Transformer-Based Fake News Classification: Evaluation of DistilBERT With CNN-LSTM and GloVe Embedding

Yin Chen[1*], Bingqi Yin[2]
[1]Chongqing Institute of Engineering, Banan, Chongqing 400056, China
[2]School of Drama, Film and Media, Dalian Art College, Dalian 116699, Liaoning, China
E-mail: chenyin20182018@163.com
[*]Corresponding author

Keywords: fake and true news classification, deep learning, distilbert, glove embedding

Received: November 30, 2024

Social media networks have changed the face of communication in recent times, but at the same time, they have brought various challenges, such as disseminating fake news and information. NLP and machine-learning algorithms try to meet these challenges by structuring online information, although dataset bias remains a critical concern. SA has helped people gain insight into the context of news dissemination. Still, sham news dissemination-often by fake accounts represents a great hazard not only to users but to the stability of society. Several researchers have tried to assess the credibility of information and reduce sham data flow. In this work, the datasets of 17,903 fake news and 20,826 real news from Kaggle will be used. Preprocessing steps included text normalization, removal of punctuation, links, usernames, and non-alphabetic characters to prepare the data for analysis. This study explored categorizing fake and true news using advanced NLP techniques, transformer-based architectures, and deep learning models. A focus on improving classification accuracy and addressing dataset bias was achieved through models like DistilBERT, CNN, and LSTM. DistilBERT demonstrated remarkable performance, achieving an accuracy of 99.65%, with precision, recall, F1-score, and ROC-AUC values of 0.992188, 1, 0.996078, and 0.996894, respectively, outperforming the other models. The study's novelty lies in its detailed evaluation of DistilBERT, which showed significant improvements in accuracy, recall, and AUC while mitigating dataset bias. The results highlight the potential of DistilBERT for robust and reliable fake news classification, addressing critical limitations in existing approaches.

Povzetek:

1 Introduction

The advent of social media networks has revolutionized the exchange of knowledge and information, expanding reach like never before. Platforms such as Facebook, Twitter, and Instagram have facilitated communication on a global scale, allowing people to connect, share, and stay informed. However, this digital revolution has not come without its challenges. One significant consequence of the widespread use of social media is the alarming rise of false news and misinformation. The proliferation of deceptive content has led to dire consequences, impacting individuals' lives and sowing confusion among the masses [1]. As a result, there is an urgent need to address this issue and develop effective methods for identifying and combating false news. The vast volume of information published daily on online news portals and corporate websites poses a new challenge—automated organization and comprehension. To tackle this, researchers have turned to the field of NLP, specifically focusing on event extraction and classification. These approaches seek to extract and categorize crucial information about occurrences, but the unstructured nature of the initial data presents difficulties [2].

Detecting false news using ML algorithms has been a proposed solution, but most efforts have concentrated on specific types, such as political news, raising concerns about dataset bias [3]. Additionally, Sentiment Analysis, a computer-based approach that identifies emotional keywords from messages, plays a crucial role in understanding the context in which news is shared [4]. As the internet continues to expand rapidly, fake news has emerged as a pervasive societal concern, with propaganda and rumors being major culprits in misleading the public [5]. Social media, in particular, has become a primary avenue for disseminating false information through fake accounts and social bots, posing various risks to users [6]. The growth of electronic media brings both benefits and challenges. While technology grants unprecedented access to information, it also amplifies the spread of fake news, leading to conflicts and jeopardizing social stability. To combat this pressing problem, researchers are striving to evaluate the reliability of information and minimize the circulation of false data on these platforms [7].

In times of crisis, accurate detection becomes crucial for authorities to implement necessary mitigation measures. Despite the abundance of real-time data available on social media, certain studies lack efficient crisis embedding and categorization methods [8]. The pandemic illnesses such as COVID-19 have further underscored the importance of detecting false news, necessitating robust techniques and models to combat its dissemination [9].

Given the exponential growth of social media, misinformation, especially false news (FN), has become a global issue of concern. The complexity and multimodality of fake news make it difficult to identify, demanding the use of Computational Intelligence Approaches (CIA) to automatically detect and combat its spread [10].

1.1 Related works

Several studies on the topic of Fake and Real News classification using ML and DL algorithms are covered in the following. Wani et al. (2021) used Convolutional Neural Network (CNN), Long-Short Term Memory Network (LSTM), and Bidirectional Encoder Representations from Transformers (BERT) to test supervised text classification techniques on the COVID-19 fake news identification database. Using an unlabeled Covid tweets corpus, they also evaluated the importance of independent learning, language framework pre-training, and distributed word reconstructions [1]. Dogra et al. (2021) used ML and DistilBERT, a pre-trained DL structure, to create a hybrid text categorization technique. They refined a basic model for Indian Banking news events and employed a rule-based strategy to filter out false positives and false negatives. In terms of conveying general domain knowledge, DistilBERT surpassed other ML classifiers such as LR, SVC, DT, and RF [2]. Khan et al. (2021) compared sophisticated pre-trained language models for false news detection across 3 datasets in a benchmark study. The findings of their studies showed that BERT and comparable models fared well, particularly with tiny datasets, giving them preferable possibilities for languages with minimal electronic content [3]. Karande et al. (2021) created a methodology for detecting false news early in the publication process by assessing material using automated extraction of characteristics and text relevance. They included posture as a feature and employed pre-trained contextualized word embeddings BERT to get cutting-edge results. The model beat earlier work in the real-world information set, with an accuracy of 95.32% for false news identification [7]. Liu et al. (2021) presented CrisisBERT, an end-to-end transformer-based model for 2 crisis classification tasks, namely crisis detection and crisis identification, with promising accuracy results. They demonstrated that the suggested CrisisBERT model outperforms other benchmarks in terms of resilience [8]. In the collaborative work of CLEF-2021-CheckThat! Lab, Balouchzahi, Shashirekha, and Sidorov (2021) created models for identifying bogus news. They used training data to fine-tune 3 transformer-based language models from HuggingFace, Roberta, Distilbert, and BERT, then assembled them as estimators using majority voting. The algorithms are text classification problems with several classes [9]. Ogundokun et al. (2022) suggested a computational intelligence-based false news detection method that reduces vector-based feature dimensions employing a dimensionality reduction technique. They used 3 different CIAs: the GA, the K-Nearest Neighbor (KNN), and the BEL. Using confusion matrix metrics, the system's performance was analyzed, and it was discovered that GA with KNN surpassed GA with BEL in accuracy, sensitivity, and precision [10]. Fine-tuned multipurpose language representation models, for example, the BERT group framework (BERT, Distil-BERT), and the word embedding-based CNN and Fast-Text models, were employed by Saha et al. (2022) to successfully carry out the research. In this investigation, Distil-BERT outperformed BERT, Fast-Text, and CNNB accuracy [4]. Abdullah, Altiti, and Obiedat (2022) employed RoBERTa, a pre-trained language model, to identify complicated propaganda strategies in online news stories. The model was assessed utilizing a reference dataset for SemEval-2020 Task 11 [5]. Verma et al. (2022) present a User Credibility (UCred) model for distinguishing between false and legitimate user accounts. To categorize profiles, the model employs RoBERT, Bi-LSTM, and RF methods. The result is input into a voting classifier, which improves accuracy above state-of-the-art methods [6].

Kaliyar et al. (2021) addressed the issue of faking news in today's news era, with social networks allowing rapid and widespread dissemination of information, at times including untrue information. Previous work utilized sequential neural networks to model news content and social contextual information, but such work processed sequences of text in a unidirectional approach, limiting them to modeling semantic and long-range dependencies in a sequence. To address this problem, in this work, the authors designed FakeBERT, a deep neural model, combining Bidirectional Encoder Representations from Transformers (BERT) with parallel single-layer Convolutional Neural Networks (CNNs) with a variety of dimensions and filters of the kernel. FakeBERT was designed to enhance classification accuracy through use of bidirectional training to model semantic dependencies and mitigate uncertainty in natural language processing. Experimental performance revealed that FakeBERT performed much better than current models, with accuracy at 98.90%. In this work, the authors introduce a successful integration of models of BERT and CNN to address the issue of faking news, offering a strong model for enhancing natural language processing in such an issue [11]. Khanam et al. (2021) addressed the increasingly common issue of social and national destructive disinformation through spurious news in social networks and general platforms, citing its possible widespread social and national destructive impact. In a move to adequately counter such an issue, the authors conducted current studies in identifying spurious news and focused specifically on choosing the most effective conventional machine learning algorithms for real and spurious news differentiation. In a process that involved utilizing supervised machine learning algorithms, tools such as Python's scikit-learn package and natural language processing (NLP) for text processing, the developed model utilized tokenization and feature extraction with tools such as Count Vectorizer and Tfidf Vectorizer, culminating in feature selection, with experiments for feature selection and maximization of precision, measured in terms of output through a confusion matrix. The work shed new insights in terms of using conventional machine learning techniques and feature engineering in enhancing spurious news detection, with an emphasis placed in terms of using efficiency in curbing the widespread circulation of counterfeit information [12].

Bangyal et al. (2021) leveraged deep and machine algorithms for Twitter COVID-19 fake news sentiment analysis and achieved high accuracy using TF-IDF representation and algorithms including CNN and LSTM. In their work, recommendations for successful social network sentiment classification were proposed [13]. Alsuwat and Alsuwat (2025) addressed the growing issue of disinformation and misinformation, particularly via social media, with a proposed Framework for a Multi-Modal Fake News Detection (MM-FND). Having acknowledged the vulnerability of current methods, utilizing a single narrow feature set, the authors proposed combining several types of data for detection accuracytection. With three datasets, ISOT Fake News Dataset, LIAR Dataset, and COVID-19 Fake News Dataset, developed in their proposed framework, global feature extraction involved Word2Vec and Term Frequency-Inverse Document Frequency (TF-IDF), temporal feature through Bidirectional Long Short-Term Memory (Bi-LSTM) networks, and spatial feature through Named Entity Recognition (NER) and Global Vectors for Word Representation (GloVe). Classifier for classification involved Random Forest, leveraging the complementary strengths of these feature extraction methodologies. The proposed MM-FND framework performed better than conventional methods in terms of accuracy in detection. It achieved 96.3% (F1: 96.4%) for ISOT, 95.6% (F1: 94.2%) for LIAR, and 97.1% (F1: 97.9%) for the COVID-19 datasets. Findings displayed the effectiveness and robustness of the proposed framework in disinformation identification in a range of datasets, offering an effective tool in countering social and individual loss incurred through disinformation [14].

