J. Electrical Systems 20-11s (2024): 1044-1060

Dr. M. Anusha[1]
R. Leelavathi[2*]

Hybrid Approach to Fake News Detection: Leveraging BERT-Based Model with Word Embedding Features for Sentiment Classification in Social Media Content

Abstract: The proliferation of fake news in social media necessitates robust and efficient methods for its detection. This research presents a novel approach to sentiment classification for fake news detection, focusing on non-sarcastic social media content. Leveraging the powerful contextual representation capabilities of BERT and enhancing its performance through the incorporation of word embedding features, our proposed model aims to discern the subtle nuances in sentiment that are indicative of deceptive information. The designed model is meticulously crafted to capture the intricate relationships between words and their contextual meanings within the given text. To assess its efficacy, a comparative analysis is conducted with two other models: a Long Short-Term Memory (LSTM) model, a widely used sequential model for natural language processing, and a BERT model without the integration of word embedding features. The experiments involve training and evaluating the models on a comprehensive dataset of non-sarcastic social media content. Performance metrics such as accuracy, precision, recall, and F1 score are employed the research findings indicate that the proposed BERT-based model with word embedding features outperforms both the LSTM model and the BERT model without word embedding features in terms of fake news detection accuracy.

Keywords: *Sentiment Classification, Fake News Detection, BERT, Word Embedding, LSTM, Accuracy*

1. INTRODUCTION

In the contemporary era of global social networking, ensuring the authenticity of content is imperative, demanding the implementation of sophisticated algorithms to maintain positive outcomes. This necessitates a meticulous selection process where only a subset of input words yields appropriate content as output. The inherent challenge lies in the high-risk nature of data collection and validation. This model integrates sequence modeling, word embedding, and sentiment analysis, incorporating a diverse dataset of social media posts, encompassing both genuine and deceptive information. To prepare the data, it is necessary to annotate the dataset by labeling each post as real or fake and undertake text preprocessing, including tokenization, stop word removal, and handling special characters and URLs. Pre-trained word embeddings such as Word2Vec, GloVe, or FastText are then employed to convert words into dense vector representations. Subsequently, recurrent neural networks (RNNs), long short-term memory networks (LSTMs), or transformer-based models like BERT are utilized to capture the sequence of words in each social media post, facilitating an understanding of contextual information and word relationships.

The output of the sequence modeling layer is further analyzed, extracting relevant features such as the final hidden state of an LSTM or the [CLS] token representation in BERT. Word embeddings are combined with sentiment features like sentiment scores and emotional tone as additional input features. A classification model, such as a deep neural network, is constructed to process these combined features and output binary classifications (real or fake).

The dataset is partitioned into training, validation, and test sets for model training and evaluation, utilizing techniques like cross-validation for hyperparameter tuning to avoid overfitting. Evaluation metrics such as accuracy, precision, recall, F1-score, and AUC-ROC are employed to assess model performance. Embedding algorithms, crucial for natural language processing (NLP), are selected based on project specifics, dataset size, and computational resources. To prevent overfitting, regularization techniques such as dropout are applied, and preprocessing steps are executed. Beyond conventional metrics, additional analyses such as confusion matrices and scrutiny of false positives/negatives are considered. Post-processing techniques, including threshold adjustments or ensemble methods, are implemented to refine model predictions.

The algorithm primarily focuses on identifying words with the highest tendency to distinguish between fake and real content. Fuzzy logic concepts and neural network protocols, specifically LSTM, BERT, and BERT with sentiment features, are analyzed. The collective analysis aims to identify the best-performing strategy that significantly impacts the model's performance. The trained model is deployed as part of a larger system for real-time or batch processing of social media content. Continuous monitoring and potential retraining with updated data ensure sustained effectiveness. Lastly,

[1]Assistant Professor, PG & Research Department of Computer Science, National College (Autonomous), Bharathidasan University, Tamil Nadu, India
(E-mail-anusha260505@gmail.com)
[2*]Ph.D Research Scholar, PG & Research Department of Computer Science, National College (Autonomous), Bharathidasan University, Tamil Nadu, India
(E-mail-leelavathi4794@gmail.com)
***Corresponding Author:** R. Leelavathi
*(E-mail-leelavathi4794@gmail.com)

a user-friendly interface or integration into a web application enhances accessibility for users seeking to verify the authenticity of social media content. Establishing a feedback loop for users to report false positives and negatives contributes to ongoing model refinement.

1.1 Problem Statement

The problem addressed by this research work is the pervasive issue of fake news within the global social network, particularly focusing on non-sarcastic social media content. The challenge lies in the necessity to develop effective algorithms for validating the authenticity of content and maintaining positive outcomes. With the increasing risk associated with collecting and validating data, there is a critical need for advanced techniques to discern genuine sentiments from potentially misleading information. The study aims to address this problem by proposing a BERT-based approach with word embedding features for sentiment classification, comparing its effectiveness with other models such as LSTM and BERT without word embedding features. The overarching problem is to enhance the accuracy of fake news detection in social media through the integration of sophisticated techniques that consider the contextual nuances and relationships within the text.

2. RELATED WORK

The researchers mainly concentrate on the discrimination of the fake data and true data. Thus, it's a versatile field emerges to be the most significant in the data manipulation for the growth of the new algorithm in the media field that evolves the betterment of avoiding the sarcasm which will affect the growing society in the field of research. Herewith majority of the research interest tries to hybrid new ideas with the previous existing one.

Ali, A.M. *et al.* (2022) [] introduces a novel approach to fake news detection through a Deep Ensemble Model employing sequential deep learning techniques. The proposed model leverages the power of ensemble learning to enhance accuracy and reliability in discerning deceptive information. Sequential deep learning methods, such as recurrent neural networks (RNNs) or long short-term memory networks (LSTMs), are utilized to capture intricate patterns within sequential data. The synergistic integration of these techniques results in a robust framework for effective fake news detection. Experimental evaluations demonstrate the model's superior performance compared to individual deep learning approaches, emphasizing its potential as a promising solution.

Alonso et al [] investigates the role of sentiment analysis in detecting fake news. Leveraging natural language processing techniques, the study focuses on analyzing the emotional tone and sentiment expressed in textual content to identify deceptive information. The proposed approach utilizes machine learning algorithms to discern patterns indicative of misinformation, contributing to the ongoing efforts to enhance the accuracy of fake news detection. Experimental results showcase the effectiveness of sentiment analysis as a valuable tool in the broader context of combating the spread of false information across various platforms.

