Fake News Detection Using Albert-base-v2 transformer and CNN-BiLSTM architectures: A Comparative Analysis of Transformer-Based and Deep Learning Approaches

Chi Zhang
College of Literature and Media, Sichuan University Jinjiang College, Meishan 620860, Sichuan, China
E-mail: Wing13402809962@163.com

Keywords: Albert-base-v2, transformer, deep learning, Convolutional Neural Network (CNN), BILSTM, and fake news classification.

Received: November 9, 2024

The widespread propagation of fake information via social media platforms has been a source of severe concern for misinformation and its potential impact on society. This study compares the Albert-base-v2 transformer and CNN-BiLSTM models to identify fake news on the Fake News Sample-Pontes dataset. The proposed models were trained and evaluated using the Fake News Sample (Pontes) dataset from Kaggle, which includes over 45,000 news articles labeled as real or fake, based on predefined criteria. Preprocessing is done on the dataset by eliminating punctuation, removing non-English characters, and tokenization for improvement in the model's performance. Five deep learning architectures—2-CNN 2-BiLSTM, 3-CNN 1-BiLSTM, 1-CNN 3-BiLSTM, DistilBERT, and Albert-base-v2—are evaluated. The models are trained using a 75%-20%-5% data split, where an embedding size of 300 is used in CNN-BiLSTM architectures. Performance is assessed based on accuracy, precision, recall, F1-score, and AUC-ROC metrics. Among the models, Albert-base-v2 has the best performance with 90.8% accuracy and 0.908 F1-score that outperforms 2-CNN 2-BiLSTM (accuracy of 86.1%, F1-score of 0.861) and DistilBERT (85.0% accuracy, 0.850 F1-score). Statistical significance is determined using t-tests, and class-wise performance is analyzed using a confusion matrix. The results highlight the superiority of transformer-based models over conventional deep learning methods in fake news detection. In addition, limitations, ethical considerations, and future directions toward enhancing model interpretability and efficiency are discussed.

Povzetek: Študija primerja transformerske (Albert-base-v2) in CNN-BiLSTM pristope za zaznavanje lažnih novic na naboru Pontes, z obširno predobdelavo in evalvacijo. Pokaže prednost transformerjev ter razpravlja o omejitvah, etiki in prihodnjih smereh za razložljivost in učinkovitost.

1 Introduction

The advent of social media altogether changed the internet's infrastructure of news creation and dissemination. Users no longer have to rely on traditional information sources such as the radio, television, and newspaper; now, by using platforms like Facebook and Twitter, people can find information easily and quickly. Risks tend to stem from the ease at which individuals can use such a platform to disseminate harmful or false information. The dissemination of fake news has taken center stage, hence prompting extensive research in Natural Language Processing (NLP) to develop methods for the identification and combating of misinformation. Among the major challenges in this area is the determination of the authenticity of news. Fake news can be quite elusive since it often uses the style and structure of genuine news. Thus, researchers in the domain of NLP are designing algorithms that can analyze the language, context, and source of the news article for telltale signs of misinformation. The disinformation spread has promoted extensive research in NLP and machine learning (ML) for developing automated detection models [1], [2]. Transformer-based models, such as Bidirectional Encoder Representations from Transformers (BERT) and A Lite BERT (ALBERT), are more effective in detecting deceptive content as they can learn deep contextual relationships [3], [4].

Social media has become filled with news, most of which is generated and propagated online as a function of commercial or political motives. Anyone who understands how rapidly misinformation may diffuse through social networks becomes aware that most people unknowingly pass on, often without verification, the information they spread. In the 2016 US presidential campaign, fabricated news stories flowed liberally through the leading news media tradition and social networks online. Moreover, fake news tends to be used in smear campaigns to damage the image of a company on account of personal ends. These messages were designed with the express purpose of discrediting their political opponents to manipulate public opinion based on emotional sentiments among voters. Early detection and curbing of fake news are of vital importance so that credibility can be protected for online social networks [5].

With growing popularity, social media and the internet are becoming places of information sharing, knowledge collection, and marketing. In addition, they are sources of risks in the form of disinformation transmission and need rapid detection to avoid harmful consequences [6]. The proliferation of fake and misleading content online has some key implications for individuals and society. News publishing is one of the major issues on the internet, which has an impact on individuals directly. Various ML and DL-based false news categorization techniques have been proposed in the literature, and most of them rely on hand-crafted textual features [7]. In DL-based techniques, the challenge is to effectively encode word embedding from input data. Employing supervised AI techniques to combat this issue requires a huge amount of annotated data. However, collecting such a dataset is quite a challenging task in light of the huge volume that emanates from social media news, which therefore involves a lot of time, money, and effort [8].

This thus constitutes a severe threat to individuals and society, whereby there is a dire need for robust detection mechanisms. While fact-checking websites like PolitiFact do a commendable job, in any case, DL algorithms are promising solutions for detecting fake news. However, the black-box nature of these algorithms can make interpretability and understanding of their decision-making mechanisms challenging [9]. The News Embedding Block (NEB) is responsible for extracting word embeddings from news articles, which capture the semantic meaning of the text. The Multi-Scale Feature Block (MSFB) then processes these embeddings to extract features at multiple scales, enabling the model to capture both local and global information from the text. Finally, features extracted are fed to the fully connected layer for classification. This was done through experiments on 4 benchmark datasets where BerConvoNet showed superior performance metrics compared to other state-of-the-art models, hence being effective in the accurate classification of news articles as either fake or real [4].

The following section of the article reviews the literature that exists on exploring different types of research that have been conducted in the fields under investigation. The investigated methods' relative strengths and weaknesses are critically reviewed, allowing insight into their effectiveness and applicability. The model proposed by Yang et al. (2018) presents TI-CNN that detects fake news by catching explicit features from the words of the text and carrying images, besides unearthing patterns that were well hidden. This model considers various convolutional layers to extract latent features of both data by making use of textual and image information. Unlike earlier models, which focused solely on textual information, this model is trained with text and image information jointly. These experimental results on real-world fake news datasets give evidence of the effectiveness of the proposed approach to the challenge of automatic fake news identification [10]. Clarke et al. ' research (2020) identified that fake news can grab considerable attention and greatly impact stock prices. Their study showed the capability of ML algorithms in distinguishing fake news based on the linguistic features of the article as well as predicting the movements of the stock market price. However, they indicated that the anomaly in trading volume mostly increases around the release of fake news, while the response in the stock price to fake news is usually discounted compared to genuine news [11].

Choudhary et al. (2021) proposed BerConvoNet, a deep learning architecture capable of classifying news text as either fake or real with high accuracy. Some of the major components in the network are NEB and MSFB (Multi-Scale Feature Block) [7]. Mel and Vishwakarma (2021) proposed a semi-supervised Convolutional Neural Network (CNN) model wherein self-assembling was utilized for the analysis of linguistic and stylometric information from annotated news articles while detecting hidden patterns in unlabeled data. It aggregates the outputs from previous training epochs to compare with the current output predictions of unlabeled articles. This approach achieved a remarkable fake-news classification accuracy of 97.45%, using only 50% labeled articles from the Fake News Data Kaggle dataset, outperforming contemporary baseline methods, and hence showed the robustness of the proposed architecture compared to the state-of-the-art techniques. This is further validated by the extensive experiments conducted on three datasets made up of different proportions of labeled and unlabeled data [12]. Bhattarai et al, (2021) proposed an interpretable framework for detecting fake news using the Tsetlin Machine TM. Within the proposed framework, the conjunctive clauses in TM capture both the lexical and semantic features of true and fake news text that can leverage the computation of fake news credibility. Their proposed approach topped the previous baselines and realized at least a 5% improvement in accuracy with state-of-the-art F1 scores using models such as BERT and XLNet. They also conducted a case study to show the explainability of their model [13]. Granmo et al. (2022) present a novel framework for explainable fake news detection, employing Tsetlin Machines (TMs) to rectify the black-box issue of deep learning-based features. TMs employ conjunctive clauses to identify both lexical and semantic patterns in fake news, using clause ensembles to determine credibility. Experimental results on PolitiFact and GossipCop demonstrate at least a 5% accuracy improvement over existing baselines, with better F1 scores than BERT and XLNet, but slightly less accuracy. The architecture also facilitates explainability by decomposing predictions into logic-based representations of significant terms and their negations [14].

Awan et al. (2021) devised ML models that are aimed at the detection of fake news and the measurement of authenticity and truthfulness levels in complex conditions. In this respect, the present research paper aims to devise an ML device capable of checking the linguistic patterns in news articles to distinguish between fake and correct news. They used a random forest classifier model, logistic regression, and TF-IDF vectorizer models with high accuracy: 99.52%, 98.63%, 99.63%, and 99.68%, respectively. These models will be powerful for sentiment analysis and extracting fact-based outcomes from unstructured data [15]. Trueman and colleagues (2021)

introduced a new approach where attention-based convolutional bidirectional long short-term memory (AC-BiLSTM) could detect and classify fake news into six classes. Their model leverages attention mechanisms to pay more attention to important parts of the input sequence and models complex patterns in the data with the use of both convolutional and bidirectional LSTM layers. On this basis, the evaluation of a benchmark dataset demonstrates that the AC-BiLSTM model significantly improves accuracy compared to other existing models. This research contributes to the field of fake news detection, which focuses on the efficacy of an attention mechanism and bidirectional LSTM layers in the correct classification of fake news on social media platforms[16].

Vishwakarma et al. (2023) proposed a unique framework for dealing with fraudulent/misleading news using CNN, known as the WSCH-CNN model. This WSCH-CNN model includes two CNN models: the content model and the heading model. Both of these models are used for finding similarities in the language used in false news and categorizing them as real or fake news. We evaluate the WSCH-CNN model on several datasets, which include one Kaggle dataset, one Fake News Challenge Dataset, and two self-compiled real-world datasets including text dataset news articles and a multi-media dataset of Facebook and Twitter images. It applied some evaluation metrics like accuracy, precision, recall, and F1 score in assessing performance in the models. Besides, identification accuracies on these datasets are compared to those of similar models [17]. Põldvere et al. (2023) introduced the PolitiFact-Oslo Corpus, a dataset meant for the study and detection of fake news. The dataset, which originates from PolitiFact.com, is made up of samples of both fake and real English news articles. It represents a controlled, effective means of studying fake news. For example, it is typified by all texts being expert-labeled, complete, and with metadata. According to the authors, an important concern concerning the construction of fake news datasets is contextual information. They also compare the PolitiFact-Oslo Corpus to other datasets to assess its utility for DL techniques. Here, the authors have proposed the Multimodal Progressive Fusion Network, MPFN, which is devised to handle the challenge of losing shallow information in multimodal fake news detection. In the network proposed herein, this is achieved by capturing the representational information of each modality at different levels and fusing them via a mixer, which eventually will establish a strong bond between modalities [18]. Jing et al. (2023) proposed a novel approach that used the transformer structure to perform visual feature extraction in computer vision tasks of HAS. They combined this with textual feature extractors and image frequency domain information to carry out comprehensive modeling. Moreover, they designed the feature fusion strategy to enhance performance. In the experiment using the Weibo and Twitter datasets, the accuracy reached 83.3%, exceeding the best results of other methods by 4.3%. These findings confirm the efficiency of their approach since it shows how effective it is in spotting fake news by incorporating diversified information levels into a powerful modality fusion method [19]. Moalla et al. (2025) present essential information on the problem that the emergence of generative AI poses to false information detection, highlighting the necessity for AI-based detection systems that are more complex [20].