Altamimi (2024) proposed a model combining FastText, FastText-Subword, and GloVe embeddings with a custom-designed CNN for detecting fake news. With 94.58 accuracy and outclassing existing models, it showed prowess over datasets, including an independent Arabic Fake News dataset, offering a general-purpose model for countering disinformation [15]. Ellaky et al. (2024) designed a hybrid model combining GloVe embeddings with BiGRU and LSTM for social bot detection. With 100% precision and 99.73% accuracy for the Twibot-20 dataset, the model outclassed current methods and performed accurately in bot detection in new samples, illuminating deep learning's potential in bot detection [16].

Abdal et al. (2023) addressed the social threat of the mass proliferation of spurious information, citing an urgency for multilinguality detection frameworks in an aim to preserve an educated information environment. Abdal et al., in their work, proposed a transformer model for detecting Bangla fake information using a distilled model of the BERT model, namely, DistilBERT. Abdal et al. prepared a large corpus of samples of real and spurious samples of Bangla information for training and fine-tuning for model development. Through contextual and semantic awareness, Abdal et al. trained the model using a large corpus of Bangla information corpus. For evaluation, Abdal et al. compared a model using DistilBERT with traditional machine algorithms such as SVM and RF, and deep training algorithms such as LSTM. Experimental results confirmed that a proposed model outperformed alternatives with accuracy in detection of 97.85% in spurious information for Bangla language. This work attests to the effectiveness of transformer-based models in multilinguality in the detection of fabricated information and offers a powerful tool in curbing disinformation in language Bangla [17]. Kula et al. (2021) discussed model development and model evaluation for detecting fake news, particularly in relation to the most important social issue posed by disinformation in the new era of telecommunications. In developing the model, current state-of-the-art neural network architectures of the Transformers family were utilized, utilizing high-performance computation capabilities through Google Colaboratory platform and Flair library capabilities. Precision, recall, and an F1-score were utilized for model quality evaluation, providing a robust analysis of model performance. In its analysis, the work documented an assurance of strong artificial intelligence and deep learning approaches in the battle against fake news, with such tools proven capable of providing effective and reliable countermeasures against disinformation. With high-tech tools, developed models in this work showed a high-potential contribution towards countering the spread of fake news and enhancing social resistance to disinformation [18].

Chabukswar et al. (2024) debated about the social peril of disinformation, highlighting the urgency of discovering counterfeit news in all languages to have an educated and dependable information environment. The work suggested a transformer-based pre-trained model combining Distilled Bidirectional Encoder Representations from Transformers (DistilBERT) and Bidirectional Gated Recurrent Units (BiGRU) for identifying fake news in the English language. To that, a careful collection of political news articles, real and fabricated, was prepared. NLTK was utilized for the preprocessing of text, and a fine-tuning of the DistilBERT model for semantic relations and contextual information in the English language was performed. Model output underwent BiGRU layers, and sequential information and adjacent token dependences were extracted and captured. Experimental evaluation confirmed that the model performed effectively, and a high accuracy rating of 97.26% in discovering English fake news was attained. In the current work, it is displayed that integration of transformer-based architectures with recurrent layers can maximize accuracy in discovering fake news in English-language collections [19]. Zhi et al. (2021) tackled the biggest problem of financial fake news, one that can sway public opinion and manipulate financial markets. Confronted with the weakness of purely feature-based language models, authors proposed an overall scheme through integration of multi-source fact-checking and analysis with fake news detection. With a range of dimensions such as user comments, information sources, and financial market information, in terms of enhancing accuracy in detection, their multi-fact CNN-LSTM model operated. User comments were leveraged with an attention mechanism for useful information extraction, and a well-prepared white-list of high-quality web sources was adopted for information source checking. Information about financial products in the news, including financial markets, was checked, and statements were checked with real-time actual market price statements. Dynamical weighting during training allowed the model to assign weight to each dimension and dynamically learn them during training. Unlike purely model-based approaches, such an overall scheme effectively addressed complex financial fake news, and its successful establishment emphasized the importance of leveraging several sources of information for developing strong detection systems [20].

Several studies have explored fake and real news classification using machine learning (ML) and deep learning (DL) algorithms. Table 1 provides a summary of methodologies, datasets, reported accuracies, and limitations from prior works.

Table 1: Summary of previous studies on fake news classification

Study	**Methodology**	**Dataset**	**Accuracy**	**Limitations**
Wani et al. (2021)	CNN, LSTM, BERT	COVID-19 Fake News Dataset	~95%	Limited generalizability to non-COVID datasets; lacks contextual depth in text.
Dogra et al. (2021)	DistilBERT, Rule-Based Filtering	Indian Banking News Dataset	~96%	Dataset-specific, not scalable to broader topics of fake news.
Khan et al. (2021)	BERT Variants	Multiple Small Datasets	~94%	Limited to small datasets; computational inefficiency with large models.
Karande et al. (2021)	BERT, Feature Extraction	Social Media Posts	95.32%	High computational cost for real-time applications.
Liu et al. (2021)	CrisisBERT	Crisis News Dataset	~92%	Focused only on crisis-related news; lacks general applicability.
Ogundokun et al. (2022)	CI Techniques (KNN, BEL, GA)	Social Media Data	~89%	High dimensionality and reduced scalability.
Abdullah et al. (2022)	RoBERTa, Propaganda Classification	SemEval-2020 Task 11 Dataset	~93%	Limited to specific propaganda detection; not generalizable to other tasks.
Saha et al. (2022)	DistilBERT, FastText, CNN	Public Sentiment Dataset	~94%	Focused on sentiment analysis rather than fake news classification.
Kaliyar et al. (2021)	FakeBERT combining BERT with parallel single-layer CNNs to enhance semantic dependency modeling.	Custom Dataset	98.90%	Demonstrated improved performance through bidirectional training and semantic dependency modeling.
Khanam et al. (2021)	Supervised ML with tokenization, feature extraction, and confusion matrix-based precision evaluation.	Custom Dataset	Not specified	Highlighted efficient spurious news detection through conventional ML and feature engineering.
Bangyal et al. (2021)	TF-IDF representation with CNN and LSTM for Twitter COVID-19 fake news sentiment analysis.	Twitter COVID-19 Dataset	High accuracy	Proposed social network sentiment classification techniques.
Alsuwat & Alsuwat (2025)	Multi-Modal Fake News Detection (MM-FND) using Word2Vec, TF-IDF, Bi-LSTM, and NER.	ISOT, LIAR, COVID-19 Datasets	96.3%-97.1%	Robust multi-modal approach leveraging complementary feature extraction methodologies.
Altamimi (2024)	Custom CNN with FastText, FastText-Subword, and GloVe embeddings.	Arabic Fake News Dataset	94.58%	Showcased a general-purpose model for countering

				disinformation, particularly in Arabic datasets.
Ellaky et al. (2024)	Hybrid model combining GloVe with BiGRU and LSTM for social bot detection.	Twibot-20 Dataset	99.73%	Highlighted the potential of deep learning for accurate bot detection.
Abdal et al. (2023)	DistilBERT model for Bangla fake news detection with contextual and semantic awareness.	Bangla Fake News Corpus	97.85%	Demonstrated transformer models' effectiveness in multilingual fake news detection.
Kula et al. (2021)	Transformer architectures evaluated using Google Colab and Flair library for model quality.	Not specified	High accuracy	Showed strong potential for using advanced AI tools to counter disinformation effectively.
Chabukswar et al. (2024)	Transformer-based DistilBERT fine-tuned with BiGRU for fake news detection in English.	Political News Corpus	97.26%	Demonstrated the integration of transformer-based architectures with recurrent layers for enhanced accuracy.
Zhi et al. (2021)	Multi-source fact-checking and detection model with CNN-LSTM leveraging user comments, sources, and market data.	Financial News Dataset	Not specified	Highlighted the importance of leveraging multi-source information to tackle complex financial fake news.

The state-of-the-art models, BERT, RoBERTa, and their derivatives like DistilBERT, have shown promising accuracy in text classification tasks, such as fake news detection. Most of them, however, suffer from certain inadequacies: large transformer models, for example, like BERT, require huge computational power and, therefore, are not suited for real-time applications. Most of the previous studies conducted had focused on datasets that were either from a specific domain or highly unbalanced, such as COVID-19-related news; hence, biased and less applicable. Furthermore, most of the models are optimized to perform well on only one dataset or a particular topic, while they perform badly on others. Lastly, even if a model is accurate, transformer-based models would tend to behave like black boxes, hence making the decision process uninterpretable. To resolve those issues, this paper uses DistilBERT, the distilled version of BERT, since it has the goal of balancing its high performance with computational overhead. It is also applied to the Kaggle Fake and True News dataset to include more fake and true news items to increase the generalizability of the findings. It further systemically compares DistilBERT against CNN-LSTM models with better results in terms of accuracy, recall, and AUC score. Further, it has applied preprocessing techniques and balanced evaluation metrics to reduce the bias of the dataset. In light of this approach, one can see how DistilBERT achieves state-of-the-art performance by dealing with computational inefficiencies and biases, hence contributing to more robust and reliable fake news classification.