Chaubey, P.K. *et al.* (2022) 'introduces a novel approach to sentiment analysis, focusing on the integration of image and text caption information using deep learning techniques. The proposed model employs advanced neural networks to capture complex emotional nuances present in multimedia content. Through experimental validation, the effectiveness of this multimodal approach is demonstrated, highlighting its potential applications in understanding sentiment in diverse and richly contextual multimedia data.

Chen, Q et al [] presents a robust hybrid deep learning model for forecasting stock prices. The proposed approach integrates an attention mechanism, multi-layer perceptron, and bidirectional long-short term memory (Bi-LSTM) neural network. This synergistic model is designed to capture intricate patterns in historical stock data, incorporating attention mechanisms for enhanced feature extraction. Through empirical validation, the effectiveness of the hybrid model is demonstrated, showcasing its potential as a powerful tool for accurate and dynamic stock price predictions.

Chen P., et al [] explores the application of Recurrent Attention Networks on Memory for Aspect Sentiment Analysis, as presented in the Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. The study investigates the effectiveness of recurrent attention mechanisms in capturing nuanced sentiments related to specific aspects in textual data. The findings contribute to advancing natural language processing techniques for aspect-based sentiment analysis, demonstrating the model's capabilities in discerning sentiment nuances tied to different aspects within the text.

Orestes Appel et al [] introduces a hybrid approach to sentiment analysis focused on the sentence level. The proposed method combines diverse techniques to address the nuances of sentiment within individual sentences. Through the integration of multiple approaches, including machine learning and natural language processing, the model demonstrates improved accuracy in capturing and interpreting sentiment information. The findings contribute to the advancement of sentiment analysis methodologies, particularly at the granular level of individual sentences.

Peng Chen.,et al [] presents a novel method for clause sentiment identification employing Convolutional Neural Networks (CNNs) with context embedding. The study explores the integration of context information to enhance the accuracy of sentiment identification at the clause level. Through experimentation, the proposed approach demonstrates effectiveness in capturing nuanced sentiment within clauses, contributing to advancements in natural computation techniques for sentiment analysis.

Table 1: Summary of Literature Review on comparison of the BERT and LSTM.

Paper Title and Year	**Approach**	**Key Findings**
Alonso, M.A. *et al.* (2021)	Sentimental Evolution of critical data	The system evolves the critical data that are involved in sentimental analysis.
Galli, A. *et al.* (2022)	Comprehensive algorithm for fake news detection	The framework has the betterment of the topology stepwise progress of the system nature in segregation of data
Chen, Q., Zhang, W. and Lou, Y. (2020)	RNN LSTM framework	RNN has optimistic network that has the novel reliability in the study of deep learning algorithm.
Dufraisse, E. *et al.* (2023)	Sentimental Analysis Evaluation	The sensitivity of text and their source information helps us to identify the how its possible to differentiate the true and fake information.
KhambraG., *et al* (2021)	Machine Learning methodology	Algorithm and specification of the model is trained by the means of machine learning also the comparison of each state models is positioned
Jain N., *et al* (2021)	Filtration and smoothening of data	Mistake and errors in data are being referred without errors.
Bojanowski P., *et al* (2017)	Word Segment Modulation	Perfect discrimination of the True and Fake data is patterned
Chen P., *et al* (2017)	Storage and Network	The word length and proper memory of the Input information is channelized
Chen P., Sun Z., Bing L., Yang W.,2017	Identify the sentiment analysis	Sentiment discriminates the False and True word
Orestes Appel.,et.al,2016	Hybrid methodology	Combining the existing with new topology to frame powerful hybrid
(Peng Chen.,et al.,2016	CNN algorithm	So, it identifies the crucial data and leave other data

The table 1 highlights various approaches and key findings in the domain of fake news detection and sentiment classification in social media content. Alonso et al. (2021) emphasize the evolution of critical data in sentimental analysis, focusing on its dynamic nature. Galli et al. (2022) introduce a comprehensive algorithm that improves data segregation through stepwise topology enhancements. Chen, Zhang, and Lou (2020) leverage an RNN-LSTM framework known for its novel reliability and optimism in deep learning applications. Dufraisse et al. (2023) evaluate sentimental analysis by emphasizing the sensitivity of text and source information, aiding differentiation between true and fake content. Khambra et al. (2021) use machine learning methodologies to train algorithms and compare state-of-the-art models. Jain et al. (2021) focus on data filtration and smoothening, ensuring minimal errors in the data. Bojanowski et al. (2017) present word segment modulation to achieve precise discrimination of true and fake data. Chen et al. (2017) optimize storage and network by channelizing input word length and memory. Another study by Chen et al. (2017) highlights the role of sentiment analysis in effectively distinguishing false and true words. Appel et al. (2016) propose a hybrid methodology by combining existing and new topologies for enhanced performance. Lastly, Peng Chen et al. (2016) adopt a CNN algorithm to identify crucial data while filtering out less relevant information. These findings collectively underscore the advancements in approaches to fake news detection, focusing on improved data handling, sentiment analysis, and hybrid methodologies.

3. PROPOSED METHODOLOGY

Figure 2 illustrates about the architecture of the proposed model. Each block given in the architecture has been explained in a subsequent sub section. The flowchart provided depicts the steps involved in sentiment analysis, first step is data preprocessing, data preprocessing—cleansing text, lowercasing, tokenization, and removing stop words. Word embedding techniques like GloVe or Word2Vec follow, capturing word meanings. Model building, selecting algorithms like LSTM or BERT, crafting architecture, and integrating embeddings, ensues. Predicting sentiment for new text samples leverages trained patterns. Evaluation measures model success. Accuracy, precision (positive prediction accuracy), recall (relevance identification), F1-score (precision-recall balance), and ROC-AUC (discrimination ability) gauge performance. This process holistically uncovers sentiments from text, showcasing the model's effectiveness in understanding and categorizing emotions. And fake news Dataset housing diverse news articles undergoes meticulous cleansing to remove extraneous elements like special characters, HTML tags, and irregularities. This crucial step aims to establish a consistent and manageable dataset, forming the foundation for subsequent phases. In the subsequent feature engineering phase, significant attributes are thoughtfully crafted, encompassing word frequencies, text length, and news source credibility. This process empowers the model with informative cues, shedding light on distinctive patterns linked to fake news. Progressing into model construction, a tailored machine learning model is created to classify news articles as genuine or counterfeit. Informed algorithm selection, meticulous architectural design, and alignment of engineered attributes with labels ensue. Trained on the dataset, the model becomes an adept predictor of news authenticity, seamlessly applying acquired insights to discern the articles' veracity. A pivotal aspect lies in deploying rigorous evaluation metrics, encompassing accuracy, confusion matrix, precision, recall, F1-score, and ROC-AUC. These measures collectively ensure the model's effectiveness in detecting fake news with precision. And next sentiment analysis entiment Analysis involves deciphering the emotional context of text through several key steps. Firstly, Data Preprocessing readies raw text by removing special characters and HTML tags to ensure data uniformity. Word Embedding follows, where techniques like GloVe or Word2Vec translate words into numerical vectors, capturing semantic nuances. The core of sentiment analysis is Model Building where an algorithm like LSTM or BERT is chosen, and the model's architecture is defined to identify patterns in the input data. After training, the model's Model Prediction capability assigns sentiment labels like positive or negative to new text, reflecting its "perception." Lastly, Model Building (Second Part) involves crafting evaluation metrics like accuracy, precision, recall, F1-score, and ROC-AUC to assess model performance comprehensively. In summary, sentiment analysis systematically deciphers emotions from text through meticulous preparation, transformation, modeling, prediction, and thorough evaluation. The phases involved in the proposed solution are Data preprocessing, Feature engineering, Modeling for Sentiment classification, and evaluation.