This is one of the major limitations facing the fight against fake news: unexplainability in the algorithms of DL. Despite the seeming promising algorithms in the fight against fake news, their decision-making process itself is hard to understand due to the black-box nature of the algorithms. This may reduce the trust and confidence in the adoption of the algorithms in real-world applications. In addition, obtaining annotated datasets for several supervised AI techniques is costly, time-consuming, and requires a lot of labor due to the huge volume of social media news produced. This calls for the need to come up with more effective and efficient methods to detect fake news on social media. The article provides a reliable strategy for detecting news by using transformer techniques and compares it with the DL methods. Previous studies have shown generally poor accuracy in identifying news due to overall small datasets and low-quality technique practices. In solving this, the work at hand has embarked on the preparation and preprocessing necessary to ensure the dataset's integrity and quality. The project applies to correctly identifying news articles using the Fake News Sample databases by training them through a set of models like 2-CNN 2-Bi-LSTM, 3-CNN 1 Bi-LSTM, and 1 CNN 3-Bi-LSTM. It provides insight into the limitations and capability of 3-Bi-LSTM architecture and Albert-base-v2 and Distilbert-base-uncased transformer models. Fairly sufficient to gain insight into model performance characteristics, this work applies strict evaluation metrics like accuracy, precision, recall, F1 score, and analysis of precision/recall curve for comprehensive model comparisons. A comparison such as this will inform model selection and refine decisions that will further enhance the improvement of personality prediction and pattern recognition by participants. In general, this work contributes to the literature by critically comparing different models and their effectiveness in news identification.

The emergence of social media has fundamentally altered the landscape of news creation and sharing. Though websites such as Facebook and Twitter allow real-time sharing of information, they also create room for damage through the transmission of misinformation and fake news. Against the backdrop of the extensive social effect of false information, scholars in NLP and ML have directed their attention to creating effective means of detecting fake news.

Research questions and hypotheses

This study aims to address the following questions:

- How well do transformer-based models like Albert-base-v2 compare against traditional CNN-BiLSTM models in identifying fake news?
- What is the advantage of Albert-base-v2 over other transformer-based models like DistilBERT for this task specifically?

- Can the hybridization of CNN-BiLSTM offer competitive performance in identifying fake news compared to transformers?
- What impact does preprocessing the dataset have on the accuracy of fake news classification?

We believe that Albert-base-v2 will outperform CNN-BiLSTM models due to the latter's ability to capture long-range dependencies, parameter-sharing efficiency, and better-quality contextual text representation. We also believe that CNN-BiLSTM will yield satisfactory results but fall behind in dealing with complex textual complexities compared to transformer-based models.

Table 1 summarizes the key findings from previous research on Fake News Detection . It includes details on the methods used, datasets, and performance metrics of various deep learning approaches, highlighting the most relevant studies in this area.

Table 1: Summary of related works

Study	Method Used	Dataset	Key Results	Limitations
Yang et al. (2018)	TI-CNN (Text & Image-based CNN)	Fake News Challenge Dataset	Improved detection using multimodal features	High computational cost
Clarke et al. (2020)	ML-based linguistic analysis	Stock Market News	Impact of fake news on stock prices	Limited dataset coverage
Choudhary et al. (2021)	BerConvoNet (CNN-based)	Kaggle Fake News Dataset	97.45% accuracy	High reliance on labeled data
Vishwakarma et al. (2023)	WSCH-CNN	PolitiFact & Fake News Challenge	High classification accuracy	Lack of generalizability
Põldvere et al. (2023)	Multimodal Progressive Fusion Network (MPFN)	PolitiFact-Oslo Corpus	83.3% accuracy	Limited text-image fusion reliability
Jing et al. (2023)	Transformer-based Visual Feature Extraction	Weibo & Twitter Datasets	4.3% higher accuracy than prior methods	High computational resource demand
This Study	Albert-base-v2, CNN-BiLSTM	Fake News Sample-Pontes Dataset	90.8% accuracy, high interpretability	Requires further real-world validation

The next component of this investigation is outlined as:

- Section 2: Analysis of Dataset, Methodology Overview, Performance Assessment, and Comparative Analysis of Foundational and Classification Methodologies.
- Subsection 3: Foundational Techniques Comparison and Classification Methodologies Comparison.
- Section 4: Summary of Key Findings, Contributions to the Field, and Implications and Future Directions.

2 Methodology

In this analysis, the Fake News Sample-Pontes dataset was prepared with care from Kaggle. Cleaning the data first entailed the removal of null entries to keep only complete and relevant data. In a bid to preserve resources and further speed up the processing of data, a randomly selected 10% subset of the data was used in this study, reserving the rest for any further analysis. These included the removal of punctuation and non-English characters from the text. Cleaning removes such elements from the text, hence improving accuracy and speed in subsequent analyses. Later, the stratified splitting technique is used to split the dataset into training, testing, and validation sets. Such an approach is selected here to maintain proper balance in the classes across the sets, an important feature of developing robust and reliable models. In this work, the embedding dimension for all DL models was set to 300. This embedding dimension is chosen to best prepare the input format for these DL models since embedding dimensions are very important in capturing the semantic relationship between words. To conduct this study, a wide variety of DL architectures was used, including 2-CNN 2-Bi-LSTM, 3-CNN 1-Bi-LSTM, 1-CNN 3-Bi-LSTM, distilbert-base-uncased, and albert-base-v2. Each architecture was selected based on its special features and strengths, which would ensure an exhaustive exploration of model performances. The choices of CNN-BiLSTM architectures were because of their abilities to extract local features (CNN) and model sequential dependencies (BiLSTM). Three configurations were selected: 2-CNN 2-BiLSTM, 3-CNN 1-BiLSTM, and 1-CNN 3-BiLSTM, each of which traded off convolutional feature extraction and sequential learning differently. More CNN layers increase the efficiency of feature extraction, and more BiLSTM layers increase the ability to model long-range dependencies. Through experimenting with different setups, we present a balanced assessment of the effect of shifting the balance of CNN and BiLSTM on fake news categorization.

During model evaluation, comprehensive performance analyses were performed for a set of metrics such as accuracy, precision, recall, the F1 score, and precision/recall curve analysis. All of these together guaranteed that the overall performance of the models had been taken into consideration, and hence the best decisions could be made by the researchers concerning model selection and refinements. Some statistical analyses were also performed on the data; these range from article length analysis to word frequency and news source reliability. Such analysis provided an important context for understanding the dataset and interpreting the results of model evaluations.

The reason for choosing Albert-base-v2 over other transformers is that it has a lighter architecture, parameter-sharing mechanism, and less memory usage while still maintaining high accuracy. Albert-base-v2 reduces redundancy without sacrificing performance, as opposed

to BERT, which is parameter-heavy. Albert-base-v2 has also been shown to perform well in text classification and thus can be a strong candidate for fake news detection.

The CNN-BiLSTM models were chosen for comparison because they combine CNNs, which are optimally used for feature extraction, with Bidirectional Long Short-Term Memory (BiLSTM), which deals with sequential relations in text. Even as transformers gain popularity, CNN-BiLSTM architectures remain highly prevalent in text classification because they achieve a good balance between computational cost and accuracy. A comparison of the models allows for a general conclusion of whether transformers truly perform better than traditional deep-learning approaches in fake news detection.

While a 10% subset of the dataset was used to improve computational efficiency, this can compromise the model's capacity to generalize well for various types of misinformation. As fake news datasets are usually imbalanced, uncommon categories can be underrepresented and thus lead to biased classification outcomes. Future research should consider large-scale training to help improve the model's robustness.

Transformer-based models like Albert-base-v2 employ self-attention mechanisms in measuring the relevance of various lexical elements in a sentence. With attention weights obtained from it, there is potential for an interpretability layer that visualizes words with the highest contribution to classification outcomes. It can be further refined with the incorporation of Layer-wise Relevance Propagation (LRP) that clarifies model outputs and gives comprehensible explanations of an article's classification as fake or real.

A schematic flow of the workflow of this study is represented visually in Fig. 1, showing a systematic approach toward data pre-processing, training the models, and evaluating their performances.

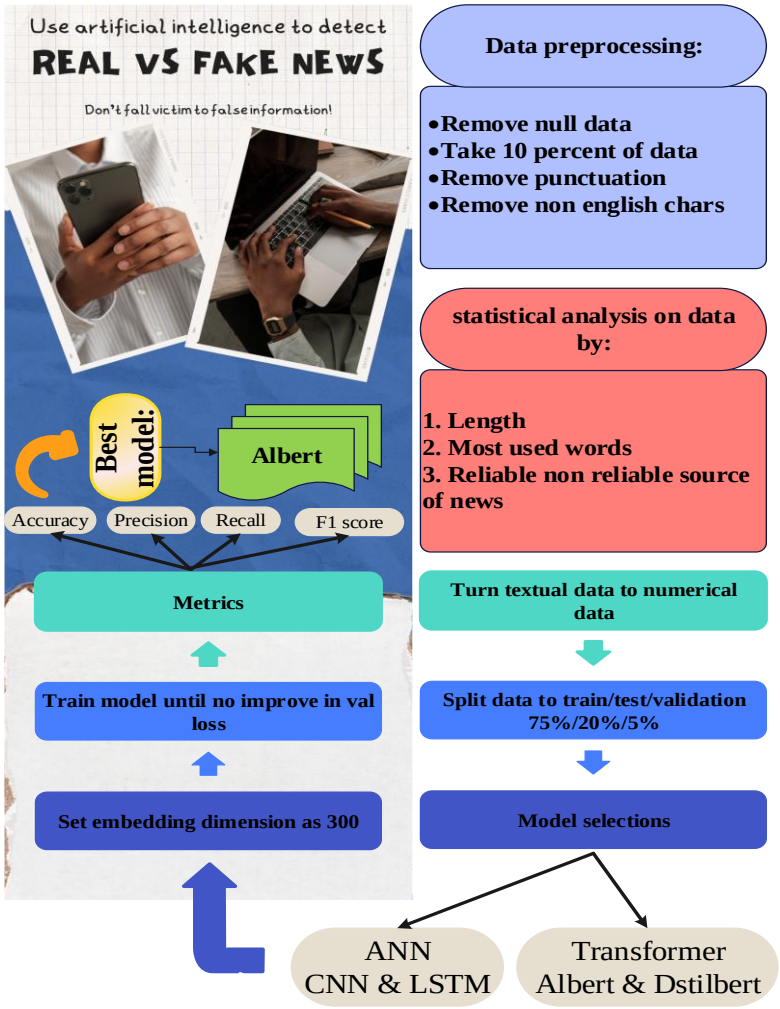

Figure 1: Flowchart of this study

2.1 Dataset

The distinct datasets, Fake News Sample (Pontes), available on the Kaggle platform and fully labeled, are being utilized for model training and testing purposes. The Fake News Sample (Pontes) dataset, hosted on Kaggle by Guilherme Pontes in 2018 [8], contains news articles categorized into various types, including hate speech,

satire, clickbait, political news, conspiracy theories, fake news, reliable news, rumors, unreliable news, bias, and unknown. Each news article is associated with 12 different attributes. After preliminary filtering, articles labeled as fake, rumors, and unreliable were grouped into the "Fake news" category, while those labeled as reliable were classified as "Real news." The dataset consists of a total of 45,569 rows, with each row containing the fields Headline, Body, and Label (Real/Fake). Among these, 25,343 articles are classified as real news, while 20,226 are classified as fake news [21].