Given these challenges, it is clear that solving the issue of distinguishing between fake and true news necessitates a rigorous method of classification on time following the publication of news. Furthermore, earlier proposed approaches have limited accuracy or are incomplete. However, the research has failed to give the reasons why these models were chosen; nor has it supported the selection in the light of fake news classification. Here, a suitable explanation for employing these models-based on their appropriateness can be CNN because of its feature extraction ability, CNN-LSTM for its ability to process sequential data, and DistilBERT because of its efficiency in transformer models. This will explain their suitability for solving the problems of fake news detection. Another thing is that the study has not mentioned the reasons for not using other deep models, such as RoBERTa, XLNet, or even GPT-based architecture, which proved to have promising performances in similar tasks and could give so much insight into this research. Adding in a performance comparison of these options would have made the onus of the study design show that these selected models were the best for this use.

The lack of justification further opens up gaps in understanding the criteria behind the choice of models, leaving one speculating whether the results could have been further improved by other approaches. This shall enhance the transparency and robustness of the methodology in establishing that the study makes a valid contribution within the domain of misinformation detection.

This study's subsequent portion is arranged as follows: Section 2 focuses on technique in many subsections, including dataset analysis, methodology, and performance evaluation. The third component of this study contrasted base and classification approaches, while section 4 highlighted the study's key findings.

2 Methods

Fig. 1 shows the flowchart of the research methodology. The dataset was chosen from the Kaggle Fake and True data sets. The preprocessing and word embedding processes were then applied to input datasets for use in classification models. DL models and the Distilbert transformer model, its own parameters structure the models are discussed further below. Finally, the best model was chosen based on performance.

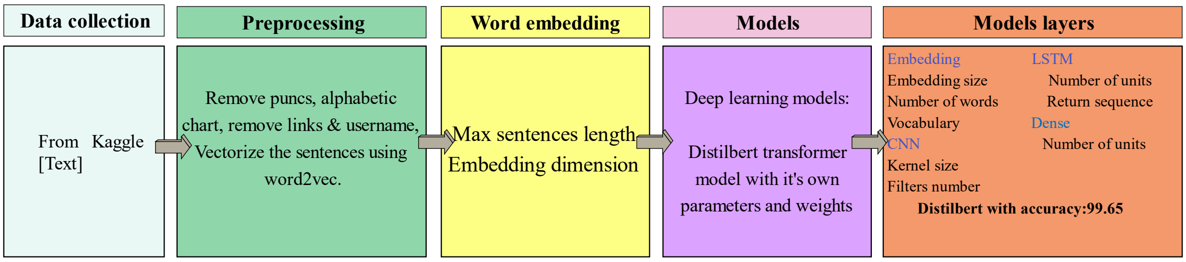

Figure 1: The flowchart diagram of this study

2.1 Research design

This study, therefore, adopted a research design to test the efficiency of different deep learning models for classification, but mainly compared performances between DistilBERT and CNN-LSTM models. The main objective of this research was testing the performance of these two models against their accuracy, recall, and overall performances in classification on a diverse dataset and also to check if DistilBERT eliminates those existing drawbacks that were revealed in other state-of-the-art models regarding inefficiency in computation and generalization issues. Hence, the hypothesis which was tested is that DistilBERT, relying on its transformer-based architecture and potentiality of fine-tuning, will outperform CNN-LSTM in terms of accuracy, recall, and robustness against biases of datasets. Therefore, the following are the research questions that this study attempted to answer:

1. How does DistilBERT's performance compare to the performance of CNN-LSTM models in fake news classification?
2. What are the most influencing factors that create the distinction between these ensembles?
3. Can DistilBERT alleviate both the computational inefficiency and poor generalizability of the traditional methods?

The proposed experiment design is targeted at modeling comparisons fairly and systematically. The reason for choosing the Kaggle Fake and True News dataset was the presence of a comprehensive and diverse collection of both fake and true news articles, which makes it suitable for testing generalizability across a variety of domains. Normalization of text, punctuation removal, and GloVe vectorization of text have been used consistently for all models to avoid any discrepancy. DistilBERT was chosen due to its reduced computational cost, along with its effectiveness in text classification tasks. CNN-LSTM models were chosen to represent a range of hybrid deep learning architectures that integrate convolution and sequence-based processing. Several CNN-LSTM variants, such as 2- and 3-layered models, have been utilized to present a detailed analysis of performance variation due to the shifting complexity in architecture.

These are the three intended outcomes of the study. Firstly, this research shall determine the best model performing the fake news classification based on Accuracy, Recall, and AUC. Secondly, this aims at proving the applicability of transformer-based models, such as DistilBERT, in real-world cases, embedding fake news detectors into social media. Finally, this research tries to bring to light the difficulties and limits of the existing methods, such as the dataset being biased or the computational inefficiency by giving ways for overcoming them. Thus, the research will hopefully develop further in constructing more robust, efficient, and generalizable models against misinformation in digital media.

2.2 Dataset

The "Fake" and "True" datasets from Kaggle were used in this work to identify fake news items. This data set contains 17,903 and 20826 distinct false and true news values, respectively.

2.2.1 Preprocessing steps

Preprocessing steps included:

1. Text normalization, such as converting text to lowercase.
2. Removal of punctuation, links, usernames, and non-alphabetic characters to clean the dataset.
3. Stratified sampling to address the slight class imbalance (53.8% true news vs. 46.2% fake news).

Also, the training of the presented deep learning model is conducted with a batch size of 32, and the Adam optimizer is employed to ensure impactful gradient optimization. The learning rate is set to 10^{-4} for the deep learning models, while a smaller learning rate of 10^{-5} is adopted for DistilBERT, as it is a larger model and needs fine-tuning to achieve optimal outputs. The input sentence length is fixed at 1500 tokens to maintain consistency in data processing. Training and fine-tuning are performed over 200 epochs to guarantee sufficient learning and convergence for the deep learning models and DistilBERT.

2.2.2 Impact of preprocessing

The preprocessing processes helped in a significant improvement in model performance and less bias in the dataset. Text processing such as converting to lowercase and removal of punctuation, links, username, and non-alphanumeric characters helped in excluding irrelevant factors interfering with the model's ability for meaningful patterns to learn. Stratified sampling helped in counterbalancing minor unbalanced datasets (53.8% for real and 46.2% for fake news) and offering balanced training representation for both datasets. All these processes together cleaned out noise in the dataset and helped in models such as CNN-LSTM and DistilBERT in focusing on intrinsic factors of a text. Embedding selection (static GloVe-300 for CNN-LSTM and dynamically generated DistilBERT embedding) added to these preprocessing processes, in addition to a further boost in capability of models for distinguishing between fake and real news in terms of contextual and semantic accuracy

This study employed GloVe-300-word embeddings for certain models to enhance text representation and enable comparisons with transformer-based embeddings used in DistilBERT. GloVe-300 embeddings, known for their ability to capture semantic relationships between words, were specifically used with CNN-LSTM models as part of this study's comparative analysis. The choice of GloVe-300 was motivated by its widespread application in deep learning for natural language processing tasks, particularly in scenarios requiring static word embeddings. The transformer model DistilBERT was compared with CNN-LSTM, 2CNN-LSTM, and 3CNN-LSTM.

Figs. 2 and 3 show word clouds from the Fake and True datasets separately. Thus, in these figures, more common subjects in news articles were shown in large font, while less repeated topics were shown in small font.

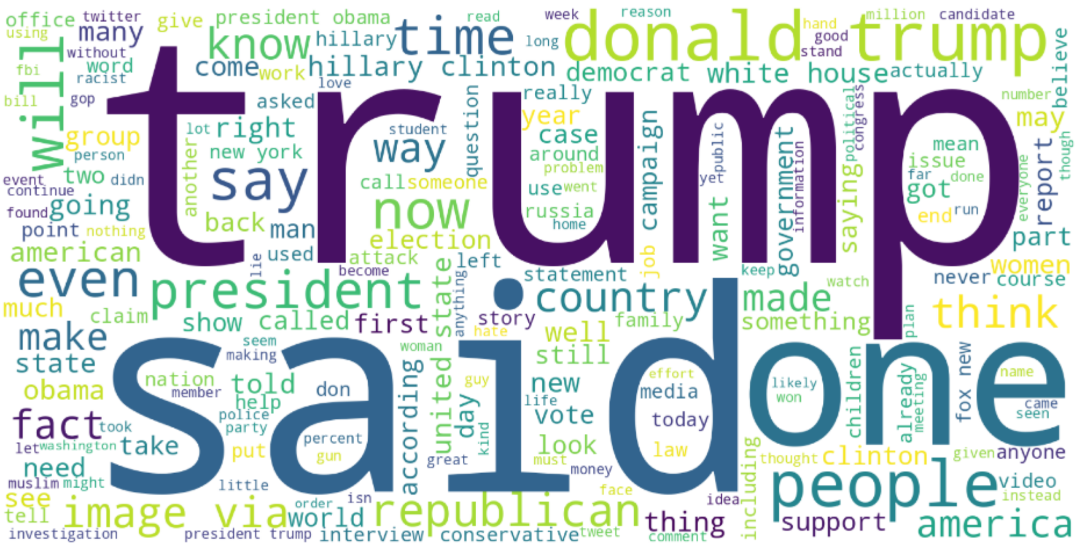

Figure 2: The word clouds of Fake class

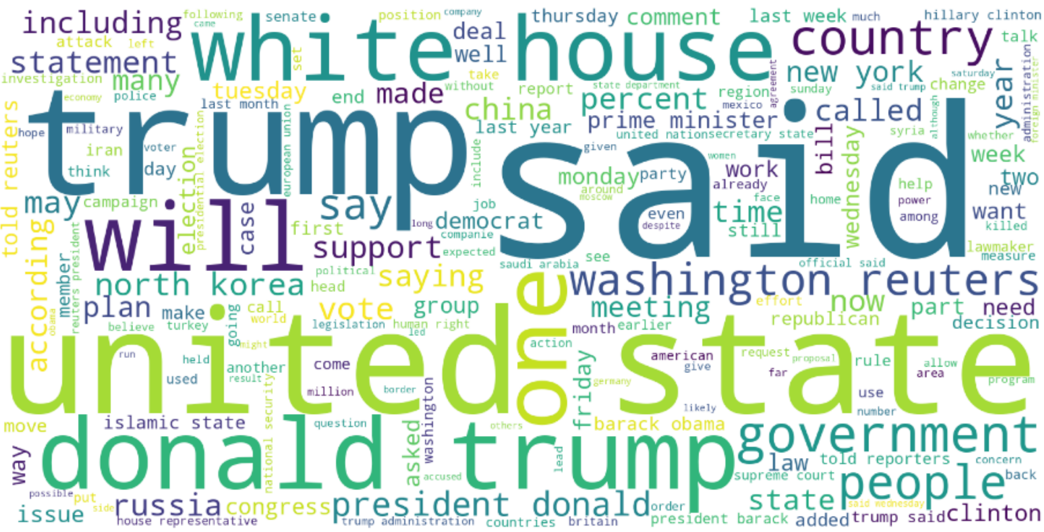

Figure 3: The word clouds of True or real class

Figure 4 illustrates sentence length analysis over the dataset for four possible thresholds: 1000, 1300, 1500, and 2000 words. Herein, it was determined that 1500, in sentence length one (yellow line), encodes 99.35% of the input dataset, including just about all sentences without adding too much padding or cutting off. This is chosen because the threshold brings a good balance between computational efficiency and useful content retention in the dataset, hence being appropriate for downstream text classification tasks.