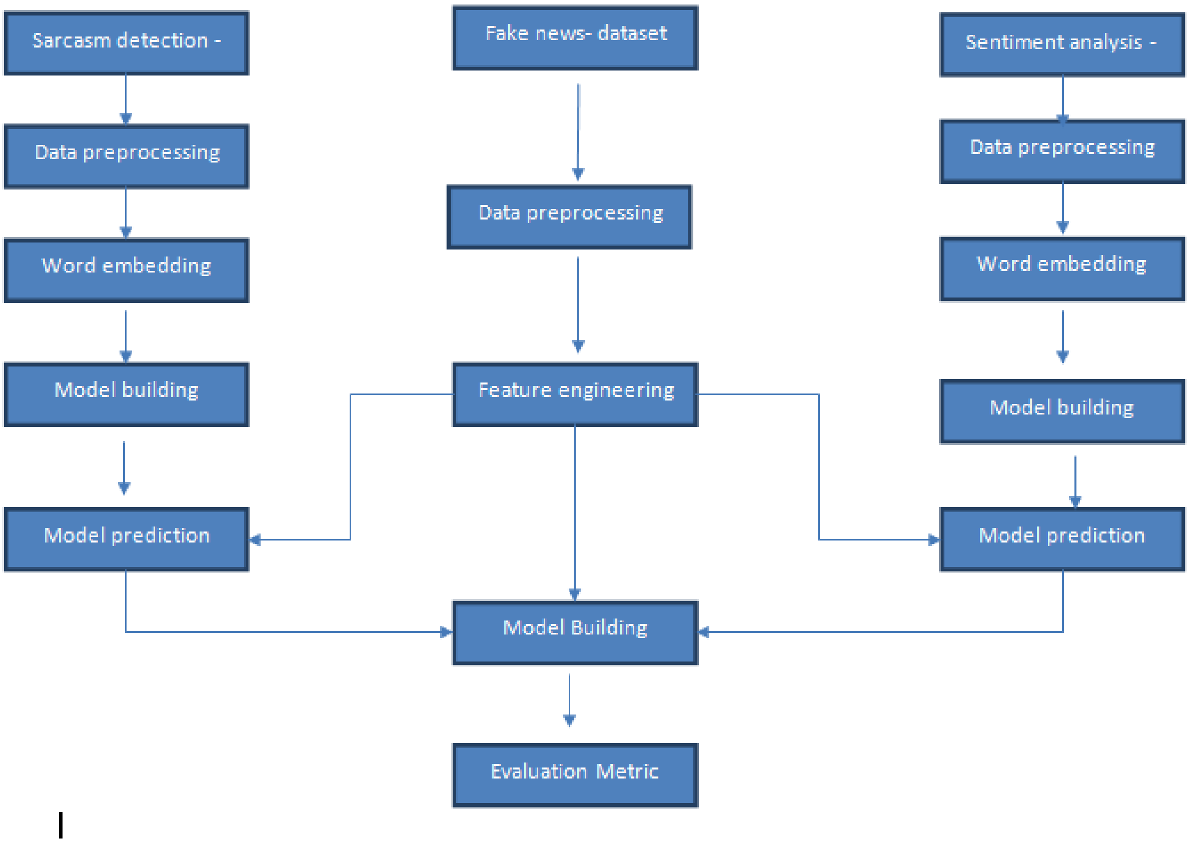

Fig 1. System Overview

3.1. Data Preprocessing

There are three preprocessing techniques has been utilized in this model such as cleansing text, lowercasing, tokenization, and removing stop words

3.1.1. Text cleansing

This process involves a series of techniques aimed at removing noise, irrelevant information, and inconsistencies from the text, thereby enhancing the model's ability to discern sentiment patterns. Common text cleansing steps include tokenization, stemming, and lemmatization to break down the text into its constituent parts, eliminating unnecessary variations in word forms. Stop word removal removes common words that do not contribute significant meaning to sentiment. Additionally, special character removal and handling of punctuation standardize the text for consistent analysis. URL removal and handling of user mentions or hashtags are essential for social media data.

3.1.2. Lowercasing

This technique helps ensure uniformity and consistency in the dataset by treating words with different cases as identical. Lowercasing prevents the duplication of words with different cases, enabling sentiment analysis models to recognize and associate sentiment-bearing terms more effectively. This standardization not only simplifies the computational process but also contributes to the overall accuracy and reliability of sentiment classification models, making them more adaptable to different styles of writing and linguistic variations present in textual data.

3.1.3. Tokenization

This process involves breaking down a given text into individual units, or tokens, which are usually words or sub words. Tokenization facilitates the analysis of text by converting it into a structured format that a machine learning model can comprehend. Each token represents a distinct unit of meaning, enabling the model to understand the relationships and nuances between words.

3.1.4. Removing stop words

Removing stop words is a pivotal preprocessing step in sentiment analysis that involves filtering out common words that do not contribute significant meaning to the overall sentiment of a text. Stop words, such as "and," "the," and "is," are ubiquitous and don't carry specific sentiment information. By eliminating these words, the focus shifts to more contextually relevant terms, enhancing the efficiency of sentiment classification models.

3.2. Word embedding

BERT relies on contextual embeddings, which means that the meaning of a word depends on its context within a sentence. Word embeddings contribute to the contextual understanding of words in a given sequence. This is particularly important in fake news detection, where the context of words and phrases can significantly impact the overall meaning of a statement.