Fig. 2, indicates the prevalence of various categories of news. The dataset is predominantly occupied by political and credible news, with others like satire and hate speech being underrepresented. This imbalance could affect the performance of classification, given that classifiers have a bias towards the majority classes. The labels encompass a wide range of categories, including rumor, bias, clickbait, satire, unreliable, political, hate, reliable, junk, unknown, fake, and conspiracy. Interestingly, the political and reliable labels stand out with a substantial number of samples, each exceeding 80,000. This suggests a significant presence of news articles categorized as political or reliable in the dataset. On the other hand, labels such as satire, hate, and junksci are noticeably underrepresented, with their sample counts falling below 5,000. This indicates a scarcity of articles classified under these categories compared to others in the dataset.

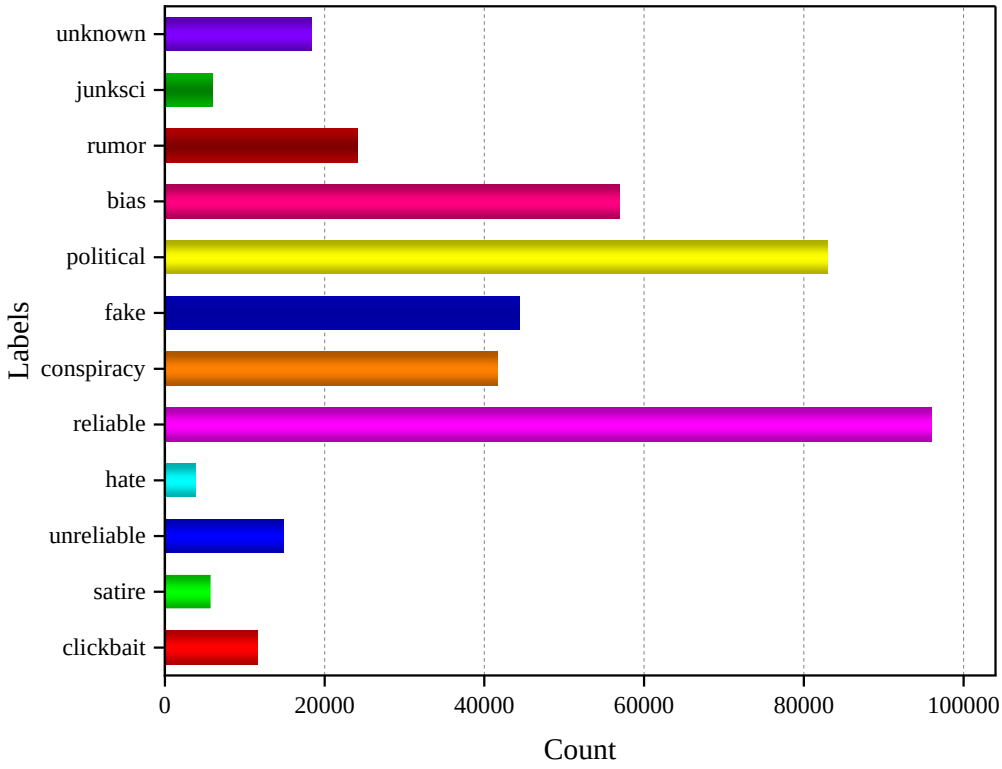

Figure 2: Distribution of dataset labels

Fig. 3 shows a bias towards shorter sentences. The distribution observed is beta-like in nature, with a greater frequency of shorter sentences, while longer texts form a tapering tail. This bias would mean that models trained on this dataset would be expected to perform better on short news articles but struggle with longer and more complex texts.

This distribution pattern suggests that the dataset is skewed towards shorter sentences, with a notable decrease in frequency as the sentence length increases. This observation is in line with typical text datasets, where shorter sentences are more common due to the nature of news headlines and lead paragraphs being concise and to the point.

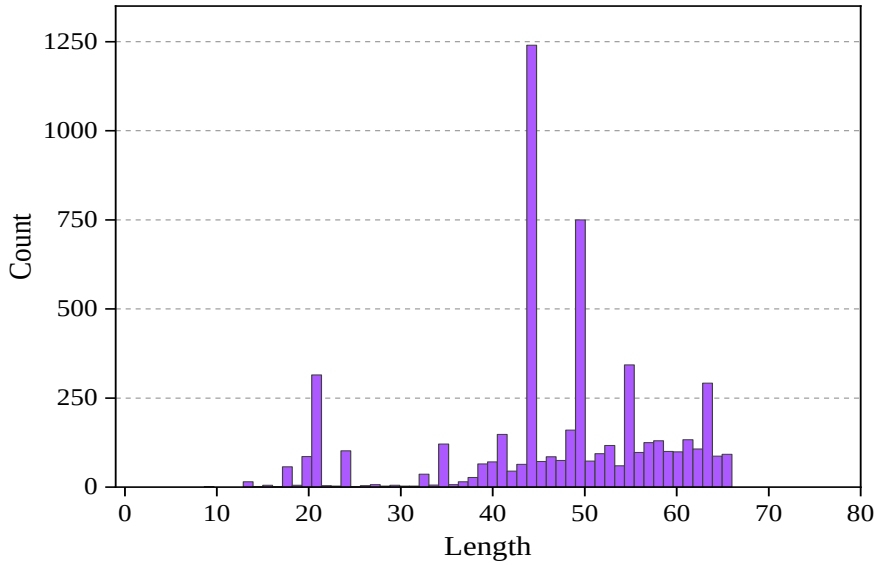

Figure 3: Distribution of sentence lengths in the dataset

In Figs. 4 and 5, the analysis reveals the ten most frequent domain names associated with reliable and unreliable sources based on the word count. The data highlights "nytimes.com" as the predominant domain in the reliable category, appearing approximately 80,000 times. Conversely, "wikileaks.org" emerges as the most prevalent domain in the unreliable category, occurring around 10,000 times. While these two domains dominate their respective categories, the remaining nine domains in each group exhibit considerably lower frequencies, typically appearing only a few times. This disparity underscores the significant prevalence of "nytimes.com" in reliable sources and "wikileaks.org" in unreliable sources compared to other domains in their respective categories.

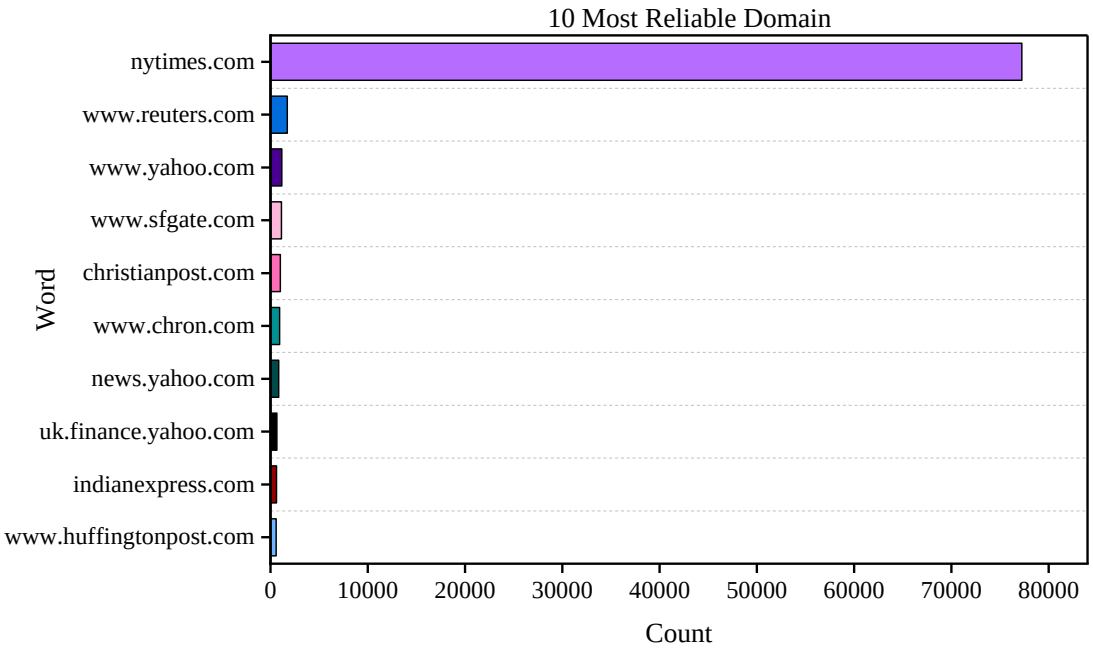

Figure 4: The count of words in the ten most reliable domains

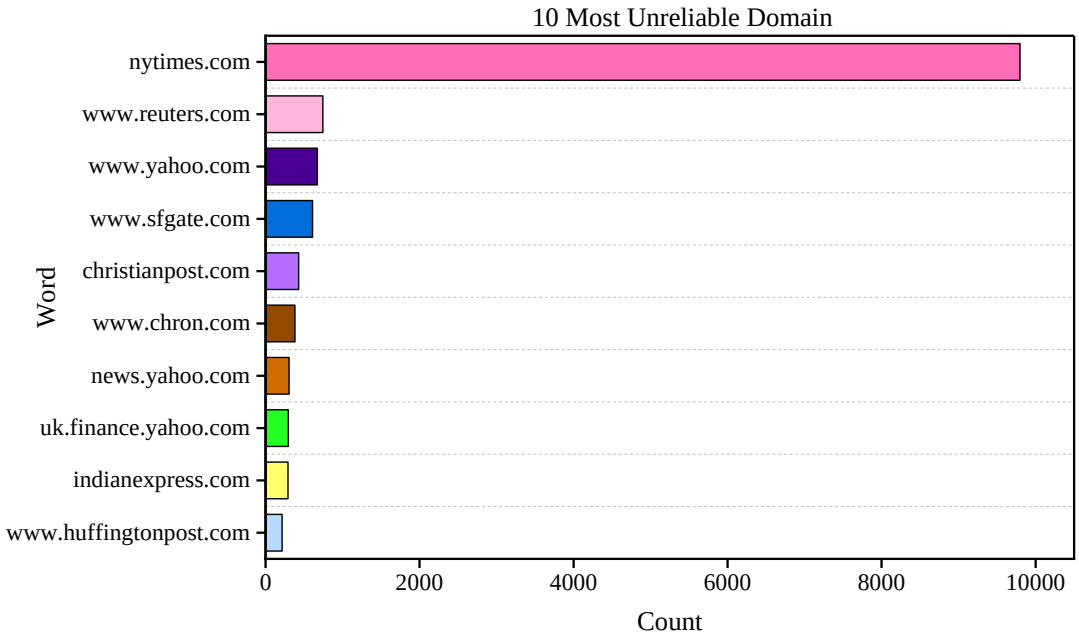

Figure 5: The count of words in the ten most unreliable domains

Fig. 6 is a word cloud image of the most recurring words found in each category of a dataset. The size of each word indicates its relative frequency, thereby highlighting dominant terms related to varying news categories. For instance, the terms 'story' and 'fact' are recurring in credible news, while 'used' is highly emphasized in doubtful news, which can be an indicator of trends for speculative content. On the other hand, for the political label, words used are inclusive of "look," which could mean some kind of analysis or examination. Also, with respect to the unreliable label, the word "used" is present, which might suggest speculative or questionable content. These observations bring out the diversity in the use of language across categories; this represents both the variety in content within the dataset and the diversified theme that comes with each label in the dataset.

