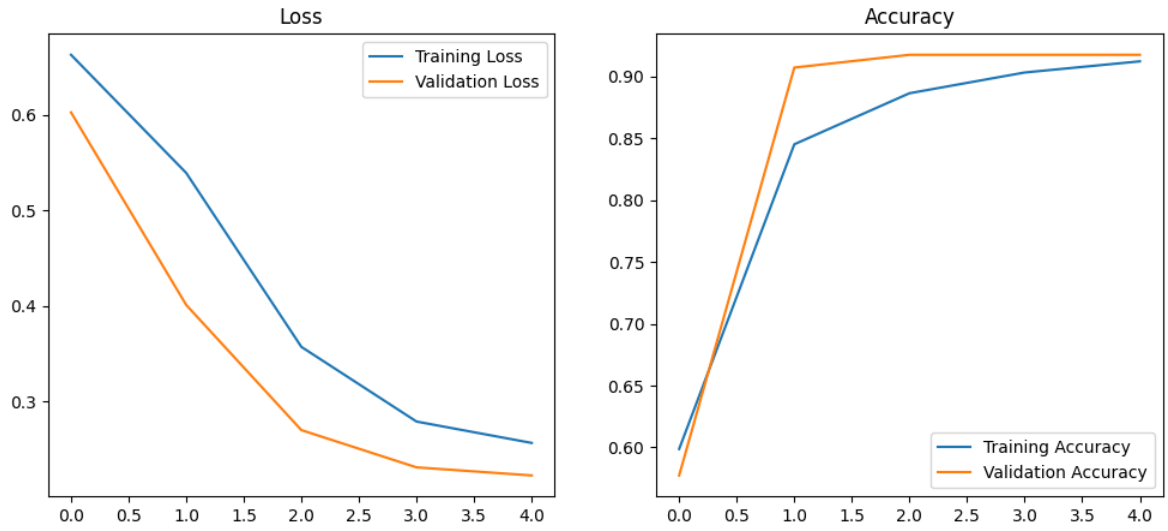

Figure 5: DistilBERT Loss and Accuracy graphs for PolitiFact dataset

Figure 5 shows the loss and accuracy graphs for the DistilBERT model on the PolitiFact dataset. Compared to RoBERTa, the DistilBERT model's loss decreases at a faster rate, indicating a more rapid adaptation to the training data. The accuracy also increases steadily, suggesting that DistilBERT quickly learns to distinguish between real and fake news articles. Despite the smaller architecture of DistilBERT, the model achieves high accuracy after a few epochs, which makes it particularly suitable for scenarios where computational resources or training time are limited. The relatively sharp drop in loss and corresponding rise in accuracy suggest that the model efficiently utilizes the information from the PolitiFact dataset.

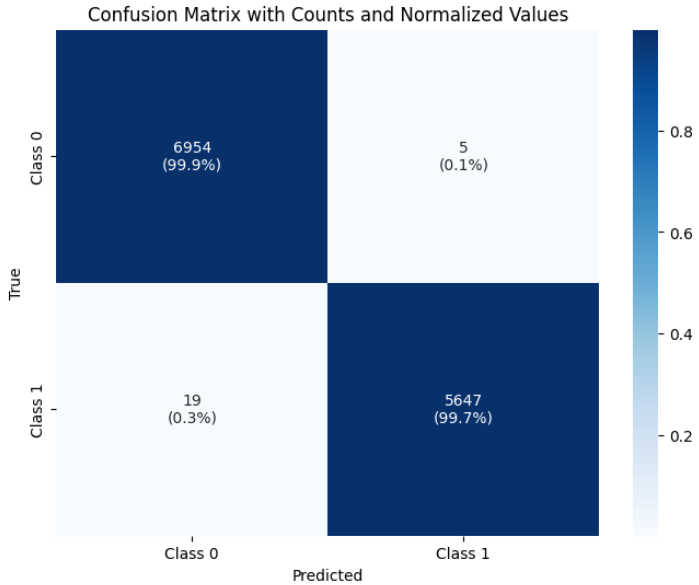

Figure 6: RoBERTa Confusion Matrix for WELLFake dataset

Figure 6 displays the confusion matrix for the RoBERTa model's performance on the WELLFake dataset. The confusion matrix provides a detailed view of the model's classification accuracy, including the *TP, TN, FP,* and *FN*. RoBERTa demonstrates a strong ability to correctly classify both real and fake news instances, with high counts in the *TP* and *TN* cells. The minimal number of misclassifications suggests that RoBERTa's understanding of linguistic features is effective in discerning deceptive content. This detailed insight into the types of errors made by the model is crucial for understanding the model's strengths in dealing with a large and diverse dataset like WELLFake.