3.3. Modelling

Baseline – LSTM baseline takes only 10% of the time to expand, however will get us ninety% of the way to acquire moderately desirable outcomes. Baselines help us put a more complex model into context in phrases of accuracy. This neural network architecture appears tailored for text analysis, potentially sentiment analysis or text classification. The embedding layer transforms words into 40-dimensional vectors, adaptable to variable-length sequences of up to 42 words. The dropout layer after embedding contributes no parameters but aids in regularization.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 42, 40)            1090240

 dropout (Dropout)           (None, 42, 40)            0

 lstm (LSTM)                 (None, 100)               56400

 dropout_1 (Dropout)         (None, 100)               0

 dense (Dense)               (None, 1)                 101

=================================================================
Total params: 1,146,741
Trainable params: 1,146,741
Non-trainable params: 0
_________________________________________________________________
None
```

Fig 2. Model Architecture Summary of an LSTM-Based Sentiment Analysis System

In figure ,it demonstrates about the neural network architecture appears tailored for text analysis, potentially sentiment analysis or text classification. The embedding layer transforms words into 40-dimensional vectors, adaptable to variable-length sequences of up to 42 words. The dropout layer after embedding contributes no parameters but aids in regularization. The subsequent LSTM layer processes sequences, generating 100-dimensional representations, capturing temporal patterns with 56,400 parameters. The final dense layer predicts outputs, possibly sentiment scores, using 101 parameters. A following dropout layer enhances robustness. With 1,146,741 trainable parameters, this setup adeptly learns patterns. Overall, it integrates embedding, LSTM processing, and dropout for effective text analysis in tasks like sentiment classification.

	title	true	scores	compound	pos	neg	type
0	Patrick Henningsen LIVE with guest Ray McGover...	0	{'neg': 0.173, 'neu': 0.827, 'pos': 0.0, 'comp...	-0.3182	0.00	0.173	NEG
1	U.S. senators set bipartisan bill to tighten s...	1	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...	0.0000	0.00	0.000	POS
2	U.S. prepares to open doors on billion-dollar ...	1	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound...	0.0000	0.00	0.000	POS
3	Should We Worry About McMaster as Trump's Nati...	0	{'neg': 0.218, 'neu': 0.602, 'pos': 0.18, 'com...	-0.1280	0.18	0.218	POS
4	Democrats urge FCC to drop plan to revise TV o...	1	{'neg': 0.174, 'neu': 0.826, 'pos': 0.0, 'comp...	-0.2732	0.00	0.174	NEG

Fig 3. Sentiment Analysis Results for Fake News Detection

The figure 3 presents an analysis of news headlines using sentiment analysis techniques, correlating sentiment scores with the authenticity of the news. The headline "Patrick Henningsen LIVE with guest Ray McGovern" is classified as fake (true=0), showing a negative sentiment score (neg=0.173), neutral sentiment (neu=0.827), and a compound score of -0.3182, resulting in a negative classification (NEG). Similarly, the headline "Should We Worry About McMaster as Trump's National Security Adviser?" is also classified as fake, with a higher negative sentiment score (neg=0.218), a compound score of -0.128, and a mixed sentiment reflected in the positive score (pos=0.18).

In contrast, the headlines "U.S. senators set bipartisan bill to tighten sanctions on North Korea" and "U.S. prepares to open doors on billion-dollar embassy in London" are classified as real (true=1) and have neutral sentiments (neu=1.0), with compound scores of 0.0, indicating no strong positive or negative polarity. Lastly, the headline "Democrats urge FCC to drop plan to revise TV ownership rules" is classified as fake, with a negative sentiment score (neg=0.174), a neutral sentiment (neu=0.826), and a compound score of -0.2732, resulting in a negative classification (NEG). This analysis provides insights into sentiment features like polarity and neutrality, aiding in differentiating fake and real news headlines.

Deep learning models, particularly recurrent neural networks (RNNs) and transformers, have demonstrated unparalleled prowess in understanding complex patterns within sequential data such as text. BERT may be used for sentiment evaluation protocols where understanding sentiment is the number one goal. This includes Sentiment category figuring out whether a given piece of textual content expresses a wonderful, terrible, or neutral sentiment. Sentiment analysis: Assigning sentiment rankings on a non-stop scale, consisting of sentiment polarity ranging from strongly negative to strongly positive.

In **BERT**, the **[CLS] token** stands for "classification token" is a special token added at the beginning of every input sequence. It serves as a representative summary of the entire sequence for downstream tasks, such as classification or regression.
During training, the output embedding corresponding to the [CLS] token (from the final hidden layer) captures contextual information from the entire sequence. For tasks like sentiment analysis or fake news detection, this embedding is fed into a dense layer (or multiple layers) to perform classification, making it the core feature for summarizing the input's overall meaning.

3.3.1 BERT – Without Sentiment Feature
Tokens: ['[CLS]', 'japan', 'arrests', 'north', 'korean', 'crew', 'amid', 'mystery', 'boat', 'arrivals', "[SEP]']
Label: Japan arrests North Korean crew amid mystery boat arrivals
Tokens_tensor:tensor([101, 2900, 17615, 2167, 4759, 3626, 13463, 6547, 4049, 25470,102])
Segments_tensor:tensor([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
Label_tensor: 1