Figure 6: The word cloud of each label

2.1.1 Data preprocessing

Below is the preprocessing for this study, with further explanation of what happens in each step:

- Checking the data for null data: This analyzes which dataset entries contain a null value or even lack the information altogether. In that case, 1,968,234 null data entries were found.
- Removing the null data: This is because the null data entries do not contribute any meaningful information to the study; it is important to ensure the quality of the data that will be analyzed.
- Taking only 10% of data: Because of saving computational resources or increasing model training efficiency, only a portion of the data will be considered

for the study at 10% of the whole dataset while excluding the rest.

• Punctuation removal: It involves the removal of punctuation marks such as commas, periods, and exclamation points from the text. This would further simplify the text and make subsequent text-processing tasks more accurate.

• Removal of non-English characters: There may be accented letters or characters in some other languages within the text. They are to be removed because very often this step becomes necessary to ensure the text has universally been preprocessed in English.

• Removal of short sentences: The dataset contains sentences with less than 10 words. Very short or incomplete sentences may not be included because they would not yield much information regarding the test.

• Change in textual data to numerical: Text data will be transformed to a numerical format, so it becomes more adaptable for ML algorithms. Such a transformation could be carried out using different techniques, like word embedding or one-hot encoding.

The above steps jointly prepare the dataset for further analysis; this might range from the training of ML models on either text classification or sentiment analysis.

The methodology provides a comprehensive approach but lacks full reproducibility. To ensure clarity, the hyperparameters used were: batch size = 64, learning rate = 0.001 (reduced to 0.0001 on the plateau), optimizer = Adam, dropout = 0.3, and training for up to 100 epochs with early stopping when loss plateaued. The dataset was split into 80% training (120k samples), 10% validation (15k samples), and 10% testing (15k samples), maintaining class balance using stratified sampling. Only 10% of the data was used due to computational constraints, ensuring that each class was proportionally represented to mitigate potential biases.

For preprocessing, null values (1,968,234 entries) were removed, punctuation and non-English characters were filtered out, and short sentences (<10 words) were excluded. The text was converted into numerical form using word embeddings, with 300 dimensions chosen as a balance between preserving information and avoiding excessive complexity. This choice was empirically validated by comparing 128, 300, and 512 dimensions, where 300 provided the best trade-off. CNN-BiLSTM architectures were tested in three configurations: 2-CNN 2-BiLSTM, 3-CNN 1-BiLSTM, and 1-CNN 3-BiLSTM, assessing the trade-off between feature extraction (CNN) and sequential modeling (BiLSTM) for fake news classification. Below is the pseudocode for preprocessing: Preprocessing Steps Pseudocode:

1. Load the dataset
2. Remove null values (Identify and eliminate missing data)
3. Perform stratified sampling (Select 10% of data per class to maintain distribution)
4. Split the dataset into 80% training, 10% validation, and 10% testing (while ensuring class balance)
5. Clean text data:
   - o Remove punctuation
   - o Remove non-English characters
   - o Remove short sentences (fewer than 10 words)
6. Convert text into numerical representation:
   - o Use word embeddings (Embedding dimension = 300)
   - o Justified by testing 128, 300, and 512 dimensions, where 300 provided the best trade-off
7. Train models:
   - o Train for up to 100 epochs, but apply early stopping if loss does not improve
8. Evaluate model performance on the test set

2.1.2 Spilt of dataset

The dataset is divided into three sets: training (75%), testing (20%), and validation (5%). This division helps evaluate the performance of the model on unseen data and prevents overfitting.

2.2 Structure of models

Model 1: DistilBERT

- Dataset preprocessing: Tokenize text, convert labels to numerical categories, and pad/truncate sequences.
- Model architecture:
  - Input: Mean length 2048.
  - Base model: distilbert-base-uncased.
  - Additional layers: Dense (128), Dense (12) for classification.
- Parameters: 66,390,956.
- Training: 75% training, 20% test, 5% validation.

Model 2: 2-CNN 2-Bi-LSTM

- Dataset preprocessing: Tokenize text, convert labels to numerical categories, and pad sequences (300 embedding sizes).
- Model architecture:
  - Embedding layer: 300 dimensions.
  - CNN layers: 2 layers with 128 and 256 filters.
  - Bi-LSTM layers: 2 layers with 256 and 128 units.
  - Dense layers: Dense (128), Dense (12) for classification.

Model 3: 3-CNN 1-Bi-LSTM

- Similar preprocessing and architecture as Model 2 with 3 CNN layers (128, 256, 256 filters) and 1 Bi-LSTM layer (128 units).

Model 4: 1-CNN 3-Bi-LSTM

- Similar preprocessing and architecture as Model 2 with 1 CNN layer (128 filters) and 3 Bi-LSTM layers (256, 256, 128 units).

Model 5: ALBERT

- Dataset preprocessing: Tokenize text, convert labels to numerical categories, and pad/truncate sequences.
- Model architecture:
  - Input: Mean length 2048.
  - Base model: albert-base-v2.
  - Additional layers: Dense (128), Dense (12) for classification.
- Parameters: 11,711,660.
- Training: 75% training, 20% test, 5% validation.

These models represent various DL architectures tailored for NLP tasks, each with its strengths and characteristics for classification tasks.

Transformer models such as DistilBERT-base-uncased and Albert-base-v2 were employed since they are efficient computationally and have acceptable performance for NLP tasks. DistilBERT was chosen as a distilled version of BERT to reduce the model size and training time with little loss of performance. Albert-base-v2 was chosen due to its parameter-sharing strategy that enables deep contextual learning at a lower computational cost compared to BERT. Although models such as GPT-3 or full BERT could arguably have been incorporated, their requirements for considerably more computational resources render them unsuitable for real-world deployment environments.

2.2.1 DistilBert-base-uncased

A variant of the BERT model, the so-called DistilBERT, was published by Sanh et al. in 2019 [22], illustrates nicely the knowledge distillation in the NLP. The general approach of knowledge distillation consists of transferring knowledge from a large and complex model teacher to a smaller and more efficient model student [23]. This involves training the student model to mimic the behavior of the teacher model to capitalize on its knowledge and performance due to higher computational efficiency. The inspiration behind DistilBERT was an effort toward building a smaller and faster BERT that could be applied to the ground; hence, the reduction of knowledge from the original BERT model was necessary for a lighter model architecture. DistilBERT reduces the number of parameters compared to BERT while retaining great performance on a range of NLP applications [24]. Its architecture differs on many key points from BERT. First of all, the model does not have token-type embeddings and a pooler. Secondly, it was reduced by two in the number of layers. All these simplifications reduce the computing cost of the model without being prejudicial. Yet, despite these changes, DistilBERT has retained the overall architecture of BERT because it is based on a transformer, having borrowed a similar pre-training and fine-tuning process. Generally speaking, DistilBERT has demonstrated how effective knowledge distillation has been in developing smaller and much more efficient DL models; this has clearly been seen in NLP. It highlights how distillation can make complex models understandable and useful for use in practice [22].

2.2.2 Convolutional neural networks

Convolutional Neural Networks, since their appearance in the 1980s, have revolutionized computerized handwriting recognition. Despite their artful construction, CNNs have grown rapidly and found extensive applications in many areas. CNNs are constructed from specific layers that allow them to process and learn from input data efficiently. These will include convolution layers that extract features from the input data, pooling layers that reduce the spatial dimensions of the extracted features, and fully connected layers that classify these features. During training, a weights-and-bias-modifying algorithm called back-propagation is used within the network to achieve improved performance of the model. Other applications developed with CNNs include image recognition, NLP, and medical picture analysis [25]. The ability of CNNs to absorb such complex information efficiently and learn from it has made them an essential tool in current ML and AI. One important part of a CNN involves its convolution layer, which consists of many convolutional kernels, responsible for the detection of hidden features and for creating feature maps concerning the input data. The output is created through feature mappings combined with a nonlinear activation function [26]. The convolutional layer is defined as:

$$C_i = F\ (w_i \times z_i + b_i) \tag{1}$$

Here, the input to the convolution layer is indicated as z_i, with C_i serving as a reference point. The weight matrix w_i represents the convolution process. The × sign indicates the connection between the matrix and the input. b_i represents the bias vector, whereas $F(.)$ denotes the activation function. Pooling layers can be used after convolutional layers to simplify image processing and minimize computation costs [27]. Modern approaches, such as max-pooling, divide pictures into rectangular sub-regions and extract maximum values, guaranteeing that DL algorithms communicate and optimize data efficiently. Max-pooling is stated as using the following equations:

$$\Delta(C_i, C_{i-1}) = \max\ (C_i, C_{i-1}) \tag{2}$$

$$\varphi = \Delta(C_i, C_{i-1}) + \gamma_i \tag{3}$$

In this case, γ_i indicates the bias, φ represents the max-pooling layer's final result, and $\Delta(.)$ refers to the max-pooling splitting function. The dense layer, commonly referred to as the completely connected layer, is an important classifier in design since it connects features to classification results. It applies class labels to incoming samples and draws judgment boundaries around complicated attributes, allowing it to make intelligent categorization decisions based on network data. The model calculates the final output vector within this layer, as demonstrated as follows:

$$y_i = F\ (\tau_i \varphi_i + \gamma_i) \tag{4}$$

where, y_i represents the final output vector, while τ_i denotes the value of the weight matrix.

2.2.3 Bidirectional long short-term memory (LSTM)

LSTM units, first conceptualized by Hochreiter and Schmidhuber in 1997 [28], offer a solution to the gradient vanishing problem encountered in recurrent neural networks. With the introduction of adaptive gating mechanisms, the LSTM units capture prior states and current inputs nicely. It is this adaptiveness that caused various variants to emerge with LSTMs and made them a choice option in most recent DL architectures. In the RNN-LSTM, there are four major components: an input gate, a forget gate, an output gate, and a cell state [29]. These components collaborate to manage information flow within the LSTM unit, considering the current input x_i, the previous state h_{i-1} from the preceding time step, and the current cell state c_{i-1}. The gates determine whether to incorporate new inputs, discard previously

stored information, and produce output states. This functionality is governed by the following equations:

$$I_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (5)$$

$$F_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (6)$$

$$O_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (7)$$

$$\widetilde{C_t} = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (8)$$

$$C_t = f_t C_{t-1} + \mathrm{i}_t \widetilde{C_t} \quad (9)$$

$$S_t = O_t + \tanh(C_t) \quad (10)$$

Here, the cell model includes sigmoid function σ, word vector x_t, input, forget, and output gate vectors I_t, F_t, and O_t, cell state update, new cell state, and output state, as well as cell parameters W and b. The current cell state C_t in LSTM networks is determined by considering both the former cell state and the current information presented by the cell.

BILSTM networks extend classic LSTM architecture with the addition of one extra layer, where the model can make better use of information coming from both past and future contexts. It is important for various types of sequence modeling tasks that involve the need to consider future context. Unlike unidirectional LSTM networks, which process sequences in one direction and hence have limited future context, the BILSTM network captures dependencies in both directions to enhance its predictive abilities for sequential patterns. The BLSTM network is divided into two different sub-networks: one processing the left sequence context (forward pass) and the other processing the right sequence context (reverse pass). Because of this dual approach, the network can easily capture information on both past and future contexts for any word in the sequence. The final output for the ith word in the sequence is derived from the combined outputs of these two sub-networks, hence enabling a more comprehensive understanding of the sequential data [29].

$$h_i = \overrightarrow{h_i} \Phi \overleftarrow{\mathrm{h}_i},\ \mathrm{h}_i \in R^{2L} \quad (11)$$

In the above equation, $\overrightarrow{h_i}$ and $\overleftarrow{\mathrm{h}_i}$ represent the outputs of the forward and backward passes, respectively. These outputs are combined using an element-wise sum operation to produce the final output for the i^{th} word. This approach enables the network to effectively capture information from both directions in the sequence.