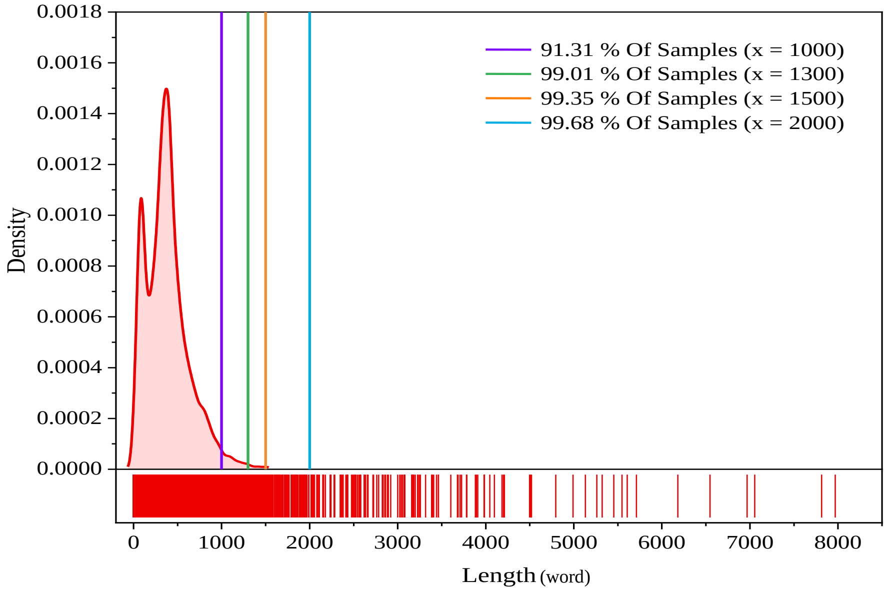

Figure 4: Histogram and line cover of input datasets

Fig. 5 displays the distribution of True and Fake news in the dataset, where 53.8% of the articles are classified as True news and 46.2% as Fake news. This slight imbalance, where True news is more prevalent, maybe a result of the dataset's composition on Kaggle, which could include more True news for validation purposes or to ensure a higher degree of authenticity in the data. While this imbalance is not extreme, it can still have implications for model performance. Specifically, the model might be biased toward predicting the majority class (True news), potentially leading to lower recall for Fake news. Such an imbalance can also inflate accuracy metrics, as models that predict the majority class more often will naturally perform better. To mitigate these effects, preprocessing techniques such as stratified sampling or class weighting were applied during model training. Moreover, evaluation metrics like F1-score and ROC-AUC, which are more sensitive to class imbalance, were chosen to ensure a fair assessment of the model's performance for both classes. This discussion emphasizes the importance of considering class distribution when evaluating fake news classification models.

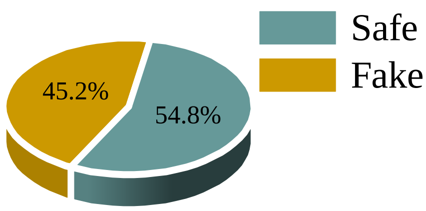

Figure 5: Pie chart of fake and true percentages

In this study, preprocessing entails lowering the text, eliminating punctuation, removing links and user names, and removing non-alphabetic letters. Texts were also converted to vectors using word2vec. The sentences were inflated to the length of the longest sentence, 64. Word2vec was utilized to vectorize the text phrases, enabling the use of word embeddings.

The GloVe-300 embeddings used in this study were only for the CNN-LSTM models. It is such embeddings that are provided to the CNN-LSTM architectures for initializing the word vectors with pre-trained word embeddings on large text corpora. Such embeddings were contrasted against an embedding matrix that has been trained on the dataset used in this study for fairness.

In contrast, the DistilBERT model used its own pre-trained transformer-based token embeddings, which capture the meaning of the word dynamically depending on the context. For DistilBERT, no static embeddings such as GloVe-300 were used, keeping in line with its transformer-based architecture. This is important for comparing the various performances of models using static embeddings, CNN-LSTM, and dynamic embeddings, DistilBERT.

To catch 99.35% of samples, consider a vocabulary size of 15000 words and a sentence length of 1500.

2.3 Convolutional neural network

Unlike other approaches, the Convolutional Neural Network (CNN) [21] comprises a feedforward neural network that identifies characteristics from the input using convolution architectures. Its design was driven by the sense of sight, with neurons from nature equating to synthetic versions and CNN kernels mimicking receptors. Activation functions replicate neural communication of signals, whereas loss functions and optimizers instruct the CNN system to acquire anticipated [22]. CNN structures are divided into 3 categories: convolutional, pooling, and fully connected layers. Pooling layers calculate distinct mappings of features, while convolutional layers learn representations of characteristics from data inputs. Each neuron in the preceding layer is linked to an area of nearby neurons. Convolving the input with a learned kernel and using an element-wise nonlinear function of activation yields a novel feature list. Several kernels are used to generate comprehensive maps of characteristics [23]. The basic elements of CNN are provided here to aid in better understanding.

2.3.1 Layers of convolutions

A convolution layer, which combines linear and nonlinear processes such as convolution and activation functions, is an important CNN layer for finding features. Convolution is a linear discovery of features procedure that applies a kernel to a tensor. At every point, the element-wise product of the kernel and the input tensor is computed, yielding the characteristic mapping. This method is repeated to generate numerous maps of characteristics reflecting distinct input tensor properties, resulting in diverse characteristic extractors [24]. A convolutional layer is a series of parallel characteristics produced by sliding various kernels. It conducts product and summation between the kernel and the input picture, translating data into feature maps. The kernel is smaller than the source picture [25].

2.3.2 Layers of pooling

Pooling is a key principle in CNNs since it reduces computational load by minimizing links between layers of convolution. Recent pooling approaches, such as max-pooling, minimize image processing complications and resolution. It divides photos into rectangular sub-regions and returns the greatest value inside each sub-region [23,26].

2.3.3 Activation function

Non-linearity is the subsequent layer following convolution, and it modifies or restricts the result of the algorithm. The activation function is utilized to activate neuronal properties while also preserving and mapping features. It specifies the output of a specific neuron following a sequence of inputs, maintaining characteristics and reducing data redundancy. The activation function is a function between 2 layers in multilayer neural networks. CNNs, like the neuron model in the human brain, may represent complex properties by using multiple activation functions. Each neuron receives the previous layer's output value as input and transfers the processed data to its subsequent layer [22,26,27]. Several common activation functions, in the following, are described.

- Rectified linear unit (RELU) functions as a well-known non-saturated function for activation. Mathematically, the RELU is stated as follows:

$$ReLU(x) = \max(0, x)$$
$$\frac{d}{dx} ReLU(x) = \{1 \; if \; x > 0; 0 \; otherwise\} \quad (1)$$

$ReLU$ acts as a piecewise linear function that eliminates negative sections while keeping positive ones. It produces sparsity in hidden units and is quicker than other activation functions. Despite the discontinuity at 0 that might impact backpropagation performance, experimentally, ReLU surpasses other activation functions.

- The SoftMax function is a scalar-valued function that has an exponential assessed at each vector component and is normalized by the sum of every one of the scalar components [28]. Regardless of the use, its look differs. The SoftMax function $SM: \mathbb{R}^k \rightarrow \mathbb{R}^k$ is stated below:

$$SM(Z)_j = \frac{e^{Z_j}}{\sum_{i=1}^{k} e^{Z_i}} \quad (2)$$

Every component Z_j in the input vector Z, is subjected to the exponential function, and the outcomes are normalized by division by the total sum of the exponentials. This guarantees that the resulting vector $SM(Z)$ sums to one [29].

2.3.4 Layer of fully connected dense

The dense layer is fully linked and serves as a classifier. Whereas convolutional layers, pooling layers, and activation function layers map the initial data to hidden characteristics, the fully connected layer maps learned characteristics into the sample markup area [25].

2.3.5 Loss function

A loss function, formerly referred to as a cost function, assesses the similarity of network forecasts of results and reality labels. The cross-entropy approach is often employed for the classification of multiple classes. Loss functions are hyperparameters that have been developed for particular duties and are used as training criteria in optimization and regression situations.

2.3.6 Embedding

The GloVe (Global Vectors for Word Representation) model was used to generate word embeddings for the CNN-LSTM models in this study. GloVe learns word representations by analyzing the co-occurrence statistics of words within a large corpus. Specifically, it reduces the dimensions of the co-occurrence matrix to create dense vector representations of words, where similar words have similar vector representations. This process captures both semantic relationships and contextual similarities, enabling the model to represent textual data effectively [30]. Glove learns word embeddings by lowering the number of dimensions of the co-occurrence number matrix and utilizing proportions of co-occurrence probability to discriminate important from extraneous words. By training on nonzero entries in a word-word cooccurrence matrix, this model successfully employs data from statistics, yielding an important vector area [31].