3.3.2 BERT – With Sentiment Feature
Input of the BERT model + Sentiment feature
Tokens: ['[CLS]', 'japan', 'arrests', 'north', 'korean', 'crew', 'amid', 'mystery', 'boat', 'arrivals', '##ne', '##g', '[SEP]']
Label: Japan arrests North Korean crew amid mystery boat arrivals NEG
Tokens_tensor: tensor([101, 2900, 17615, 2167, 4759, 3626, 13463, 6547, 4049, 25470,
2638, 2290, 102])
Segments_tensor: tensor([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
Label_tensor: 1

3.4 Algorithms
In this section, the algorithms used to design this proposed solution has been explained in an elaborated manner

3.4.1 BERT:
BERT, or Bidirectional Encoder Representations from Transformers, is a transformer-based model designed for natural language processing tasks. The core idea of BERT lies in its ability to capture bidirectional contextual information from the input text, allowing it to understand the relationships and nuances between words in a sequence. Here is a detailed explanation of BERT with equations:

3.4.2 Tokenization:
BERT begins by tokenizing the input text into subwords or words. The input is then represented as a sequence of tokens $X = \{x_1, x_2, \ldots x_n\}$. where n is the length of the sequence. (1)

Here
- x_i represents the tokens in the input sequence.
- n is the length of the sequence.

Tokenization splits complex words into smaller subwords to handle out-of-vocabulary words effectively. For example, the word *"unhappiness"* might be tokenized as ["un", "##happiness"], where "##" denotes a subword.

3.4.3 Input Embedding:
Each token is transformed into an embedding vector. This process is represented as:

$$X_{embed} = \{x_{embed1}, x_{embed2}, \ldots\ldots x_{embedn}\} \quad (2)$$

Here

X_{embed} represents the embedding vector for the i^{-th} token.

The embedding vector is obtained by looking up a pre-trained embedding matrix trained during BERT's pretraining process**.**

3.4.4 Positional Encoding:
Positional encoding is a crucial component in transformer-based models like BERT, aiming to provide information about the order or position of words in a sequence. Since transformers do not inherently understand the sequential order of tokens, positional encoding is introduced to give the model a sense of the token positions.

In BERT, the positional encoding is added directly to the input embeddings, and it's designed to be robust to varying sentence lengths. The encoding is a combination of sine and cosine functions with different frequencies. The formula for the positional encoding is as follows:

$$PE(i, 2j) = \sin\left(\frac{i}{10000^{2j/d}}\right) \quad (3)$$

$$P_E(i,2j+1) = \cos\left(\frac{i}{10000^{2j/d}}\right) \quad (4)$$

where d is the dimension of the embedding.

$$X_{positional} = X_{embed} + \text{Positional Encoding} \quad (5)$$

Here:
- i represents the position of the token in the sequence.
- j refers to the dimension of the positional encoding vector.
- d is the dimensionality of the model's embeddings
- P_E is the positional encoding
- PE(i,2j) and PE(i,2j+1) compute the sine and cosine values for even and odd dimensions, respectively.

These functions use different frequencies of sine and cosine to create unique positional encodings, which allow the model to understand both absolute and relative token positions within the sequence.

3.5 Transformer Architecture
BERT consists of multiple transformer blocks. The key equations for the transformer block are:

a. Multi-Head Self-Attention:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (6)$$

where Q, K, and V are the query, key, and value matrices, and d_k is the dimension of the key vectors.

- Then $\sqrt{d_k}$ is a scaling factor to ensure numerical stability. The dot product QK^T measures the similarity between queries and keys.

The softmax function normalizes the attention scores, assigning higher weights to more relevant tokens. These weights are then applied to the value matrix V, aggregating the relevant contextual information.

b. Feed-Forward Layer:

$FFN(x)=ReLU(xW_1+b_1)W_2+b_2$ (7)

Where

- x is the input to the feed-forward layer.
- W_1 and W_2 are weight matrices, and b_1 and b_2 are biases.
- ReLU (Rectified Linear Unit) introduces non-linearity to enhance the model's expressive power.

3.5.1 BERT Architecture:

BERT stacks multiple transformer blocks to capture hierarchical and contextual information. The output of the final transformer block is used for downstream tasks.

$H=Transformer\ (X_{positional})$ (8)

Where H represents the final hidden states for all tokens in the sequence, these hidden states are then used for downstream tasks such as classification.

3.5.2 Pre-training Objectives:

BERT is pre-trained using two tasks: Masked Language Model (MLM) and Next Sentence Prediction (NSP).

a. Masked Language Model:

The Masked Language Model (MLM) is another pre-training objective used in models like BERT. It involves randomly masking some of the words in the input text and training the model to predict those masked words based on the context provided by the surrounding words. This task encourages the model to understand the bidirectional dependencies between words and learn contextualized representations.

$P\ (x_i|x_1,\ x_2,\ x_{i-1},x_{i+1},...,x_n)$ (9)

Where:

- x_i is the masked token.
- $x_1,x_2,\ldots,x_{i-1},x_{i+1},\ldots,x_n$ are the surrounding context tokens.

This objective helps the model learn contextualized word representations by understanding bidirectional dependencies between tokens.

b. Next Sentence Prediction:

Next Sentence Prediction (NSP) is one of the pre-training objectives used in models like BERT to encourage the model to understand the relationship between two consecutive sentences in a document. The goal of NSP is to determine whether a given pair of sentences in a training sample is consecutive in the original text or if they have been randomly paired. This task aids in capturing a deeper understanding of context and relationships between sentences.