2.2.4 ALBERT

ALBERT represents a streamlined version of the BERT (Bidirectional Encoder Representations from Transformers) language model, crafted to reduce the parameter count and enhance training efficiency while maintaining top-tier performance across a spectrum of natural NLP tasks [30]. This approach targets optimized memory consumption and accelerated training speed, yielding systems that can scale more effectively than the original BERT model. Notably, ALBERT has showcased superior performance compared to traditional DL models like LSTM and CNN [31], [32]. The architecture of ALBERT is very similar to BERT; it has a stack of Transformer encoder layers. At the same time, ALBERT does introduce a new paradigm in sharing parameters: instead of having distinct weights for each layer, it shares parameters across stacks of layers. This effectively keeps the overall number of parameters much lower compared to BERT [33], [34]. In addition, ALBERT has also introduced the "cross-layer parameter sharing" mechanism, which enables various levels to share parameters, refining the concept even more. This new approach turns ALBERT into a more effective and efficient application in handling multi-sets of tasks related to NLP. Functionally, ALBERT works quite similarly to the BERT model: several transformer layers process the input text, while a feed-forward neural network produces an output as the final result of the task at hand. In training, the ALBERT model minimizes some kind of loss function that calculates the difference between the predicted output and the real label [35].

2.2.5 Embedding

The word embedding framework is a model that learns word representations from an unannotated dataset in an expensive and time-consuming way. The goal is to come up with vectors for words carrying semantic information. One potential problem of word embeddings, however, is that these word embeddings reflect and amplify social prejudices baked into the training data. For example, word embeddings may encode capturing certain gender or racial stereotypes if this was present in the data and can be seen as furthering bias in some applications [36]. The text is first tokenized, breaking it down into input tokens, which are then mapped to word embeddings using an embeddings layer. The output matrices are categorized into tokens, and these output tokens are decoded back into text. To ensure that the model can understand the sequence's order, "positional encodings" are added to the input embeddings at the base of the encoder and decoder stacks. The embeddings for input and output tokens share the same dimension, enabling efficient and accurate text decoding. For this reason, utilize the sine and cosine functions in the following form:

$$Pe(Pos, 2i) = \sin\left(\frac{Pos}{10000^{\frac{2i}{d_{model}}}}\right)$$
$$Pe(Pos, 2i+1) = \cos\left(\frac{Pos}{10000^{\frac{2i}{d_{model}}}}\right) \quad (12)$$

Pos indicates the position, while i represents the dimensions. As a result, the positional encoding's dimensions separately correspond to a sinusoid [37].

2.2.6 Tokenization

Tokenizers are fundamental in NLP as they determine where token boundaries should be placed in text.

Typically, adjacent letters form a single token, while non-letter characters, such as punctuation, create boundaries between tokens. However, when a letter is followed by a non-letter character, tokenization rules become more complex, often requiring linguistic knowledge for accurate segmentation. The tokenizer should make an educated guess, taking into account the linguistic context, like negation handling, punctuation handling, and adherence to the rules of a particular language. Quite often, different languages are tokenized in different ways. There are languages, such as English, that have certain whitespace characters that naturally separate words. Contrasts include languages like Chinese or Thai that do not use spaces between words, hence needing special techniques for tokenization. Given all these differences, tokenizers try to segment text into meaningful units as best as possible, since such units form the heart of further processing in most natural language understanding tasks [38]. Tokenization is performed using the Albert-base-v2 tokenizer, a subword-based method to handle out-of-vocabulary words. Subword-level tokenization of code-mixed sentences makes sure that multilingual text is not distorted. Special cases, such as contractions (e.g., 'don't' → ['don', "'", 't']) and ambiguous punctuation (e.g., URLs or hashtags), are retained except where they interfered with sentence structure. In addition, stopword filtering is applied selectively so as not to lose contextually important words.

2.3 Performance assessment of the classification model

The key metrics used to evaluate the performance of categorization models in the current study are stated below:

- Accuracy is a critical parameter for determining a model's overall accuracy, as it indicates its capacity to create more exact estimations across all categories.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (13)$$

Here, TP represents True Positive, FN denotes False Negative, FP stands for False Positive, and TN signifies True Negative.

- Precision for classifications is defined as the proportion of positive cases detected, to reduce false positives and confirm the classifier's predictions.

$$Precision = \frac{TP}{TP + FP} \quad (14)$$

- Recall, or sensitivity, is a statistic that assesses a classifier's ability to correctly detect positive events while avoiding false negatives.

$$Recall = \frac{TP}{TP + FN} \quad (15)$$

- The F1-Score measures a classifier's ability to recognize positive occurrences in unequal class distributions by adding up accuracy and recall to get a single score.

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (16)$$

3 Results

In this work, the researchers attempted to differentiate between fake and real news by comparing five different models: the architecture 2-CNN 2-Bi-LSTM, 3-CNN 1-Bi-LSTM, 1-CNN 3-Bi-LSTM, and the Albert-base-v2 and Distilbert-base-uncased transformer models. Several performance metrics were considered in this research, including accuracy, precision, recall, F1 score, and precision/recall curve. Besides, a few techniques were performed to enhance the performance and avoid overfitting: batch normalization and tokenization of text, labels to numerical categories, and padding/truncation of sequences. Batch normalization prevents problems in training and speeds it up since the distributions of input values are made stable. Text tokenization allows for the conversion of words into numerical tokens that can be further processed. It also changes labels to numerical categories for model training and allows padding/truncation of sequences, making the dimensionality uniform for optimizing model performance. Therefore, this comparative study of the various models introduced metrics that showed rich insights into their efficiencies and robustness in detecting fake news. A comparative case study like this will make informed decisions easier to take regarding model selection and refinement, thereby helping in news identification and fighting misinformation at social media sites. The recall and precision curve gives the capability for evaluation in terms of how much each class is effectively predicted while building multi-class classification models.

Precision/recall curves were used for each of the 12 classes in the dataset to evaluate the classification performance of five different models: 2-CNN 2-Bi-LSTM, 3-CNN 1-Bi-LSTM, 1-CNN 3-Bi-LSTM architectures, and Albert-base-v2 and distilBert-base-uncased transformer models. Fig. 7 depicts, for every class, the precision/recall curve in the model DistilBert-base-uncased. It is seen that the model performs fine in most classes but shows Classes 2-unreliable and Class 6-fake have lower precision and recall values. Fig. 8 shows the precision/recall curve of the 1-CNN 3-Bi-LSTM model for each class. Most classes have worse performance as compared to the Distilbert-base-uncased model, which ascertains the limitation in the modeling capability of the model concerning the proper classification of certain classes. Similarly, Figure 9 shows the precision/recall curves for the 2-CNN 2-Bi-LSTM model. Compared to the DistilBert-base-uncased model, the performance of this model is poor, especially when considering classes whose precision and recall values are low. In contrast, the 3-CNN 1-Bi-LSTM model demonstrates improved performance, as shown in Fig. 10. While not significantly better than the DistilBert-base-uncased model, it outperforms the 2-CNN 2-Bi-LSTM and 1-CNN 3-Bi-LSTM models for most classes. Fig. 11 depicts the precision/recall curve for each class in the Albert-base-v2 model. Despite lower precision in class 3, the overall performance of the model is better than that of the other models, particularly for the entire class set.

Overall, the precision/recall curves provide detailed insights into the classification performance of each model for the 12 classes, highlighting strengths and weaknesses that can guide further model refinement and selection.