Figure 7 provides the confusion matrix for the DistilBERT model on the PolitiFact dataset.

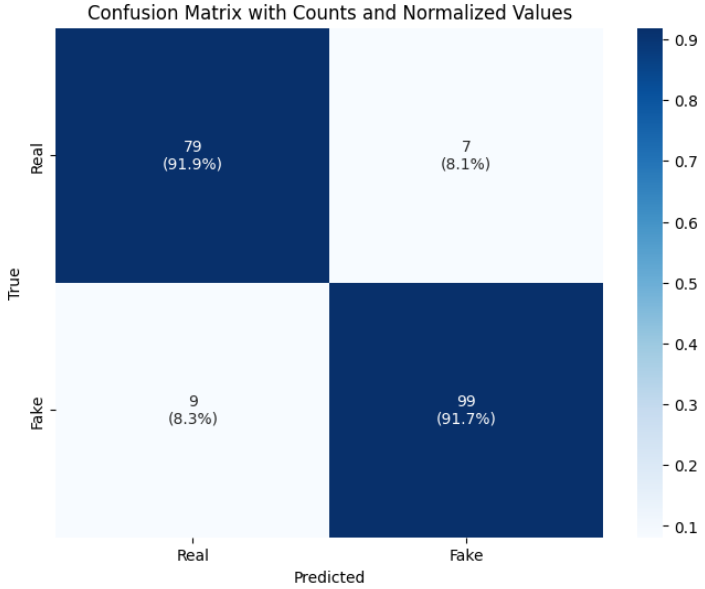

Figure 7: DistillBERT Confusion Matrix for PolitiFact dataset

The matrix in Figure 7 reveals that DistilBERT is accurate in identifying fake news, as reflected by the high *TP* rate. The model's ability to minimize *FN* is crucial in a context where failing to identify fake news can have significant consequences. The overall balance in correct classifications across both classes reflects the model's adaptability to the shorter and more concise news articles, typical of the PolitiFact dataset.

6. Discussion and conclusion

The results of our study illustrate the strengths and compromises of different BERT-based models in the context of fake news detection. RoBERTa demonstrated exceptional efficiency on the WELLFake dataset, achieving an accuracy of 0.998, highlighting its ability to handle complex and diverse text data. Its robust training process, which focuses heavily on the masked language modeling task, allows RoBERTa to capture subtle linguistic cues and contextual relationships that are often indicative of fake news. This makes RoBERTa a suitable model for applications where high accuracy is paramount, even if it comes at the cost of increased computational requirements.

DistilBERT, on the other hand, excelled on the smaller PolitiFact dataset, where it achieved an accuracy of 0.917. Its lightweight architecture, derived from the knowledge distillation process, enables it to learn efficiently from fewer data points while maintaining a high level of accuracy. This makes DistilBERT an ideal choice in scenarios where computational resources are limited, such as real-time fake news detection on edge devices. The model's rapid convergence and lower training time also make it more practical for applications that require quick deployment and frequent retraining.

However, the study also highlights certain limitations associated with each model. While RoBERTa offers superior accuracy on larger datasets like WELLFake, its performance on the smaller PolitiFact dataset was relatively low, with an accuracy of 0.891. This suggests that the model's complexity might require further fine-tuning to adapt to datasets with shorter text lengths and less diverse content.

XLM-RoBERTa's results provide additional insights into the role of multilingual models in fake news detection. Its high accuracy on the WELLFake dataset (0.994) suggests that cross-lingual training can enhance a model's ability to generalize across diverse linguistic styles. However, its relatively lower performance on the domain-specific PolitiFact dataset (accuracy of 0.872) indicates that models optimized for multilingual capabilities may not always perform best on specific, monolingual datasets without additional fine-tuning. This points to a potential compromise between multilingual versatility and domain-specific accuracy that researchers must consider when selecting models for fake news detection.

Overall, the comparison between these BERT-based models suggests that there is no one-size-fits-all solution for fake news detection. The choice of model depends largely on the characteristics of the dataset, the computational resources available, and the specific requirements of the application. For large-scale fake news detection tasks where accuracy is critical, RoBERTa is likely the most effective choice. For environments where speed and resource efficiency are priorities, such as mobile platforms or real-time applications, DistilBERT provides a viable alternative with its compact structure and faster training time.