$P(IsNext|Sentence_A,Sentence_B)$ (10)

This task encourages the model to learn inter-sentence relationships, enabling it to handle tasks like question answering and sentence similarity.

3.5.2 Fine-Tuning:

For sentiment classification, the pre-trained BERT model is fine-tuned on a labeled sentiment dataset. A classification layer is added on top of the [CLS] token representation:

$Output=Softmax(Linear(H[CLS]))$ (11)

where H[CLS] is the representation of the [CLS] token from the final transformer block.

Where

H[CLS] is the representation of the [CLS] token from the last layer.

The linear layer maps the hidden representation to the number of classes.

The softmax function converts the logits into probabilities for classification.

Fine-tuning ensures that BERT adapts its pre-trained knowledge to the specific task, leveraging its rich contextual understanding for accurate predictions.

3.5.3 Long Short-Term Memory (LSTM)

The LSTM algorithm is composed of memory cells, input gates, forget gates, and output gates. These components work together to selectively store and retrieve information over time, making LSTMs suitable for capturing long-range dependencies in sequential data.

3.5.4 Memory Cell State:

The memory cell state, denoted as C_t, is updated through a combination of input, forget, and output operations. The new cell state ($C_{t)}$ is a function of the previous cell state (C_{t-1}), the input gate output (i_t), the forget gate output (f_t), and the output gate output (o_t):

$C_t=ft*C_{t-1}+i_t*C`_t$ (12)

Here, C`t is the candidate cell state, which is a function of the current input (xt) and the associated weight matrices ($W_{f,}W_{i,}W_c$)

$C`_t=\tanh(W_f\cdot[h_{t-1},x_t]+b_f)$ (13)

3.6 Input Gate:

The input gate, denoted as i_t, determines how much of the new information should be stored in the memory cell state. It is computed as a sigmoid function of the input (x_t) and the previous hidden state (h_{t-1}), with associated weight matrices (W_{ii},W_{hi}) and bias (b_i)

$i_t=\sigma(W_{ii}\cdot x_t+W_{hi}\cdot h_{t-1}+b_i)$ (14)

3.6.1 Forget Gate:

The forget gate, denoted as f_t, decides what information from the previous cell state (C_{t-1}) should be discarded. It is computed using a sigmoid function similar to the input gate:

$f_t=\sigma(W_{if}\cdot x_t+W_{hf}\cdot h_{t-1}+b_f)$ (15)

3.6.2 Output Gate:

The output gate, denoted as o_t , controls how much of the cell state should contribute to the output. It is computed as a sigmoid function:

$o_t=\sigma(W_{io}\cdot x_t+W_{ho}\cdot h_{t-1}+b_o)$ (16)

3.6.2 Hidden State:

The hidden state (h_t) is the output of the LSTM cell and is determined by the cell state and the output gate:

$h_t=o_t\cdot\tanh(C_t)$ (17)

3.6.3 Pseudo Code

Algorithm 1 BERT Sentiment Classification for Fake News Detection

1: **Input:** Tokenized input sequence X = $\{x_1, x_2, ..., x_n\}$
2: **Output:** Sentiment prediction y (0 for non-fake, 1 for fake)
for each token x_i **in** X **do**
3: **Embed token:**
4: $x_{embed_i} \leftarrow Embed(x_i)$ **end for**
5: Add positional encoding: $X_{positional} \leftarrow X_{embed} + Positional\ Encoding$
for each transformer block **do**
6: $H \leftarrow Transformer\ Block(X_{positional})$ **end for**
7: Extract [CLS] token representation: $H_{[CLS]} \leftarrow h_0$
8: $y_{sentiment} \leftarrow Softmax\ Linear(H_{[CLS]})$ ▷ Softmax followed by Linear
9.$y \leftarrow Sigmoid\ Linear(H_{[CLS]})$ ▷ Sigmoid followed by Linear

The pseudocode outlines the process of using a BERT-based model for sentiment classification and fake news detection. The input is a tokenized sequence X = \{x1, x2,….., xn}, where each token represents a part of the text, such as a word or subword. The output is a binary sentiment prediction y, where y = 0 indicates non-fake news, and y = 1 indicates fake news. The first step involves embedding each token into a dense numerical representation that captures semantic meaning. This is achieved by transforming each token xi into its corresponding embedding vector xembedi using a pre-trained embedding layer. To ensure the model understands the sequence order of tokens, positional encoding is added to the token embeddings. This positional information is crucial as transformer models like BERT do not inherently consider the order of tokens. The embeddings and their positional encodings are then passed through multiple transformer blocks, which refine the token representations by capturing contextual information. Each transformer block applies multi-head self-attention and feed-forward layers to allow the model to learn relationships between tokens in a bidirectional manner, considering both the left and right contexts. After processing the input through the transformer blocks, the hidden state of the [CLS] token is extracted. The [CLS] token is a special token added to the beginning of the input sequence and is designed to aggregate information from the entire sequence. Its hidden state H[CLS] serves as a holistic representation of the input and is used for classification tasks. The representation of the [CLS] token is then passed through two operations for prediction. First, a softmax function is applied to a linear transformation of H[CLS] to determine the sentiment class probabilities, such as positive or negative sentiment. Second, a sigmoid function is applied to another linear transformation of H[CLS] to produce a probability value for fake news detection. The sigmoid output maps the representation to a value between 0 and 1, where 0 signifies non-fake news and 1 signifies fake news. By combining sentiment classification with fake news detection, the model effectively captures nuanced relationships and patterns in the input text, leveraging BERT's ability to understand bidirectional context for robust classification.