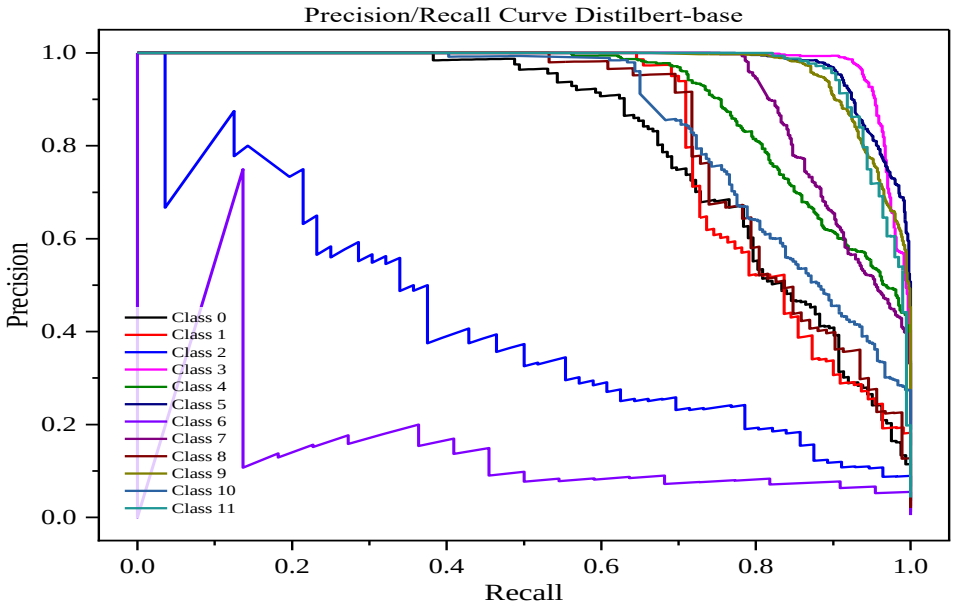

Figure 7: The precision/recall curve is distilbert-base-uncased for every class

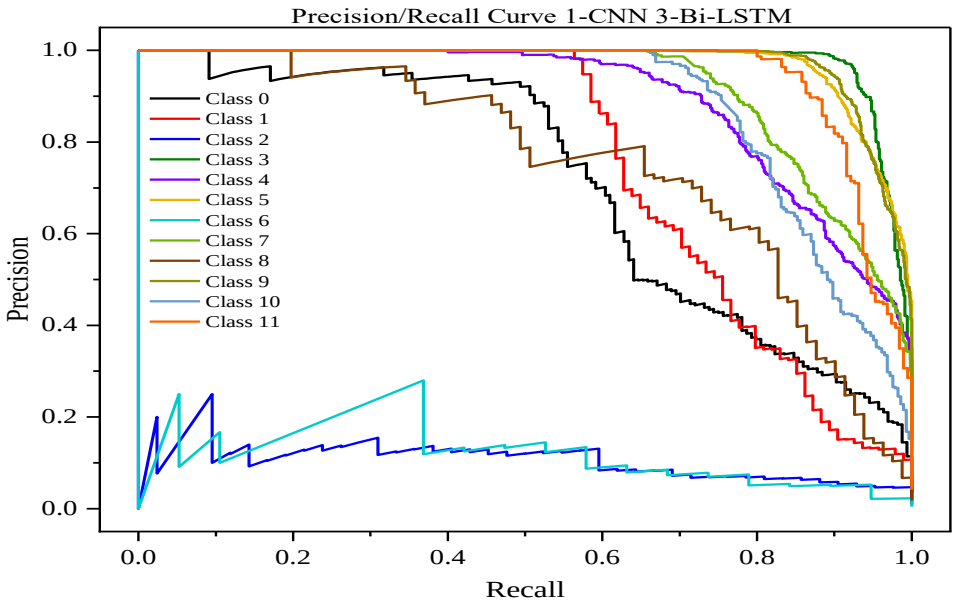

Figure 8: The precision/recall curve 1-CNN 3-Bi-LSTM of every class

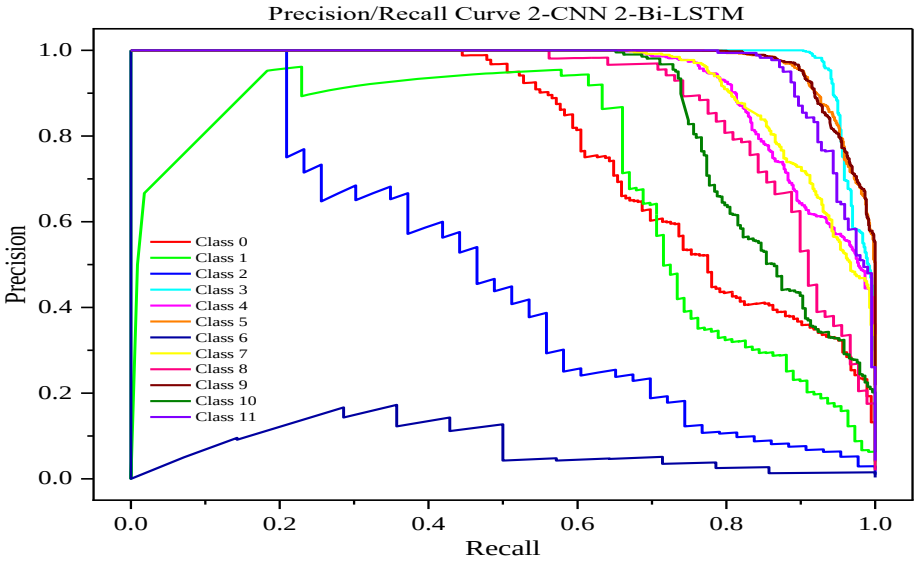

Figure 9: The precision/recall curve 2-CNN 2-Bi-LSTM of every class

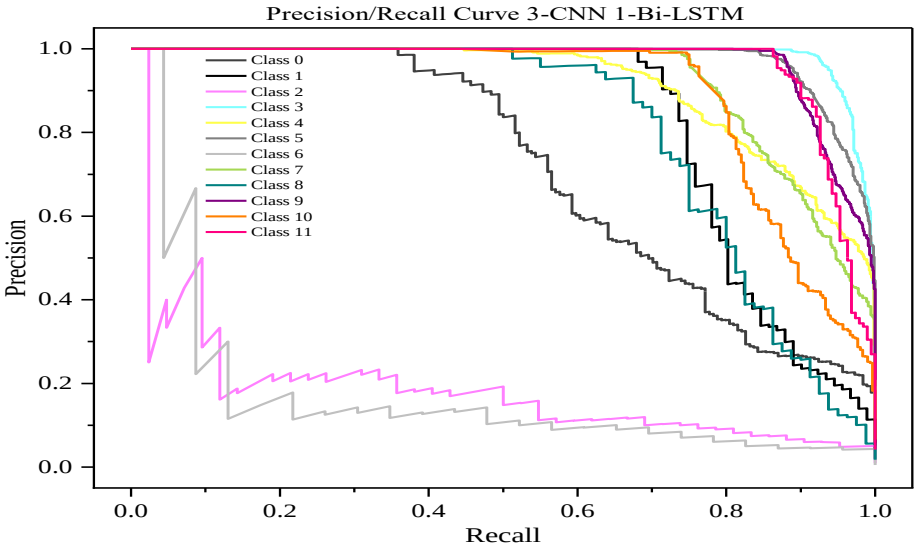

Figure 10: The precision/recall curve 3-CNN 1-Bi-LSTM of every class

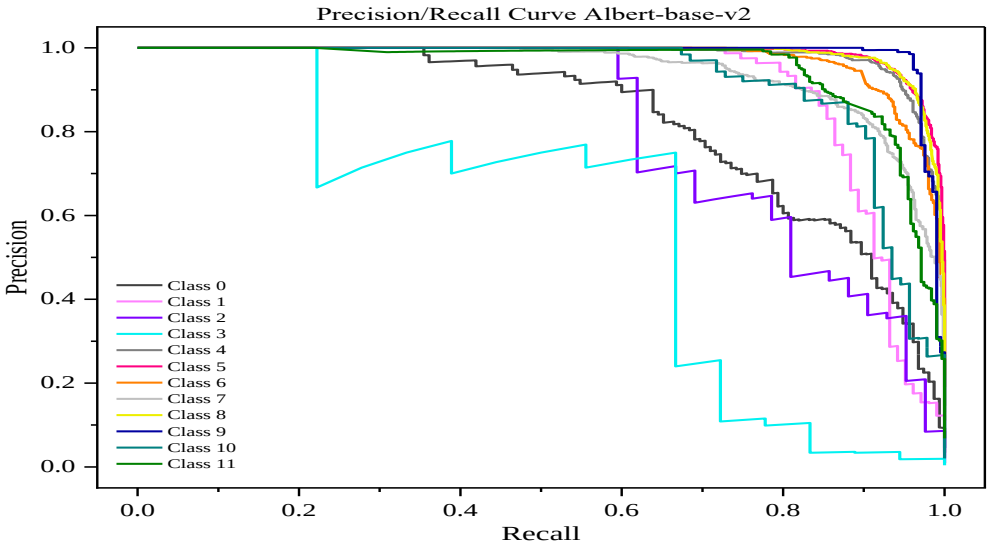

Figure 11: The precision/recall curve Albert-base-v2 of every class

Albert-base-v2 classification performance was also assessed using per-class evaluation metrics, as indicated in Fig. 12. The result indicates that the model achieves high precision, recall, and F1-scores in all classes except one, leading to 91% overall accuracy. The weighted F1-score of 0.92 indicates the strength of the model in fake news classification. Interestingly, a low recall of 0.50 among certain minority classes, i.e., Class 11, maybe a sign of dataset imbalance. Albert-base-v2's better-classifying abilities compared to traditional CNN-BiLSTM and DistilBERT models are supported by the macro-average F1-score value of 0.88, confirming its effectiveness.

	precision	recall	f1-score	support
0	0.91	0.93	0.92	2061
1	0.94	0.91	0.92	2810
2	0.84	0.97	0.90	4135
3	0.98	0.95	0.96	2227
4	1.00	0.91	0.95	4748
5	0.95	0.93	0.92	320
6	0.95	0.89	0.92	802
7	1.00	0.93	0.94	1185
8	0.84	0.84	0.84	954
9	0.92	0.76	0.83	310
10	0.77	0.89	0.83	603
11	0.73	0.50	0.59	165
accuracy			0.91	20320
macro avg	0.90	0.87	0.88	20320
weighted avg	0.93	0.92	0.92	20320

Figure 12: Per-class performance metrics for albert-base-v2

The classifying discriminative capacity of the Albert-base-v2 model was also examined based on ROC curves, presented in Fig. 13. The ROC curve offers a comprehensive assessment of the model's performance in differentiating between real and false news across different classes. The area under the curve (AUC) measurements of most classes were close to 1.00, which suggests outstanding discriminatory effectiveness. The findings confirm that Albert-base-v2 successfully reduces the incidence of false positives without sacrificing high true positive rates, thereby exhibiting its advanced capability in dealing with intricate fake news classification issues. Additionally, the high AUC scores for all classes further strengthen the model's superiority in comparison to CNN-BiLSTM and DistilBERT-based models.

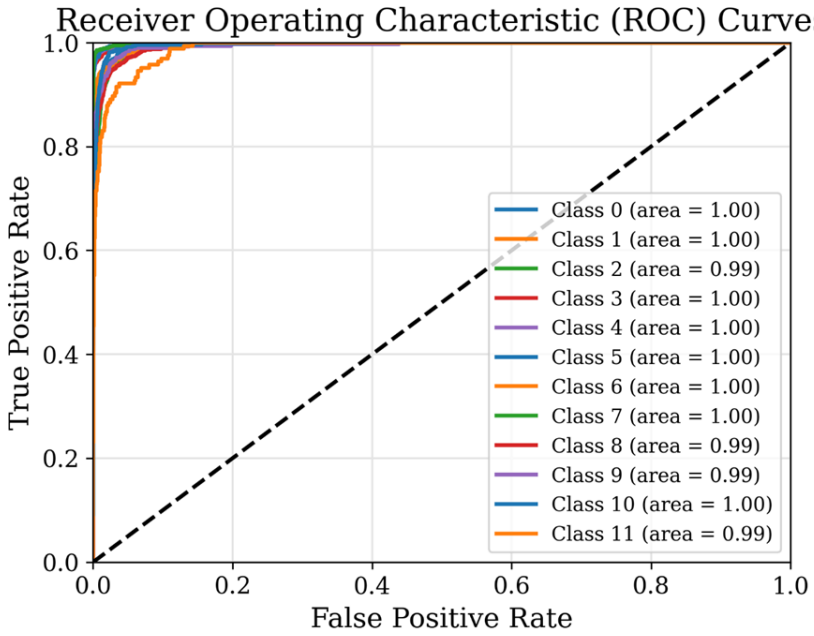

Figure 13: Receiver operating characteristic (ROC) Curves for albert-base-v2

Fig. 14 and Table 3provide a detailed comparison of the performance metrics for the 2-CNN 2-Bi-LSTM, 3-CNN 1-Bi-LSTM, 1-CNN 3-Bi-LSTM, Distilbert-base-uncased, and Albert-base-v2 models. The evaluation metrics considered in this comparison include accuracy, precision, recall, and F1-score. Analyzing the results reveals several key insights. The 1-CNN 3-Bi-LSTM model demonstrates lower performance compared to the other models in detecting certain classes, although it still shows reasonable overall performance. The 3-CNN 1-Bi-LSTM model exhibits a slight performance improvement over the 1-CNN 3-Bi-LSTM model. The 2-CNN 2-Bi-LSTM model achieves an F1-score of 0.8612 across all metrics, indicating a balanced performance across all classes. The Albert-base-v2 model outperforms all other models, achieving an F1-score of 0.9080 (Table 3).

This superior performance underscores the effectiveness of the Albert-base-v2 model in accurately classifying the Fake News Sample dataset. It demonstrates robust performance across all metrics, indicating its suitability for handling a variety of news detection tasks. In summary, the results show that the Albert-base-v2 transformer model is more effective than both the CNN and Bi-LSTM-based models, as well as the Distilbert-base-uncased model, in accurately classifying the Fake News Sample dataset. This highlights the efficacy of transformer models in handling news detection tasks, particularly when dealing with complex datasets containing multiple classes.

One of the primary concerns in model selection was a trade-off between classification accuracy and computational cost. As illustrated in Table 2, CNN-BiLSTM models are lighter computationally, requiring less resources for prediction and training, but are worse performing than transformers. DistilBERT is a cost-effective alternative, offering 40% fewer parameters than BERT but still an F1-score of 0.85. Albert-base-v2, being more computationally demanding than DistilBERT, achieves better classification accuracy (F1 = 0.908) with a good trade-off in efficiency through its parameter-sharing strategy. This underscores the need to balance accuracy needs with computational resources available in choosing models for detecting fake news.