For this study, GloVe-300 embeddings, pre-trained on a large corpus, were chosen to initialize the embedding layer in CNN-LSTM models. These embeddings provide a static representation of words, which is well-suited for the sequential processing capabilities of CNN-LSTM architectures. By using GloVe, the models can leverage pre-existing knowledge about word relationships, improving their ability to classify fake and true news articles.

However, static embeddings like GloVe have limitations, such as the inability to account for word meanings in different contexts (e.g., "bank" as a financial institution vs. "bank" as a riverbank). This contrasts with the transformer-based embeddings in DistilBERT, which dynamically adjust word meanings based on context. Despite this, GloVe was selected for CNN-LSTM models due to its efficiency and compatibility with the architecture, providing a strong baseline for comparison with transformer-based models

2.4 Long short-term memory

The long short-term memory (LSTM) approach, initially introduced by Hochreiter and Schmid Huber in 1997 [32], is a popular deep-learning technique for digesting and forecasting significant occurrences in time series with large intervals. Because of its lengthy data processing capabilities, it is especially effective for handling information with long-term dependencies, where the outcome is dependent on earlier time steps [33–35]. There are 3 gates in an LSTM memory cell: forget, input, and output. During training, each gate gets assigned a distinct duty. The cell state stores the alteration in the state, ensuring that earlier data is preserved. The forget gate determines persistence by testing the relevance of data from the prior state via a sigmoid function. The input gate provides new data, while the output gate determines forecasts from the current time step's modified state and input [36]. An LSTM network consists of the following components:

The input gate determines how much network information is required for internal storage as well as information flow to memory cells.

$$I_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i) \quad (3)$$

Here σ refers to the logistic function, h_{t-1}, and x_t demonstrate the memory cell target and the input vector, respectively; W_{ih}, W_{ix} denotes the weight matrix of the input gate, and b_i indicates the term that defines the bias of the input gate.

The forget gate regulates data flow and removes previous data in memory cells.

$$F_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f) \quad (4)$$

Memory cell output to the network is controlled by the output gate, which determines the internal state data required for external state output.

$$O_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o) \quad (5)$$

In memory cells, the cell state maintains the information.

$$C_t = f_t C_{t-1} + \tanh\,(W_{ch}h_{t-1} + W_{cx}x_t + b_c) \quad (6)$$

In the hidden state, the LSTM unit outputs forecasts or data to the following unit.

$$S_t = O_t + \tanh\,(C_t) \quad (7)$$

2.5 DistilBERT

DistilBERT is a simplified variant of BERT that was introduced by [37] that employs the distillation of information, does away with token-type embeddings and the pooler, and reduces layers by an amount of 2 [38]. DistilBERT is a knowledge distillation-based ML algorithm assessment method that includes moving data from the primary model, the teacher, to a lesser, distilled version, the student. This compression technique explains how a model generalizes and transfers knowledge from instructor to student. DistilBERT uses the same procedures as the original BERT to design a student version and teach it on a bespoke dataset for a given purpose, involving pre-training and fine-tuning [39]. The DistilBERT student structure is similar to BERT, however, it eliminates token-type embeddings and the pooler, as well as reduces the number of layers through an amount of 2. In recent algebraic frameworks, the majority of operations in Transformer construction, including linear layer and layer normalization, are significantly optimized. Variations in the hidden size dimension have a lesser influence on computing efficiency for constant variable budgets than variables such as the number of layers, according to research. As a result, the emphasis is on minimizing the number of layers [37]. Student initialization plays an important role in training processes that make use of the shared dimensionality of teacher and student systems. In removing just one of the 2 layers, the sub-network converges, improving the training method. Distil BERT has been trained on huge batches using excellent approaches, such as gradient accumulation and dynamic masking, despite the next sentence forecasting aim.

2.6 Architectures of models

The article's performance of the models was assessed in this section. In the whole model, training was done on GPU. The architects of models are expressed as follows.

In the Distilbert transformer model that was used pre-trained model. The Distilbert model was employed as the suggested model's backbone and as the final layers. And one Dense layer with 64 units activated by RELU. In addition, there is one thick layer with one unit and an activation sigmoid for classification.

3-LSTM models involve embedding layer with 300 dimensions, LSTM with 64 units with return_sequens=True (in a bidirectional way), LSTM with 128 units with return Sequens=True (in a bidirectional way), LSTM with 256 units (in a bidirectional way), Dense layer with 64 units with RELU activation, and Dense layer with 1 unit for classification with sigmoid activation.

In the CNN-LSTM model with Embedding layer with 300 dimensions, a one-dimensional CNN layer (CNN1d) with kernel size 32 filters and 2 for kernel size with RELU activation, LSTM with 32 units (in a bidirectional way), Dense layer with 64 units with RELU activation, and Dense layer with 1 unit for classification with sigmoid activation was trained.

While included in the 2CNN-LSTM model are the Embedding layer with 300 dimensions, CNN1d with kernel size 32 filters and 2 for kernel size with RELU activation, CNN1d with kernel size 64 filters and 2 for kernel size with RELU activation, LSTM with 32 units with return_sequens=True (in a bidirectional way), LSTM with 64 units (in a bidirectional way), Dense layer with 64 units with RELU activation, and Dense layer with 1 unit for classification with sigmoid activation.

3CNN-LSTM model architects use Embedding layer with 300 dimensions, CNN1d with kernel size 32 filters and 2 for kernel size with RELU activation, CNN1d with kernel size 64 filters and 2 for kernel size with RELU activation, CNN1d with kernel size 128 filters and 2 for kernel size with RELU activation, LSTM with 32 units with return_sequens=True (in a bidirectional way), LSTM with 64 units with return Sequens=True (in bidirectional way), LSTM with 128 units (in a bidirectional way), Dense layer with 64 units with RELU activation, and Dense layer with 2 units for classification with sigmoid activation.

2.7 Evaluation of the performance

A confusion matrix is a table that is used to characterize and evaluate the efficiency of a classification method, effectively analyzing classification challenges. It visualizes and summarizes the performance of the method, using row and column numbers corresponding to class numbers. The confusion matrix is made up of 4 main properties which are utilized to create the classifier's measuring criteria [40]. These 4 figures are:

- TP stands true positive. The method's anticipated worth matches false news, indicating accurate categorization and false news propagating through social networks.
- FN represents a false negative. Its output happens when legitimate news is wrongly labeled as negative.
- TN (True negative) indicates that the system successfully identified a news item whenever its anticipated value fits what is happening.
- FP (False positive): News is mistakenly categorized as legitimate news even when it's fake news [41].

The confusion matrix was utilized for assessing the performance of data classification techniques utilizing criteria such as accuracy, error, precision, recall, F1-score, and AUC, which is expressed as follows:

Accuracy

The accuracy criteria are used to assess the model's correct diagnosis. The proportion of correct diagnoses to total data is used to calculate accuracy for the balanced dataset.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

Precision

The precision factors signify the ratio of the number of specimens with a correct positive diagnosis to the overall number of positive diagnoses.

$$Pre = \frac{TP}{TP + FP} \quad (9)$$

Recall

The recall metric is given as the percentage of accurate positive results out of the overall number of specimens that have to be positive.

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

F1 score

The F1 score represents the balance of accuracy and recall.

$$F1 = \frac{2 \cdot Pre \cdot Recall}{Pre + Recall} \quad (11)$$

AUC

AUC measures the accuracy of model predictions independent of the categorization criterion. AUC may be critical in lowering classification errors in circumstances with dramatically differing false negatives and false positives costs, such as emphasizing reduced false positives despite a considerable rise in false negatives.

2.8 Reason for model choice

The research utilized DistilBERT for its performance and computational efficiency and leveraged its use of a distilled model in minimizing computational requirements. CNN-LSTM models were utilized as examples of hybrid architectures with capabilities in feature extraction (with CNN) and sequence modeling (with LSTM). Nevertheless, lighter transformer alternatives such as RoBERTa or XLNet were not included, a limitation for future studies to capitalize on.

3 Results and discussion

In this section of the paper, the performance of the Distilbert models and 3-LSTM, CNN-LSTM, 2CNN-LSTM, and 3CNN-LSTM methods are compared to each other to classify Fake and True news. The compared criteria involved accuracy, precision, recall, F1-score, and AUC of the prediction models. Then, the best prediction model is chosen according to the performance of the criteria.

The research utilized DistilBERT for its performance and computational efficiency and leveraged its use of a distilled model in minimizing computational requirements. CNN-LSTM models were utilized as examples of hybrid architectures with capabilities in feature extraction (with CNN) and sequence modeling (with LSTM). Nevertheless, lighter transformer alternatives such as RoBERTa or XLNet were not included, a limitation for future studies to capitalize on.

Fig. 6 depicts the accuracy, precision, recall, F1-score, and AUC criteria of models, with further information in Table 2. 3CNN-LSTM is the weakest model based on classification model criterion values. The performance of the 2CNN-LSTM, CNN-LSTM, and 3-LSTM ranges from poor to excellent. The Distilbert transformer model is the most resilient and best model based on all criteria. Although all of the models utilized in this study are appropriate for classifying Fake and True news, the Distilbert model has the potential for substantial accuracy and performance.