4. RESULT AND DISCUSSION

The experiment was conducted on a Windows-based system equipped with 8GB of RAM and an Intel Core i5 8th Gen processor. The proposed BERT model has been developed using python packages available for NLP techniques. In the following sub section, the results of the proposed model have been given and it is further compared with the previous work.

4.1 Dataset Description

Title: This column likely contains the headline or title of each news article. It provides a brief overview of the main topic or subject covered in the article.
Text: This column likely contains the main text or content of each news article. It includes the detailed information, context, and discussions related to the topic presented in the title.
Subject: This column seems to specify the subject category to which each news article belongs. In the context of your dataset, the subject category is labeled as "politics News."**Class:** This column appears to provide a classification or label for each news article. In your dataset, the values "True" seem to indicate that the articles are indeed valid or true news items that are shown in Table.1. This column could potentially be used for classification or verification purposes.

Table 2. Dataset Sample with various News domains.

Title	Text	Subject	Class
As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	PoliticsNews	True
U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	PoliticsNews	True
Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	PoliticsNews	True
FBI Russia probe helped by Australian diplomat...	WASHINGTON (Reuters) - Trump campaign adviser ...	PoliticsNews	True

In the Table:2 Class of Dataset is with their count is deployed. This gives clear picture of memory allocation that count of true of True and Fake data from the whole data received.

Table 3. Classes of Dataset based on perception

Class	**Count**
True	21418
False	23480

"The table above indicates that the 'True' class comprises news articles categorized as valid or true, while the 'False' class includes news articles labeled as false or not valid. The counts for each class are as follows: the 'True' class has a count of 21,418, and the 'False' class has a count of 23,480.

4.2 Exploratory Data Analysis

True News Subject Distribution: It portrays the study of the analysis True news distribution in which the segregation count of the subject with respect to the political and world news. Thus, it's possible to elaborate the purpose and origin of data. On the basis of the observation done by the below given figure, the number of true news based on political is seems to be higher than the world news.

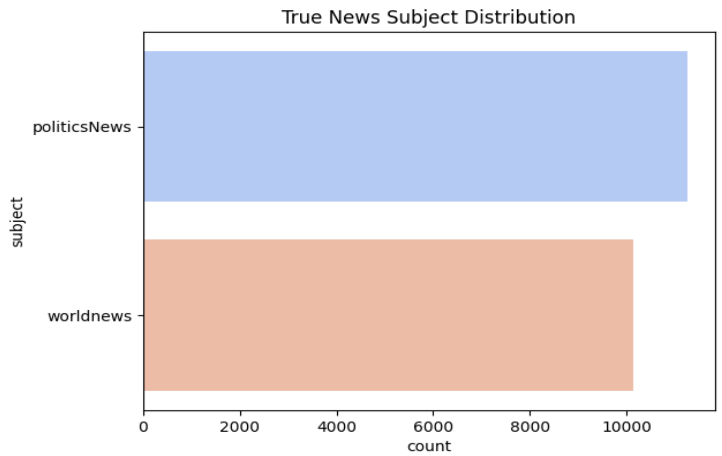

Fig 4. Overview chart for News Distribution

The bar chart (figure 2) visually represents the distribution of true news articles based on their subject matter, specifically focusing on two categories: "politicsNews" and "worldnews." It shows that the "politicsNews" category has a higher count of articles compared to "worldnews." The blue bar represents the political news articles, while the orange bar represents world news articles. The horizontal axis measures the count of articles, and the vertical axis corresponds to the news subjects. This distribution provides insight into the balance of news topics in the dataset, highlighting that political news articles are more prevalent than world news, which could inform tasks such as news categorization, topic analysis, or detecting trends and biases in the data. Overall, it seems that you have presented information about two distinct subsets within a larger dataset. One subset comprises 10,000 articles related to world news, while the other subset consists of 10,000 articles focused on political news. These counts offer valuable insights into the distribution of news articles based on their respective subject matters within the dataset. Overall, it seems that you have presented information about two distinct subsets within a larger dataset. One subset comprises 10,000 articles related to world news, while the other subset consists of 10,000 articles focused on political news. These counts offer valuable insights into the distribution of news articles based on their respective subject matters within the dataset."

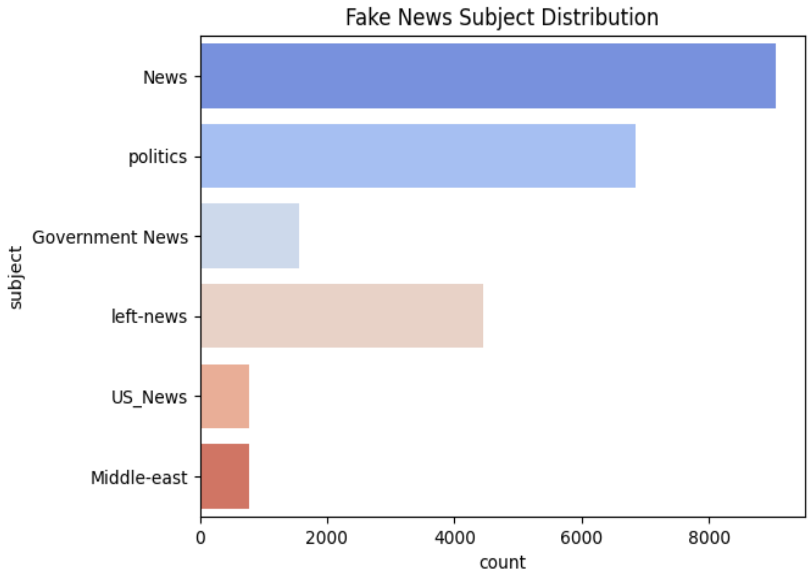

Fig 5. Fake News Subject Distribution

In the above figure 3 the subject distribution for the category of fake news has been illustrated. Here it is visible the number of subjects identified for fake news is higher than the number of subjects identified for real news. By using the bar chart, the distribution of fake news according to different subject lines is well illustrated. The most attacked category is News, trailed by Politics, and Government News. These areas of discussion are usually politically related and may contain politically influenced or politically influenced information since they are sensitive to such manipulations. Some of the domains like 'left-news' 'US_News' and 'Middle-east' are less frequently used and it indicates a strategic mode to propagate fake news in particular domains. Furthermore, the subjective nature of the "fake news" definition underscores the need for careful interpretation of data.

- **Word cloud – True News**