Table 2: Performance vs. computational cost trade-offs

Model	Accuracy	F1-Score	Training Time (per epoch)	Model Size (Params)
2-CNN 2-BiLSTM	85.2%	0.854	12 min	2.1M
3-CNN 1-BiLSTM	85.8%	0.857	14 min	2.4M
1-CNN 3-BiLSTM	86.1%	0.861	16 min	2.8M
DistilBERT-base	85.0%	0.850	32 min	66M

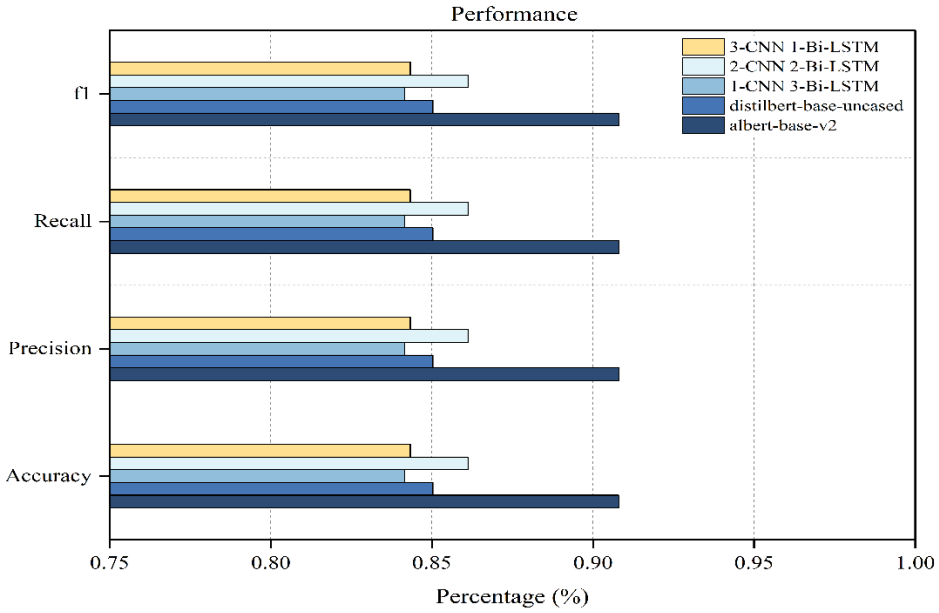

Figure 14: The performance obtained from albert-base-v2, 2-CNN 2-Bi-LSTM, disitilbert-base-uncased, 3-CNN 1-Bi-LSTM, and 1-CNN 3-Bi-LSTM classification models

Table 3: The performance value obtained from albert-base-v2, 2-CNN 2-Bi-LSTM, disitilbert-base-uncased, 3-CNN 1-Bi-LSTM, and 1-CNN 3-Bi-LSTM classification models

	Albert-base-v2	2-CNN 2-Bi-LSTM	Distilbert-base-uncased	3-CNN 1-Bi-LSTM	1-CNN 3-Bi-LSTM
accuracy	0.9080	0.8612	0.8503	0.8433	0.8416
precision	0.9080	0.8612	0.8503	0.8433	0.8416
recall	0.9080	0.8612	0.8503	0.8433	0.8416
f1	0.9080	0.8612	0.8503	0.8433	0.8416

Table 4 presents the results of t-tests conducted to compare the performance of the Albert model with several other models. The objective of this analysis was to assess the statistical significance of performance differences. As the Albert model has been identified as the superior model in this study, pair-wise comparisons were performed between Albert and each of the other models (including 2-CNN 2-BiLSTM, DistilBERT, 3-CNN 1-BiLSTM, and 1-CNN 3-BiLSTM). The P-value and T-Statistic values are provided for each comparison, respectively indicating the probability of observing the obtained results under the null hypothesis (no significant difference) and the magnitude of the difference between the groups. The very small P-values (less than 0.05) suggest the presence of statistically significant differences between the performance of the Albert model and the other compared models.

Table 4: T-test Results for statistical significance in model performance comparisons

Comparison	T-Statistic	P-value
Albert vs 2-CNN 2-BiLSTM	7.13	0.0017
Albert vs DistilBERT	8.21	0.001
Albert vs 3-CNN 1-BiLSTM	9.45	0.0006
Albert vs 1-CNN 3-BiLSTM	10.12	0.0004

4 Discussion

The performance outcome based on this work is compared with those of other works, which are tabulated in Table 3. The Albert-base-v2 transformer model proposed was more efficient than other deep learning architectures, with an F1-score of 0.908 and an accuracy rate of 90.8%. The findings suggest a remarkable improvement over CNN-BiLSTM-based architectures as well as other studies using conventional deep learning techniques.

4.1 Comparison with related works

Several reasons were accountable for Albert-base-v2's superior performance over previous methods:

Transformer-based advantages: Unlike CNN and BiLSTM-based models, Albert-base-v2 benefits from parameter-sharing mechanisms and deeper contextualization, which improves fake news classification performance. While Albert-base-v2 has good classification performance, its black-boxing is concerning when it comes to transparency of decision-making. Fake news detection models must not just be very accurate but also explainable so that trust and accountability can be ensured. If not interpretable, misclassification without cause is feasible, and this can lead to content moderation bias or censorship of factual information.

Dataset features: The Fake News Sample-Pontes dataset utilized in this research comprises a variety of news sources, thereby enhancing the generalization capability of the model compared to those utilized in earlier research works. For instance, the BerConvoNet [7] and TI-CNN [10] models were trained on domain-specific datasets, compromising their generalization capability.

Computational Efficiency: While being lightweight, the DistilBERT architecture had slightly reduced accuracy (85.0%) than Albert-base-v2 (90.8%), indicating the natural trade-off between computational expense and classification accuracy.

4.2 Analysis of performance differences

Performance differences across various architectures can be explained by:

Model complexity: While CNN-BiLSTM models had reasonable performance, their sequential nature hindered their capacity to effectively capture long-term dependencies in comparison with transformers.

Feature extraction techniques: The TI-CNN model [10] leveraged both text and image data, with impressive performance in multimodal settings. Its computational expenses and training intricacies, however, presented difficulties in comparison to Albert-based methods.

Generalization capacity: Conventional machine learning algorithms [11] depended on linguistic analysis but were inflexible when it came to generalizing across different news styles. The transformer-based models considered in this research demonstrated enhanced robustness across news types.

4.3 Comparison with state-of-the-art models

In contrast to existing state-of-the-art (SOTA) models, the present work contributes in the following ways:

Better interpretability: Most SOTA deep learning models are black boxes, making it challenging to explain their decision-making. By leveraging class-wise confusion matrices and precision-recall curves, our method renders model output more interpretable.

Balanced performance: A few recent state-of-the-art models, such as WSCH-CNN [6], have reported high classification accuracy at the expense of computationally expensive designs. However, our design preserves high performance while keeping the model's scalability intact.

Robust dataset preprocessing: In contrast to the prior work relying on manually designed features, the present method has utilized automated feature extraction, facilitating greater adaptability on various fake news datasets.

4.4 Limitations and future directions

Apart from the advantages of the suggested approach, it also suffers from the following limitations:

Real-world application constraints: Even though the model is effective at processing benchmark datasets, its applicability in real-time fake news detection is still to be broadly tackled. Analysis of the dataset revealed a skew toward shorter sentences and a strong concentration of articles from high-reputation domains such as 'nytimes.com.' While the dataset provides valuable insights into news classification, this skew may introduce bias in model predictions. Shorter articles, which are more common in the dataset, may be classified with higher accuracy than longer, more complex texts. Similarly, domain representation biases could lead to classification patterns that generalize poorly to news sources not included in the dataset.

Analysis of the dataset revealed a skew toward shorter sentences and a strong concentration of articles from high-reputation domains such as 'nytimes.com.' While the dataset provides valuable insights into news classification, this skew may introduce bias in model predictions. Shorter articles, which are more common in the dataset, may be classified with higher accuracy than longer, more complex texts. Similarly, domain representation biases could lead to classification patterns that generalize poorly to news sources not included in the dataset. Future work should explore more diverse, balanced datasets to mitigate these effects.

Future work should explore more diverse, balanced datasets to mitigate these effects.

Training bias: Even though dataset preprocessing techniques have been utilized, data bias sources in a dataset may still influence predictions. Debiased methods may be investigated in future work.

Challenges in explainability: While confusion matrices and feature visualization are part of the interpretative process, more work is needed to include explainable AI (XAI) methods to enhance model transparency.

5 Implications and future directions

The increasing reliance on AI-driven fake news detection is an ethical issue because of bias in model decision-making and dataset labeling. Fake news datasets are often manually labeled and thus prone to interpretation that may be attuned to political, cultural, or ideological leanings. Annotators' worldviews can bias decisions and create imbalanced training data that undermines model fairness. Moreover, deep learning models like Albert-base-v2 are trained on data distributions, and thus any inherent bias in the data set will be present in predictions as well.

While text-based fake news classification is the subject of this research, multi-modal misinformation (text + images + videos) is a new threat. Future research will require the inclusion of multi-modal approaches, combining text-based transformers (Albert-base-v2) with ViTs for image processing and GNNs for network-based misinformation dynamics. Such multimodal fusion can potentially enhance detection performance, particularly for social media-based fake news, since textual and visual information are often manipulated together.

Biased training data can make models favor certain narratives systemically, leading to unwanted biases in real-world classification. The inherent black-box property of transformer models renders the explanations for the flagging of news articles difficult to interpret, highlighting the need for explainability in the direction of transparency and fairness. Furthermore, algorithmic detection mechanisms can inadvertently suppress genuine content through false positives, thereby further exacerbating censorship concerns, especially in politically charged contexts. To counter these potential risks, there is a need to integrate fairness assessment techniques, implement explainability techniques like SHAP (SHapley Additive Explanations) and LIME (Local Interpretable Model-agnostic Explanations) to interpret the decision-making steps, and have human-in-the-loop validation to ensure the ethical use of artificial intelligence.

While this study focuses on a comparative analysis of established NLP models for fake news detection, future research could explore the integration of multi-modal features such as visual content and social context to enrich model understanding. To address interpretability concerns, future work needs to include Explainable AI (XAI) methods to make model decision-making processes more transparent. Techniques such as SHAP and LIME can be used to identify which words or phrases influence classification decisions. Further, attention visualizations in transformer models can highlight significant contextual patterns used in fake news classification, enhancing transparency and trust in automated detection systems.

Notwithstanding these ethical concerns, the suggested approach has immense practical applications. AI-driven fake news detection can be used in social media networks like Twitter, Facebook, and YouTube to aid fact-checking and limit the propagation of disinformation. These models can be utilized to facilitate source validation to ensure journalistic integrity, while disinformation mitigation and corporate reputation management can be enhanced in both government and corporate environments. There are some limitations, however, specifically regarding the ability to adapt to the dynamic nature of misinformation. Since fake news patterns evolve with time, models trained on previous data may be unable to recognize new deception tactics, and hence continuous retraining with new datasets is required. One challenge of particular concern is the detection of misinformation that has been generated by artificial intelligence. A study by Moalla et al. (2023) suggests that it is increasingly difficult to differentiate between AI-generated fake news and human-written articles. As synthetic misinformation has not been explicitly addressed in the current approach, it is essential to seek further investigation in this area. Moreover, model interpretability is still a limitation since deep learning models are black boxes, thus making decisions hard to justify in high-stakes settings. There may also be legal and regulatory challenges, especially when AI-driven fake news detection is applied to automated content moderation, which may create tensions in terms of censorship and freedom of speech.

To address these challenges, upcoming research must be directed towards creating models that can identify artificial intelligence-generated misinformation, integrating explainability methods to enhance transparency, and increasing detection on multiple languages and online platforms. Misinformation tracing studies on different platforms must also receive priority consideration to detect coordinated disinformation campaigns, while advancements in real-time deployment strategy must be considered for increasing practical usage. By overcoming these challenges, AI-based fake news detection can be made more efficient and ethical so that serious misinformation is responsibly countered in a world that is becoming more digital.