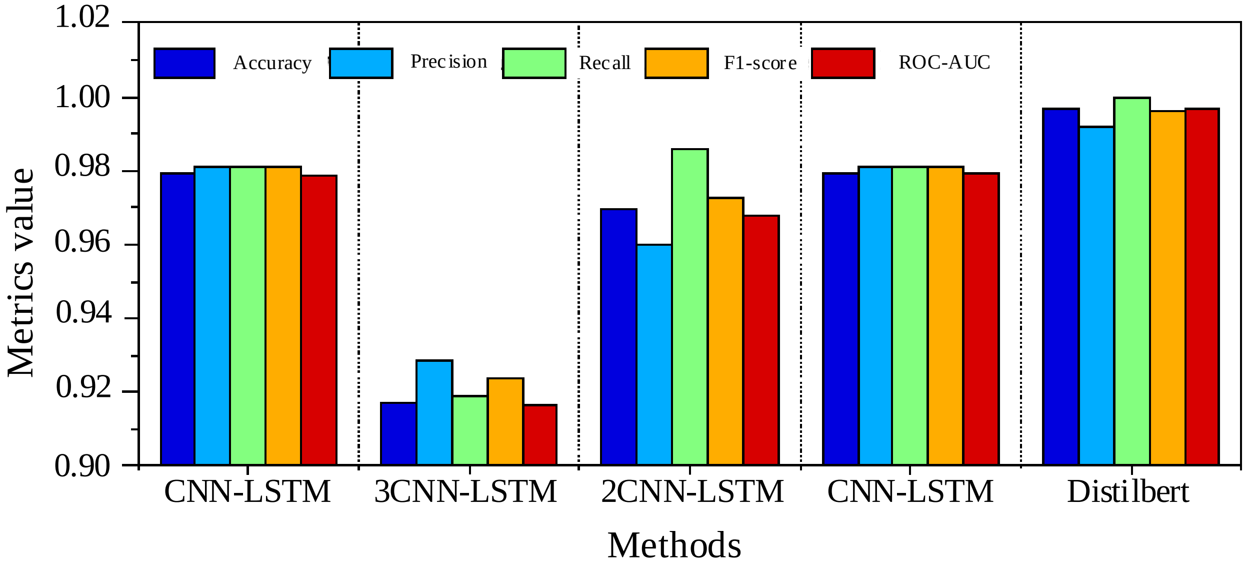

Figure 6: Metrics value for Distilbert, 3-LSTM, CNN-LSTM, 2CNN-LSTM, and 3CNN-LSTM methods

Table 2: Criteria values of Distilbert, 3-LSTM, CNN-LSTM, 2CNN-LSTM, and 3CNN-LSTM methods

Models	Accuracy	Precision	Recall	F1	ROC-AUC
3-LSTM	0.980207	0.971984	0.992435	0.982103	0.978932
3CNN-LSTM	0.917076	0.928588	0.919149	0.923845	0.91686
2CNN-LSTM	0.969599	0.959724	0.985816	0.972595	0.967908
CNN-LSTM	0.979301	0.981315	0.980851	0.981083	0.97914
Distilbert	0.996528	0.992188	1	0.996078	0.996894
Naive Bayes	0.830207	0.811984	0.842435	0.8269	0.838932

The comparative analysis revealed that DistilBERT achieved the highest metrics across all evaluation criteria. Specifically, DistilBERT outperformed the CNN-LSTM-based models (3-LSTM, CNN-LSTM, 2CNN-LSTM, and 3CNN-LSTM), with an accuracy of 99.65%, precision of 0.992188, recall of 1, F1-score of 0.996078, and ROC-AUC of 0.996894. In contrast, the 3CNN-LSTM model showed the weakest performance with an accuracy of 91.71%, followed by 2CNN-LSTM at 96.96%. Such good performance from DistilBERT has been attributed to the transformer-based architecture of this model. Indeed, it is very good at catching long-range dependencies and subtle contextual information. On the contrary, CNN-LSTM-based architectures sequentially process information and use static embeddings such as GloVe; hence, failing to adapt to the complications that may be present within a dataset.

The results demonstrated that the DistilBERT model outperformed all CNN-LSTM variants across all metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. Importantly, while CNN-LSTM models relied on GloVe-300 embeddings, the DistilBERT model leveraged its pre-trained transformer embeddings, highlighting its ability to dynamically adapt to contextual nuances in the data. This distinction emphasizes the superiority of DistilBERT's dynamic embeddings over static embeddings like GloVe-300 for the task of fake news classification.

The results of this study align with prior research indicating the superiority of transformer-based models for text classification tasks. State-of-the-art models such as BERT and RoBERTa have demonstrated high accuracy in text classification tasks by leveraging contextualized word embeddings and self-attention mechanisms. DistilBERT, a distilled version of BERT, inherits these advantages while reducing computational overhead, making it particularly suitable for real-world applications.

For example:

1. Wani et al. (2021) reported approximately 95% accuracy using BERT for COVID-19 fake news classification.
2. Karande et al. (2021) achieved 95.32% accuracy using a BERT-based approach for social media stance detection.

The proposed DistilBERT model outperformed these results, achieving a higher accuracy and recall. This improvement can be attributed to the extensive preprocessing techniques employed in this study, which reduced noise in the dataset, and the inherent efficiency of DistilBERT in capturing long-range dependencies and subtle contextual nuances.

The application possibilities of DistilBERT lie in incorporating the same into an automated fake news detection system that can be deployed on social networking sites such as Facebook. It should be able to yield exact results across all types of heterogeneous data in bulk. However, computational cost, although reduced from BERT, can again become a problem for very resource-constrained applications or applications that need to be performed in real-time. Probable lightweight transformer models or hybrid architectures are potential future research directions that try to strike a proper balance between accuracy and efficiency.

Although the dataset used from Kaggle is somewhat balanced, every possibility of biases arising because of data imbalance can be accounted for. Different preprocessing techniques, such as uniform sampling and word embedding for vectorization, have been performed in the presented work to nullify these effects. However, residual biases may remain because they are inherently designed to represent variation in different styles of writing of fake and true news. Advanced techniques, such as data augmentation or adversarial training, can be used in future works to make the model more robust against these biases.

The performance of DistilBERT is relatively better than just CNN-LSTM but shows competitive results compared to larger transformer-based models like BERT or RoBERTa; however, DistilBERT reaches comparable accuracy while keeping computational efficiency, hence being attractive for practical applications. However, the dependency on quality labeled data and heavy computational resources points at least to two fields that deserve further optimization and innovation for them to be more accessible and applicable under diverse real-world conditions.

While DistilBERT is computationally lighter than BERT, still resource-consuming, particularly for systems with resource limitations in real-time applications. This reliance upon pre-trained embeddings, both from GloVe and DistilBERT, restricts adaptability to domain-specific nuances not represented within the pre-training corpus. Similarly, static embeddings, such as those provided by GloVe in CNN-LSTM models, don't capture variability and are, therefore, less effective than transformer-based architectures. Second, this work has not compared DistilBERT against other state-of-the-art transformers such as RoBERTa and XLNet. Finally, other interesting avenues for future research include the following: the combination of DistilBERT with other recent variants of transformers, advanced preprocessing by data augmentation or adversarial training, and the development of hybrid architectures that leverage the best from CNN-LSTM and transformers. Overcoming these limitations might result in even more robust, efficient, and generalizable models for fake news detection in practical scenarios.

A separate discussion can also be made about the confusion matrix. The confusion matrix for the DistilBERT model demonstrates outstanding classification performance in distinguishing fake and real news. As shown in Figure 7, the model correctly classified 98% of fake news as fake, with a minor 1.8% false positive rate, where some fake news was misclassified as real. Notably, the model achieved perfect recall (100%) for real news, meaning it did not misclassify any real news as fake. The high precision (98.18%) and recall (98%) for fake news, along with a near-perfect F1-score (98.09%), confirm the model’s robustness. These results indicate that DistilBERT effectively captures contextual patterns in text, minimizing errors, though a slight tendency remains to misclassify a small fraction of fake news as real.

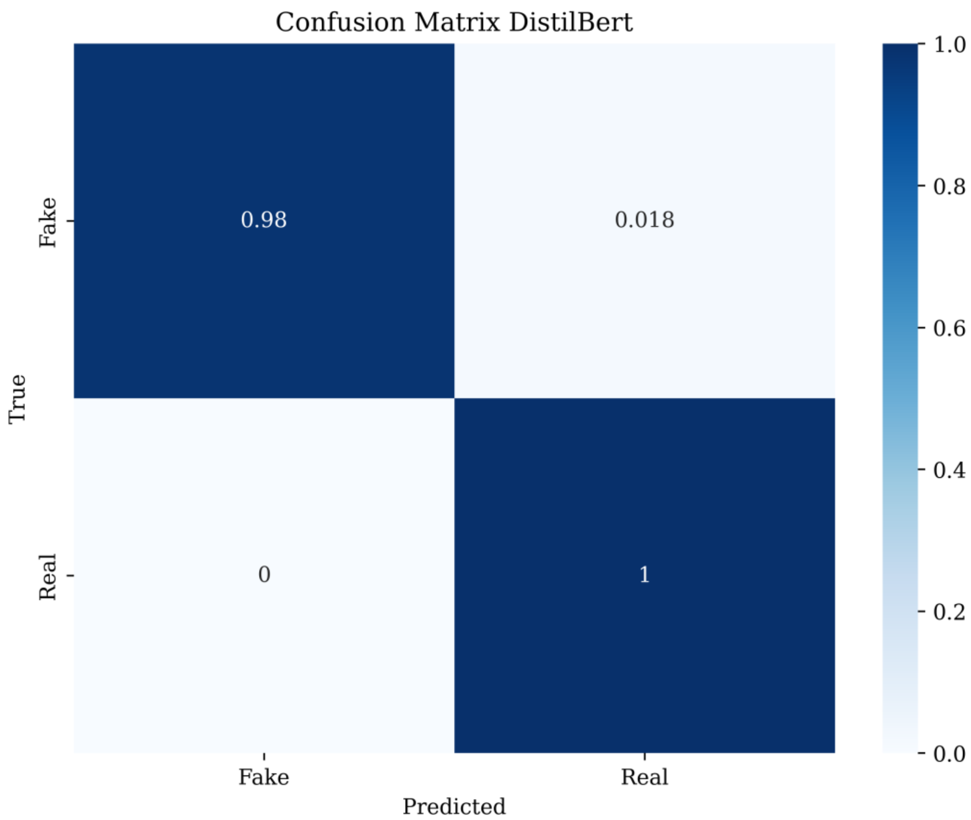

Figure7: Confusion matrix for the DistilBERT model.