Another key finding of our study is the importance of dataset diversity and structure in influencing model performance. The WELLFake dataset, with its large volume and longer average text length, allowed RoBERTa and XLM-RoBERTa to excel by leveraging their deeper architectures and advanced contextual understanding. Meanwhile, the PolitiFact dataset, characterized by shorter text samples, contributed to the effectiveness of DistilBERT in learning from more concise linguistic patterns. These differences emphasize the need for tailored approaches in selecting models for fake news detection, depending on the dataset's nature.

The study also underscores the role of minimal preprocessing in leveraging the strengths of BERT-based models. By allowing the models to handle raw text inputs, we preserved linguistic nuances that are critical for distinguishing fake news. This approach highlights the power of pre-trained models in adapting to specific tasks without extensive preprocessing, making them versatile tools for a wide range of applications in the field of NLP.

In conclusion it should be noted, that proposed approach to detecting fake content in digital media based on fine-tuned models such as BERT focuses on understanding linguistic nuances and contextual relationships in text. However, these models do not directly check the factual content of claims against external databases or sources. Instead, they work by detecting hidden linguistic features and patterns commonly associated with fake or misleading information.

This approach is advantageous in situations where external fact-checking is either impossible or time-consuming, but it also introduces certain limitations as the models rely heavily on linguistic cues rather than external verification.

Thus, the findings of this study contribute to the broader effort of developing reliable AI-driven tools for combating misinformation, which is a critical need in today's digital information landscape. Future research could explore further fine-tuning techniques and hybrid approaches that combine the strengths of multiple models to create even more robust solutions for fake news detection.

Acknowledgements

This paper is a part of the project "iResilience: Strengthening democratic practices and media literacy", which is funded by the Swedish Institute within the Baltic Sea Neighbourhood Programme (project No. 00152/2024).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] D. M. Lazer, M. A. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, ... J. L. Zittrain, The science of fake news, Science 359(6380) (2018) 1094-1096. doi:10.1126/science.aao2998.

[2] K. Shu, A. Sliva, S. Wang, J. Tang, H. Liu, Fake News Detection on Social Media: A Data Mining Perspective, ACM SIGKDD Explorations Newsletter 19(1) (2017) 22-36. doi:10.1145/3137597.3137600.

[3] V. Derbentsev, V. Bezkorovainyi, R. Akhmedov, Machine learning approach of analysis of emotional polarity of electronic social media, Neuro-Fuzzy Modeling Techniques in Economics 9 (2020) 95-137. doi:10.33111/nfmte.2020.095.

[4] V. Derbentsev, V. Bezkorovainyi, A. Matviychuk, O. Pomazun, A. Hrabariev, A. Hostryk, A comparative study of deep learning models for sentiment analysis of social media texts, CEUR Workshop Proceedings 3465 (2023) 168–188. URL: https://ceur-ws.org/Vol-3465/paper18.pdf.

[5] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, ... D. Amodei, Language Models are Few-Shot Learners, arXiv (2020). URL: https://arxiv.org/abs/2005.14165.

[6] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, arXiv (2018). URL: https://arxiv.org/abs/1810.04805.

[7] R. Zellers, A. Holtzman, H. Rashkin, Y. Bisk, A. Farhadi, F. Roesner, Y. Choi, Defending Against Neural Fake News, arXiv (2019). URL: https://arxiv.org/abs/1905.12616.

[8] J. Howard, S. Ruder, Universal Language Model Fine-tuning for Text Classification, arXiv (2018). URL: https://arxiv.org/abs/1801.06146.

[9] E. M. Bender, T. Gebru, A. McMillan-Major, S. Shmitchell, On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?, in: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, ACM Press, Canada, 2021, pp. 610-623. doi:10.1145/3442188.3445922.

[10] S. Harris, H. J. Hadi, N. Ahmad, M. A. Alshara, Fake news detection revisited: An extensive review of theoretical frameworks, dataset assessments, model constraints, and forward-looking research agendas, Technologies 12(11) (2024) 222. doi:10.3390/technologies12110222.

[11] B. Hu, Z. Mao, Y. Zhang, An overview of fake news detection: From a new perspective, Fundamental Research (2024). doi:10.1016/j.fmre.2024.01.017.

[12] J. Alghamdi, Y. Lin, S. Luo, A comparative study of machine learning and deep learning techniques for fake news detection, Information 13(12) (2022) 576. doi:10.3390/info13120576.

[13] S. K. Hamed, M. J. Ab Aziz, M. R. Yaakub, A review of fake news detection approaches: A critical analysis of relevant studies and highlighting key challenges associated with the dataset, feature representation, and data fusion, Heliyon 9(10) (2023) e20382. doi:10.1016/j.heliyon.2023.e20382.