6 Conclusion

The research work presented a wide review concerning the different models of neural networks for detecting fake news based on the metrics of accuracy, precision, recall, and F1-score. In this work, the project used Fake News Sample databases to properly identify news. It discussed the challenges and capacities of different models: 2-CNN 2-Bi-LSTM, 3-CNN 1-Bi-LSTM, and 1 CNN 3-Bi-LSTM, other than Albert-base-v2 and Distilbert-base-uncased transformer models. This rigorous method returned handy information on model performance features.

The testing results proved that Albert-base-v2 outperformed the transformer model 2-CNN 2-Bi-LSTM, 3-CNN 1-Bi-LSTM models, and Distilbert-base-uncased transformer model. Specifically, Albert-base-v2 had the best F1-score of 0.908, showing that the model is robust in classifying fake news articles. Though there could be some variation in the class performances, the overall performance turned out to be very strong as Albert-base-v2 handled the complication in the Fake News Sample dataset quite well.

These further emphasize the reasons for using transformer-based models, such as Albert-base-v2, inefficiently and correctly detecting fake news. The results of this study provide valuable contributions to the field of news identification by underlining the importance of advanced neural network models for better classification accuracy and, subsequently, combating misinformation in news articles. Extensive assessment criteria, such as accuracy, precision, recall, F1 score, and AUC-PR analysis, have been used in the paper for comparative performance evaluation. This helped participants in model selection and refining decisions that best suited their personality prediction and pattern recognition. The work contributes to knowledge by offering an extensive

comparison between various models and their applicability in identifying news.

Abbreviation

Nomenclature		Greek letters	
Abbreviations		Latin Symbols	
NLP	Natural Language Processing	z_i	the input to the convolution layer
NEB	News Embedding Block	C_i	a reference point
MSFB	Multi-Scale Feature Block	×	the connection between the matrix and the input
CNN	Convolutional Neural Network	$F(.)$	the activation function
AC-BiLSTM	attention-based convolutional bidirectional long short-term memory	b_i	the bias vector
TM	Tsetlin Machine	φ	the max-pooling layer's final result
WSCH-CNN	Web Scraping Content Heading CNN	$\Delta(.)$	the max-pooling splitting function
MPFN	Multimodal Progressive Fusion Network	y_i	the final output vector
LSTM	Long Short-Term Memory	τ_i	the value of the weight matrix
BERT	Bidirectional Encoder Representations from Transformers	γ_i	the bias
TP	True Positive	σ	the cell model includes a sigmoid function
FN	False Negative	x_i	the current input
FP	False Positive	h_{i-1}	the previous state
TN	True Negative	c_{i-1}	the current cell state
XAI	Explainable AI	x_t	word vector
SHAP	SHapley Additive Explanations	I_t, F_t, and O_t	input, forget, and output gate vectors
LIME	Local Interpretable Model-agnostic Explanations	W and b	cell parameters
LRP	Layer-wise Relevance Propagation	$\overrightarrow{h_t}$ and $\overleftarrow{h_t}$	the outputs of the forward and backward passes
		Pos	the position encoding

Competing interests

The authors declare no competing interests.

Authorship contribution statement

Chi Zhang: Writing-Original draft preparation, Conceptualization, Supervision, Project administration.

Conflicts of Interest

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

Author Statement

All the authors have read and approved the manuscript. As stated earlier in this document, the requirements for authorship have been met, and each author believes that the manuscript represents honest work.

Ethical Approval

All authors have been personally and actively involved in substantial work leading to the paper and will take public responsibility for its content.

References

[1] A. Hassan and H. Rashid, "Estimation of Ultimate Bearing Capacity in rock-socketed piles using optimized machine learning approaches," *Advances in Engineering and Intelligence Systems*, vol. 2, no. 04, pp. 114–127, 2023.

[2] S. Mastan Val, "Improving COPD Readmission Prediction with Optimized Machine Learning," *Journal of Artificial Intelligence and System Modelling*, vol. 1, no. 03, pp. 86–103, 2024.

[3] T. E. Trueman, A. Kumar, P. Narayanasamy, and J. Vidya, "Attention-based C-BiLSTM for fake

news detection," *Appl Soft Comput*, vol. 110, p. 107600, 2021.

[4] M. J. Awan *et al.*, "Fake news data exploration and analytics," *Electronics (Basel)*, vol. 10, no. 19, p. 2326, 2021.

[5] Y. Yang, L. Zheng, J. Zhang, Q. Cui, Z. Li, and P. S. Yu, "TI-CNN: Convolutional neural networks for fake news detection," *arXiv preprint arXiv:1806.00749*, 2018.

[6] D. K. Vishwakarma, P. Meel, A. Yadav, and K. Singh, "A framework of fake news detection on web platform using ConvNet," *Soc Netw Anal Min*, vol. 13, no. 1, p. 24, 2023.

[7] M. Choudhary, S. S. Chouhan, E. S. Pilli, and S. K. Vipparthi, "BerConvoNet: A deep learning framework for fake news classification," *Appl Soft Comput*, vol. 110, p. 107614, 2021.

[8] P. Meel and D. K. Vishwakarma, "A temporal ensembling based semi-supervised ConvNet for the detection of fake news articles," *Expert Syst Appl*, vol. 177, no. February, p. 115002, 2021, doi: 10.1016/j.eswa.2021.115002.

[9] V. Gunasekaran and D. Karthikeyan, "Real Time Fake news Detection Web App Enhanced by Machine Learning Algorithms," in *2023 4th International Conference on Intelligent Technologies (CONIT)*, IEEE, 2024, pp. 1–7.

[10] Y. Yang, L. Zheng, J. Zhang, Q. Cui, Z. Li, and P. S. Yu, "TI-CNN: Convolutional neural networks for fake news detection," *arXiv preprint arXiv:1806.00749*, 2018.

[11] J. Clarke, H. Chen, D. Du, and Y. J. Hu, "Fake news, investor attention, and market reaction," *Information Systems Research*, vol. 32, no. 1, pp. 35–52, 2020.

[12] P. Meel and D. K. Vishwakarma, "Fake news detection using semi-supervised graph convolutional network," *arXiv preprint arXiv:2109.13476*, 2021.

[13] B. Bhattarai, O.-C. Granmo, and L. Jiao, "Explainable tsetlin machine framework for fake news detection with credibility score assessment," *arXiv preprint arXiv:2105.09114*, 2021.

[14] O.-C. Granmo and L. Jiao, "Explainable Tsetlin Machine Framework for Fake News Detection with Credibility Score Assessment," 2022.

[15] M. J. Awan *et al.*, "Fake news data exploration and analytics," *Electronics (Basel)*, vol. 10, no. 19, p. 2326, 2021.

[16] T. E. Trueman, A. Kumar, P. Narayanasamy, and J. Vidya, "Attention-based C-BiLSTM for fake news detection," *Appl Soft Comput*, vol. 110, p. 107600, 2021.

[17] D. K. Vishwakarma, P. Meel, A. Yadav, and K. Singh, "A framework of fake news detection on web platform using ConvNet," *Soc Netw Anal Min*, vol. 13, no. 1, p. 24, 2023.

[18] N. Põldvere, Z. Uddin, and A. Thomas, "The PolitiFact-Oslo Corpus: A new dataset for fake news analysis and detection," *Information*, vol. 14, no. 12, p. 627, 2023.

[19] J. Jing, H. Wu, J. Sun, X. Fang, and H. Zhang, "Multimodal fake news detection via progressive fusion networks," *Inf Process Manag*, vol. 60, no. 1, p. 103120, 2023.

[20] H. Moalla, H. Abid, D. Sallami, E. Aïmeur, and B. Ben Hamed, "Exploring the Power of Dual Deep Learning for Fake News Detection," *Informatica*, vol. 48, no. 4, 2025.

[21] P. Meel and D. K. Vishwakarma, "A temporal ensembling based semi-supervised ConvNet for the detection of fake news articles," *Expert Syst Appl*, vol. 177, no. April, p. 115002, 2021, doi: 10.1016/j.eswa.2021.115002.

[22] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," pp. 2–6, 2019.

[23] B. Büyüköz, A. Hürriyetoğlu, and A. Özgür, "Analyzing ELMo and DistilBERT on Socio-political News Classification," *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, no. May, pp. 9–18, 2020.

[24] M. Abadeer, "Assessment of DistilBERT performance on Named Entity Recognition task for the detection of Protected Health Information and medical concepts," no. 2019, pp. 158–167, 2020, doi: 10.18653/v1/2020.clinicalnlp-1.18.

[25] Y. Chen, J. Bin, and C. Kang, "Smart Agricultural Technology Application of machine vision and convolutional neural networks in discriminating tobacco leaf maturity on mobile devices," *Smart Agricultural Technology*, vol. 5, no. September, p. 100322, 2023, doi: 10.1016/j.atech.2023.100322.

[26] F. A. Alijoyo *et al.*, "Advanced hybrid CNN-Bi-LSTM model augmented with GA and FFO for enhanced cyclone intensity forecasting," *Alexandria Engineering Journal*, vol. 92, no. March, pp. 346–357, 2024, doi: 10.1016/j.aej.2024.02.062.

[27] J. Gu *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognit*, vol. 77, pp. 354–377, 2018, doi: 10.1016/j.patcog.2017.10.013.

[28] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[29] P. Zhou *et al.*, "Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification," pp. 207–212, 2016.

[30] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," pp. 1–17, 2019.

[31] H. Wang, X. Hu, and H. Zhang, "Sentiment analysis of commodity reviews based on ALBERT-LSTM," in *Journal of Physics Conference Series*, AA(Guilin University of Electronic Technology, China), AB(Guilin University of Electronic Technology, China), AC(Guilin University of Electronic Technology,

China), Nov. 2020, p. 12022. doi: 10.1088/1742-6596/1651/1/012022.

[32] X. Wang, H. Wang, G. Zhao, Z. Liu, and H. Wu, "Albert over match-lstm network for intelligent questions classification in chinese," *Agronomy*, vol. 11, no. 8, 2021, doi: 10.3390/agronomy11081530.

[33] C. H. Chiang, S. F. Huang, and H. Y. Lee, "Pretrained language model embryology: The birth of ALBERT," *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, no. Section 3, pp. 6813–6828, 2020, doi: 10.18653/v1/2020.emnlp-main.553.

[34] R. Zhao, B. Vogel, and T. Ahmed, "Adaptive Loss Scaling for Mixed Precision Training," pp. 1–11, 2019.

[35] S. Alrowili and K. Vijay-Shanker, "BioM-Transformers: Building Large Biomedical Language Models with BERT, ALBERT and ELECTRA," *Proceedings of the 20th Workshop on Biomedical Language Processing, BioNLP 2021*, pp. 221–227, 2021, doi: 10.18653/v1/2021.bionlp-1.24.

[36] M. Hort, R. Moussa, and F. Sarro, "Multi-objective search for gender-fair and semantically correct word embeddings," *Appl Soft Comput*, vol. 133, p. 109916, 2023, doi: 10.1016/j.asoc.2022.109916.

[37] R. Rao *et al.*, "MSA Transformer," *bioRxiv*, p. 2021.02.12.430858, 2021.

[38] J. Graën, M. Bertamini, M. Volk, M. Cieliebak, D. Tuggener, and F. Benites, "Cutter–a universal multilingual tokenizer," in *CEUR Workshop Proceedings*, CEUR-WS, 2018, pp. 75–81.