Computational feasibility is crucial for real-world applications because the amount of time required for model training and inference time are of dual importance in practical applications. These values are examined in Table 3.

Table 3: Training and inference times for all models.

Model	Accuracy	Training_ Time_relative	Inference_ Time_relative
3-LSTM	0.9802	1.429	1
3CNN-LSTM	0.9171	1.786	1.312
2CNN-LSTM	0.9696	1.571	1.188
CNN-LSTM	0.9793	1	1.062
DistilBERT	0.9965	4.286	3.312

Figure 8 how well DistilBERT classifies fake and real news, the ROC curve indicates almost perfect classification performance: AUC equal to 0.999709 against fake news and 0.999711 against real news. Indicating that the model scored extremely high true positive rates while keeping very low false positive rates, which proves its extreme reliability for misinformation detection. The curve is practically at the top-left corner, meaning it falsely classifies very few instances of fake news as real and vice versa. Such performance indicates that DistilBERT is a robust tool for fake news detection. Nevertheless, despite the very high AUC values, further working on false positives and false negatives is critical due to the potential societal consequences of these classification errors. A single false positive could amplify a significant misinformation story, while an equal number of false negatives could discredit a legitimate story, so it is important to scrutinize patterns of errors and model bias for deployment in the real world.

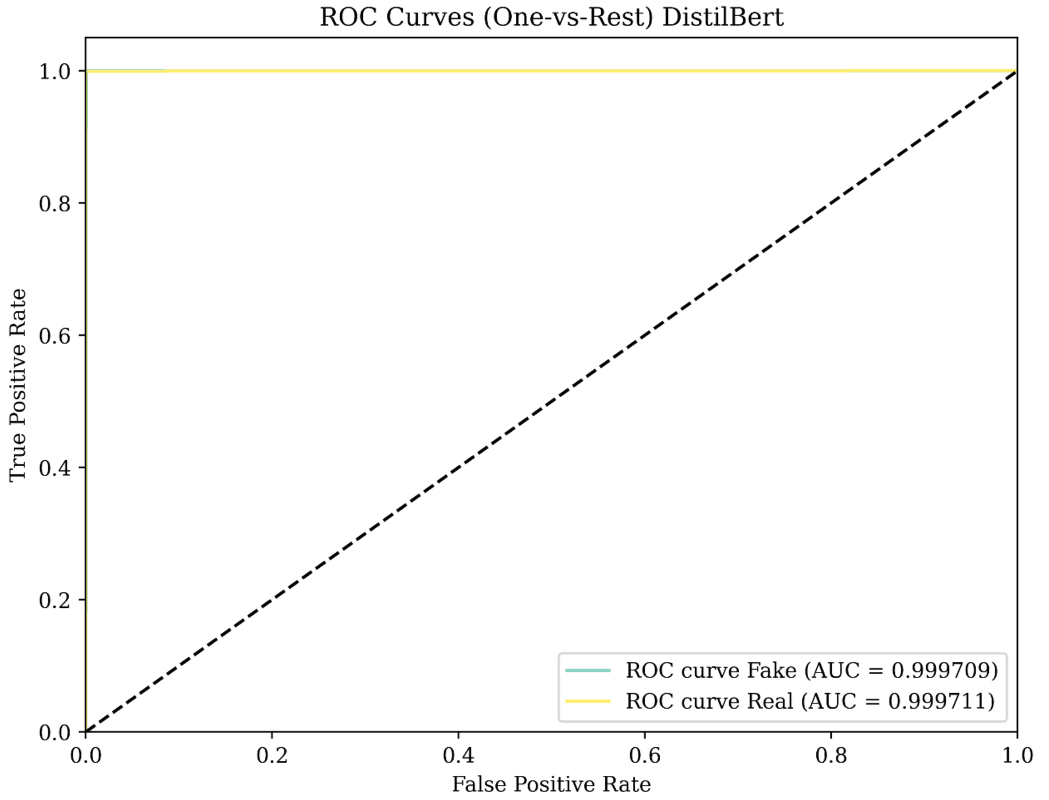

Figure 8: Classifies fake and real news by DistilBERT model.

4 Conclusions

The categorization of Fake and True news was assessed in this study. Because time categorization and accuracy in important occurrences are required. As a result, having trustworthy categorization algorithms is critical. As a consequence, evaluate 5 DL approaches often employed in this industry to select the one that finally performs the best. 5 classification approaches include the Distilbert, 3-layer-long short-term memory, one, 2, and 3-layer-convolutional neural network-long short-term memory. In addition, multiple techniques were evaluated using the performance measures accuracy, precision, recall, F1-score, and ROC-AUC. The following are the important findings of this study:

The Distilbert model has greater accuracy, precision, recall, F1-score, and ROC-AUC metrics than other models. Furthermore, the Distilbert model has a recall value of one.

3-Layer-long short-term memory outperforms convolutional neural network-long short-term memory models.

Although the performance of the 3-layer CNN-LSTM model was reasonable, the efficiency and accuracy of DistilBERT were far more superior to it. Increasing the depth of architecture can improve feature extraction, but going beyond a few layers results in diminished returns, extra computational complexity, and potential for overfitting. The trend in performance suggested that although the 3-layer CNN-LSTM model was competitive, a 2-layer model had a better bargain on accuracy versus efficiency. Hence, yes, deeper CNN-LSTM architectures can improve classification to some extent. The best trade-off between complexity and performance, however, lay with the shallower models, as DistilBERT outperforms all CNN-LSTM variants.

The classification model became weaker as the number of layers in convolutional neural network-long short-term memory models increased. As a result, the 3-layer convolutional neural network-long short-term memory model performs worse than the 2-layer and one-layer convolutional neural network-long short-term memory models.

Nomenclature

Nomenclature		Greek letters	
Abbreviations		Latin Symbols	
NLP	Natural Language Processing	σ	Logistic Function
FN	False News	h_{t-1}	Memory Cell Target
CIA	Computational Intelligence Approaches	x_t	Input Vector
CNN	Convolutional Neural Network	h_{t-1}	Previous Hidden Cell State

LSTM	Long-Short-Term Memory Network	W_{ih}, W_{ix}	Weight Matrix of the Input Gate
BERT	Bidirectional Encoder Representations from Transformers	b_i	Bias of the Input Gate
LR	Logistic Regression	W_{fh}, W_{fx}	Input and Recurrent Weight Matrices
SVC	Linear Support Vector Classification	I_t	Input Gate
DT	Decision Tree	F_t	Forget Gate
RF	Random Forest	O_t	Memory Cell Output
GA	Genetic Algorithm	C_t	Memory Cells
KNN	K-Nearest Neighbor	S_t	Hidden State
		TP	True Positive
BEL	Bagged Ensembled Learning	FN	False Negative
UCRED	User Credibility	TN	True Negative
RELU	Rectified linear unit	FP	False Positive
		Acc	Accuracy
		Pre	Precision
		$f1$	F1-score

Acknowledgements

I would like to take this opportunity to acknowledge that there are no individuals or organizations that require acknowledgment for their contributions to this work.

Competing of interests

The authors declare no competing of interests.

Authorship contribution statement

YC performed Data collection, simulation and analysis. BY evaluate the first draft of the manuscript, editing and writing.

Data availability

Data can be shared upon request.

Declarations

Not applicable

Conflicts of interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Author statement

The manuscript has been read and approved by all the authors, the requirements for authorship, as stated earlier in this document, have been met, and each author believes that the manuscript represents honest work.

Funding

This work was supported by Scientific and Technological Research Program of Chongqing Municipal Education Commission (Grant No. KJQN202401905)

Ethical approval

The research paper has received ethical approval from the institutional review board, ensuring the protection of participants' rights and compliance with the relevant ethical guidelines.

References

[1] Wani, A., I. Joshi, S. Khandve, V. Wagh and R. Joshi (2021). Evaluating deep learning approaches for covid19 fake news detection, In *Combating Online Hostile Posts in Regional Languages during Emergency Situation: First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers 1*, Springer, Cham, pp. 153–163. https://doi.org/10.1007/978-3-030-73696-5_15.

[2] Dogra, V., A. Singh, S. Verma, Kavita, N.Z. Jhanjhi and M.N. Talib (2021). Analyzing DistilBERT for sentiment classification of banking financial news, In *Intelligent Computing and Innovation on Data Science: Proceedings of ICTIDS 2021*, Springer, Singapore, pp. 501–510. https://doi.org/10.1007/978-981-16-3153-5_53.

[3] Khan, J.Y., M.T.I. Khondaker, S. Afroz, G. Uddin and A. Iqbal (2021). A benchmark study of machine learning models for online fake news detection. *Machine Learning with Applications*, Elsevier, 4, p. 100032. https://doi.org/10.1016/j.mlwa.2021.100032.

[4] Saha, U., M.S. Mahmud, M. Keya, E.A.E. Lucky, S.A. Khushbu, S.R.H. Noori and M.M. Syed (2022). Exploring public attitude towards children by leveraging emoji to track out sentiment using

distil-BERT a fine-tuned model, In *International Conference on Image Processing and Capsule Networks*, Springer, Cham, pp. 332–346. https://doi.org/10.1007/978-3-031-12413-6_26.

[5] Abdullah, M., O. Altiti and R. Obiedat (2022). Detecting propaganda techniques in english news articles using pre-trained transformers, In *2022 13th International Conference on Information and Communication Systems (ICICS)*, IEEE, Irbid, Jordan, pp. 301–308. https://doi.org/10.1109/ICICS55353.2022.9811117.

[6] Verma, P.K., P. Agrawal, V. Madaan and C. Gupta (2022). UCred: fusion of machine learning and deep learning methods for user credibility on social media. *Social Network Analysis and Mining*, Springer, 12(1), pp. 54. https://doi.org/10.1007/s13278-022-00880-1.