[14] H. Melnyk, V. Melnyk, Enhancing Mood Detection in Textual Analysis through Fuzzy Logic Integration, in: 2024 14th International Conference on Advanced Computer Information Technologies, ACIT, IEEE, Ceske Budejovice, Czech Republic, 2024, pp. 23-26, doi:10.1109/ACIT62333.2024.10712628.

[15] V. Hraniak, V. Mazur, V. Matvijchuk, Artificial neural-like network as a basis for forming logical conclusions in systems of exceptional complexity, Neuro-Fuzzy Modeling Techniques in Economics 9 (2020) 65-94. doi:10.33111/nfmte.2020.065.

[16] S. Kozlovskyi, P. Syniehub, A. Kozlovskyi, R. Lavrov, Intellectual capital management of the business community based on the neuro-fuzzy hybrid system, Neuro-Fuzzy Modeling Techniques in Economics 11 (2022) 25-47. doi:10.33111/nfmte.2022.025.

[17] A. Matviychuk, O. Lukianenko, I. Miroshnychenko, Neuro-fuzzy model of country's investment potential assessment, Fuzzy economic review 24(2) (2019) 65-88. doi:10.25102/fer.2019.02.04.

[18] A. A. Tanvir, E. M. Mahir, S. Akhter, M. R. Huq, Detecting fake news using machine learning and deep learning algorithms, in: 2019 7th International Conference on Smart Computing & Communications, ICSCC, IEEE, Sarawak, Malaysia, 2019, pp. 1-5. doi:10.1109/ICSCC.2019.8843612.
[19] J. A. Nasir, O. S. Khan, I. Varlamis, Fake news detection: A hybrid CNN-RNN-based deep learning approach, International Journal of Information Management Data Insights 1 (2021) 100007. doi:10.1016/j.jjimei.2020.100007.
[20] S. Tipper, H. F. Atlam, H. S. Lallie, An investigation into the utilisation of CNN with LSTM for video deepfake detection, Applied Sciences 14(21) (2024) 9754. doi:10.3390/app14219754.
[21] W. S. Paka, R. Bansal, A. Kaushik, S. Sengupta, T. Chakraborty, Cross-SEAN: A cross-stitch semi-supervised neural attention model for COVID-19 fake news detection, Applied Soft Computing 107 (2021) 107393. doi:10.1016/j.asoc.2021.107393.
[22] A. D. Radhi, H. A. H. Al Naffakh, A.-I. Fuqdan, B. A. Hakim, B. Al-Attar, A comprehensive review of machine learning-based models for fake news detection, BIO Web of Conferences 97 (2024) 00123. doi:10.1051/bioconf/20249700123.
[23] R. K. Kaliyar, A. Goswami, P. Narang, FakeBERT: Fake news detection in social media with a BERT-based deep learning approach, Multimedia Tools and Applications 80(8) (2021) 11765–11788. doi:10.1007/s11042-020-10183-2.
[24] M. Q. Alnabhan, P. Branco, BERTGuard: Two-tiered multi-domain fake news detection with class imbalance mitigation, Big Data and Cognitive Computing 8(8) (2024) 93. doi:10.3390/bdcc8080093.
[25] P. Dhiman, A. Kaur, D. Gupta, S. Juneja, A. Nauman, G. Muhammad, GBERT: A hybrid deep learning model based on GPT-BERT for fake news detection. Heliyon 10(16) (2024) e35865. doi:10.1016/j.heliyon.2024.e35865.
[26] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach, arXiv (2019). URL: https://arxiv.org/abs/1907.11692.
[27] V. Sanh, L. Debut, J. Chaumond, T. Wolf, DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter, arXiv (2019). URL: https://arxiv.org/abs/1910.01108.
[28] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, ALBERT: A lite BERT for self-supervised learning of language representations, in: International Conference on Learning Representations, ICLR 2020, OpenReview, Addis Ababa, Ethiopia, 2020, pp. 1-17. URL: https://openreview.net/forum?id=H1eA7AEtvS.
[29] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, ELECTRA: Pre-training text encoders as discriminators rather than generators, in: International Conference on Learning Representations, ICLR 2020, OpenReview, Addis Ababa, Ethiopia, 2020, pp. 1-18. URL: https://openreview.net/forum?id=r1xMH1BtvB.
[30] FakeNewsNet Dataset, 2019. URL: https://github.com/KaiDMML/FakeNewsNet.
[31] P. K. Verma, P. Agrawal, R. Prodan. WELFake dataset for fake news detection in text data, 2021. doi:10.1109/TCSS.2021.3068519.