[7] Karande, H., R. Walambe, V. Benjamin, K. Kotecha and T.S. Raghu (2021). Stance detection with BERT embeddings for credibility analysis of information on social media. *PeerJ Computer Science*, PeerJ, 7, p. e467. https://doi.org/10.7717/peerj-cs.467.

[8] Liu, J., T. Singhal, L.T.M. Blessing, K.L. Wood and K.H. Lim (2021). Crisisbert: a robust transformer for crisis classification and contextual crisis embedding, In *Proceedings of the 32nd ACM Conference on Hypertext and Social Media*, Association for Computing Machinery (ACM), New York, NY, USA, pp. 133–141. https://doi.org/10.1145/3465336.3475117.

[9] Balouchzahi, F., H.L. Shashirekha and G. Sidorov (2021). MUCIC at CheckThat! 2021: FaDo-Fake News Detection and Domain Identification using Transformers Ensembling., In *CLEF (Working Notes)*, CEUR Workshop Proceedings, Bucharest, Romania, pp. 455–464.

[10] Ogundokun, R.O., M.O. Arowolo, S. Misra and I.D. Oladipo (2022). Early detection of fake news from social media networks using computational intelligence approaches. *Combating Fake News with Computational Intelligence Techniques*, Springer, pp.71–89. https://doi.org/10.1007/978-3-030-90087-8_4.

[11] Kaliyar, R.K., A. Goswami and P. Narang (2021). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia Tools and Applications*, Springer, 80(8), pp. 11765–11788. https://doi.org/10.1007/s11042-020-10183-2.

[12] Khanam, Z., B.N. Alwasel, H. Sirafi and M. Rashid (2021). Fake news detection using machine learning approaches, In *IOP Conf Ser Mater Sci Eng*, IOP Publishing, Jaipur, India, p. 012040. DOI: 10.1088/1757-899X/1099/1/012040.

[13] Bangyal, W.H., R. Qasim, N.U. Rehman, Z. Ahmad, H. Dar, L. Rukhsar, Z. Aman and J. Ahmad (2021). Detection of Fake News Text Classification on COVID-19 Using Deep Learning Approaches. *Computational and Mathematical Methods in Medicine*, Wiley Online Library, 2021(1), pp. 5514220. https://doi.org/10.1155/2021/5514220.

[14] Alsuwat, E. and H. Alsuwat (2025). An improved multi-modal framework for fake news detection using NLP and Bi-LSTM. *The Journal of Supercomputing*, Springer, 81(1), pp. 177. https://doi.org/10.1007/s11227-024-06671-z.

[15] Altamimi, A (2024). Novel approach for predicting fake news stance detection using large word embedding blending and customized CNN model. *PloS One*, PLOS, 19(12), p. e0314174. https://doi.org/10.1371/journal.pone.0314174.

[16] Ellaky, Z., F. Benabbou, Y. Matrane and S. Qaqa (2024). A hybrid deep learning architecture for social media bots detection based on BiGRU-LSTM and GloVe word embedding. *IEEE Access*, IEEE. https://doi.org/10.1109/ACCESS.2024.3430859.

[17] Abdal, M.N., M.H.K. Oshie, M.A. Haque and S. Rahman (2023). A Transformer Based Model for Bangla Fake News Detection Using DistilBERT, In *2023 5th International Conference on Sustainable Technologies for Industry* 5.0 (STI), IEEE, Dhaka, Bangladesh, pp. 1–6. https://doi.org/10.1109/STI59863.2023.10465127.

[18] Kula, S., R. Kozik, M. Choraś and M. Woźniak (2021). Transformer based models in fake news detection, In *International Conference on Computational Science*, Springer, Cham, pp. 28–38. https://doi.org/10.1007/978-3-030-77970-2_3.

[19] Chabukswar, A., P.D. Shenoy and K.R. Venugopal (2024). A Hybrid DistilBERT-BiGRU Model for Enhanced Misinformation Detection: Leveraging Transformer-Based Pretraining Language Model, In *2024 IEEE Region 10 Symposium (TENSYMP)*, IEEE, New Delhi, India, pp. 1–6. https://doi.org/10.1109/TENSYMP61132.2024.10752290.

[20] Zhi, X., L. Xue, W. Zhi, Z. Li, B. Zhao, Y. Wang and Z. Shen (2021). Financial fake news detection with multi fact CNN-LSTM model, In *2021 IEEE 4th International Conference on Electronics Technology (ICET)*, IEEE, Chengdu, China, pp. 1338–1341. https://doi.org/10.1109/ICET51757.2021.9450924.

[21] Lo, S.-C.B., H.-P. Chan, J.-S. Lin, H. Li, M.T. Freedman and S.K. Mun (1995). Artificial convolution neural network for medical image pattern recognition. *Neural Networks*, Elsevier, 8(7–8), pp. 1201–1214. https://doi.org/10.1016/0893-6080(95)00061-5.

[22] Li, Z., F. Liu, W. Yang, S. Peng and J. Zhou (2021). A survey of convolutional neural

networks: analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, IEEE, 33(12), pp. 6999–7019. https://doi.org/10.1109/TNNLS.2021.3084827.

[23] Gu, J., Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang and J. Cai (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, Elsevier, 77, pp. 354–377. https://doi.org/10.1016/j.patcog.2017.10.013.

[24] Patil, A. and M. Rane (2021). Convolutional neural networks: an overview and its applications in pattern recognition. *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2020*, Springer, Singapore, Volume 1, pp. 21–30. https://doi.org/10.1007/978-981-15-7078-0_3.

[25] Xu, Z., S. Qian, X. Ran and J. Zhou (2022). Application of Deep Convolution Neural Network in Crack Identification. *Applied Artificial Intelligence*, Taylor & Francis, 36(1), pp. 2014188. https://doi.org/10.1080/08839514.2021.2014188.

[26] Shyam, R (2021). Convolutional neural network and its architectures. *Journal of Computer Technology & Applications*, STM Journals, 12(2), pp. 6–14p. DOI: 10.37591/JoCTA.

[27] Yang, J. and J. Li (2017). Application of deep convolution neural network, In *2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, IEEE, Chengdu, China, pp. 229–232. https://doi.org/10.1109/ICCWAMTIP.2017.8301485.

[28] Gao, B. and L. Pavel (2017). On the properties of the softmax function with application in game theory and reinforcement learning. *ArXiv Preprint ArXiv:1704.00805*, Cornell University. https://doi.org/10.48550/arXiv.1704.00805.

[29] Banerjee, K., R.R. Gupta, K. Vyas and B. Mishra (2020). Exploring alternatives to softmax function. *ArXiv Preprint ArXiv:2011.11538*. Cornell University. https://doi.org/10.48550/arXiv.2011.11538.

[30] Hort, M., R. Moussa and F. Sarro (2023). Multi-objective search for gender-fair and semantically correct word embeddings. *Applied Soft Computing*, Elsevier, 133, pp. 109916. https://doi.org/10.1016/j.asoc.2022.109916

[31] Badri, N., F. Kboubi and A.H. Chaibi (2022). Combining fasttext and glove word embedding for offensive and hate speech text detection. *Procedia Computer Science*, Elsevier, 207, pp. 769–778. https://doi.org/10.1016/j.procs.2022.09.132.

[32] Hochreiter, S (1997). Long Short-term Memory. *Neural Computation MIT-Press*, IEEE. https://doi.org/10.1162/neco.1997.9.8.1735.

[33] Gülmez, B (2023). Stock price prediction with optimized deep LSTM network with artificial rabbit's optimization algorithm. *Expert Systems with Applications*, Elsevier, 227, pp. 120346. https://doi.org/10.1016/j.eswa.2023.120346.

[34] Kang, Q., E.J. Chen, Z.-C. Li, H.-B. Luo and Y. Liu (2023). Attention-based LSTM predictive model for the attitude and position of shield machine in tunneling. *Underground Space*, Elsevier, 13, pp. 335–350. https://doi.org/10.1016/j.undsp.2023.05.006.

[35] Luo, J. and Y. Gong (2023). Air pollutant prediction based on ARIMA-WOA-LSTM model. *Atmospheric Pollution Research*, Elsevier, 14(6), pp. 101761. https://doi.org/10.1016/j.apr.2023.101761.

[36] Patra, S.R. and H.-J. Chu (2023). Regional groundwater sequential forecasting using global and local LSTM models. *Journal of Hydrology: Regional Studies*, Elsevier, 47, p. 101442. https://doi.org/10.1016/j.ejrh.2023.101442.

[37] Sanh, V., L. Debut, J. Chaumond and T. Wolf (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv:1910.01108*, Cornell University, pp. 2–6. https://doi.org/10.48550/arXiv.1910.01108.

[38] Büyüköz, B., A. Hürriyetoğlu and A. Özgür (2020). Analyzing ELMo and DistilBERT on socio-political news classification, In *Proceedings of the Workshop on Automated Extraction of Socio-Political Events from News* 2020, European Language Resources Association (ELRA), Marseille, France, pp. 9–18.

[39] Abadeer, M (2020). Assessment of DistilBERT performance on named entity recognition task for the detection of protected health information and medical concepts, In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, Association for Computational Linguistics, Online, pp. 158–167. https://doi.org/10.18653/v1/2020.clinicalnlp-1.18.

[40] Singh, P., N. Singh, K.K. Singh and A. Singh (2021). Diagnosing of disease using machine learning, In *Machine Learning and the Internet of Medical Things in Healthcare*, Elsevier, Amsterdam, Netherlands, pp. 89–111. https://doi.org/10.1016/B978-0-12-821229-5.00003-3.

[41] Reddi, S. and G. V Eswar (2021). Fake news in social media recognition using Modified Long Short-Term Memory network, In *Security in IoT Social Networks*, Elsevier, Amsterdam, Netherlands, pp. 205–227. https://doi.org/10.1016/B978-0-12-821599-9.00009-1.