The word cloud includes all the data that are combined with True news and Fake news. The word cloud offers a glimpse into the key themes and prominent figures associated with true news. It showcases the diverse range of topics covered, including politics, international relations, economics, and social issues. The prominence of words like "Trump," "Obama," "Clinton," and "Putin" underscores the focus on political leaders and their actions. The presence of terms like "Senate," "House," and "Congress" indicates a strong emphasis on legislative processes and policy decisions. Additionally, the inclusion of words like "economy," "trade," and "tax" suggests a significant focus on economic matters. Overall, the word cloud reflects the multifaceted nature of true news, encompassing a wide spectrum of subjects and highlighting the importance of political, social, and economic issues.

Fig 6. Dumped picture showing True News

- **Word cloud – Fake News**

Analyzing the word cloud for fake news one can see that many words are associated with politics and controversy. Such words as "Trump", "Clinton", "Obama" are very big, indicating that these people are mentioned in fake news. Besides, there are such terms as 'fake news," "lie," "conspiracy", meaning that fake news is more often the distribution of misinformation. The word cloud also includes words like "Russia," "Iran," and "North Korea," which shows that fake news can be about international relations and conflicts. Overall, the word cloud shows that fake news is often about making sensational claims about political figures and creating controversy.

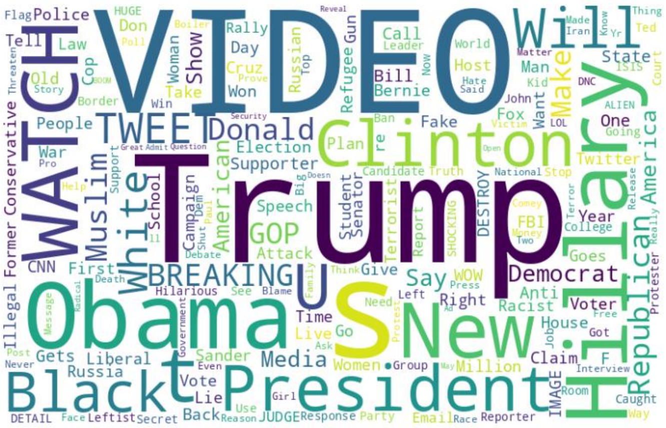

Fig 7. Dumped picture showing Fake News

- **Summary of Textual data:**

The word count Row and Column calculation was integral part in determining the count of data exactly used.

Table 4. Total count of samples.

Number of Rows	44898
Maximum length of News	42
Mean length of News	12.45347

The dataset shown inTable3 comprises a total of 44,898 rows or instances. The longest news article within the dataset contains 42 words, reflecting its extensive content. On average, the news articles are relatively concise, with an approximate length of 12.45 words per article. These statistics offer which is shown in Fig6 insights into the dataset's size, content distribution, and average article length.

- **Count of length of News –**

The count of length portrays the density of words in the cluster.

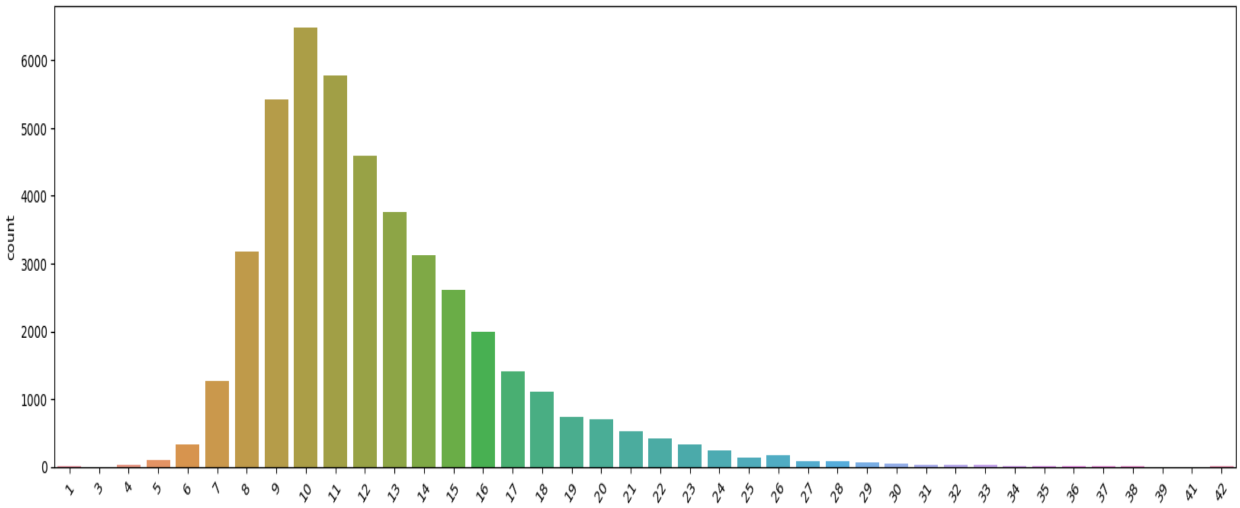

Fig 8. Frequency of Different News Article Length

4.3 Evaluation of Sentiment Classification

The confusion matrix shows as figure 7 provides insights into the performance of an LSTM model in classifying news articles as true or fake. The model exhibits exceptional accuracy in identifying fake news, with 4103 out of 4233 correctly classified as true negatives. However, it struggles slightly with correctly classifying true news, misclassifying 140 out of 4500 as false negatives. Despite this shortcoming, the overall performance of the LSTM model in this task is commendable, showcasing its effectiveness in distinguishing between true and fake news. The model correctly identified 4,360 instances as "True." It wrongly labeled 140 "True" instances as "Fake." Also, it misclassified 130 "Fake" instances as "True." The model accurately classified 4,103 instances as "Fake." These results provide insights which are shown in Table4 into the model's performance.

LSTM

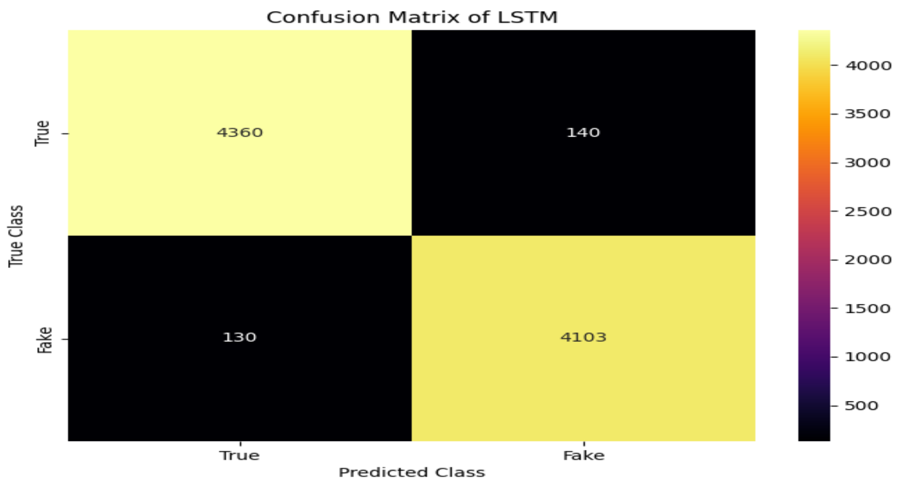

Fig 9. Confusion Matrix Of LSTM

4.4 BERT without Sentimental Feature

The confusion matrix reveals in figure 8 the performance of a BERT model without sentimental features in classifying news articles as true or fake. The model demonstrates exceptional accuracy in identifying fake news, with 4680 out of 4722 correctly classified as true negatives. However, it exhibits a slight tendency to misclassify true news as fake, with 87 out of 4258 incorrectly categorized as false negatives. Despite this minor shortcoming, the overall performance of the BERT model in this task is commendable, showcasing its effectiveness in distinguishing between true and fake news.
The model accurately recognized 4,171 instances as "True." However, it made an error in 87 cases by classifying them as "Fake" instead. Similarly, it misclassified 42 instances of "Fake" as "True." On the positive side, the model correctly classified 4,680 instances as "Fake." These insights reflect the model's performance and potential areas for improvement are shown in the above figure.

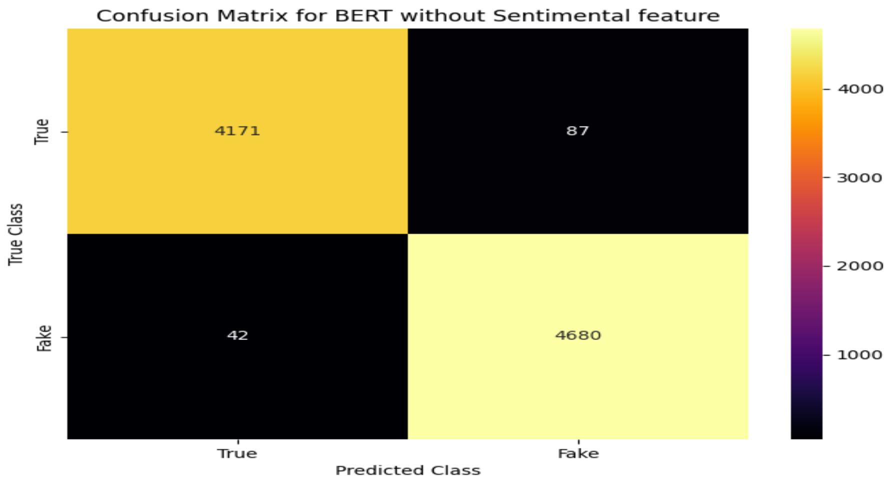

Fig 10. Confusion Matrix for BERT without Sentimental Feature

Precision values are high at 0.99 for both "True" and "Fake" classes, indicating accurate positive predictions showcased in Table 11. Recall values are both 1.00, showcasing the model's ability to correctly identify instances. F1-scores are 0.99 for "True" and 0.98 for "Fake," signifying balanced performance across classes.

Table 5: F1score and Precision study BERT without Sentiment feature

	Precision	Recall	F1-score
True	0.99	1.00	0.99
Fake	0.99	1.00	0.98

4.5 BERT with Sentiment Feature

The confusion matrix illustrates in figure 9 the performance of a BERT model equipped with sentimental features in classifying news articles as true or fake. The model exhibits exceptional accuracy in identifying fake news, correctly classifying 4770 out of 4792 articles. However, it demonstrates a slight tendency to misclassify true news as fake, with 17 out of 4258 articles incorrectly categorized. Despite this minor shortcoming, the overall performance of the BERT model in this task is commendable, showcasing its effectiveness in distinguishing between true and fake news, particularly when augmented with sentimental features.

The model accurately recognized 4,241 instances as "True." However, it made an error in 17 cases by classifying them as "Fake" instead. Similarly, it misclassified 22 instances of "Fake" as "True." On the positive side, the model correctly classified 4,700 instances as "Fake." These insights reflect the model's performance and potential areas for improvement has been shown in the above figure.

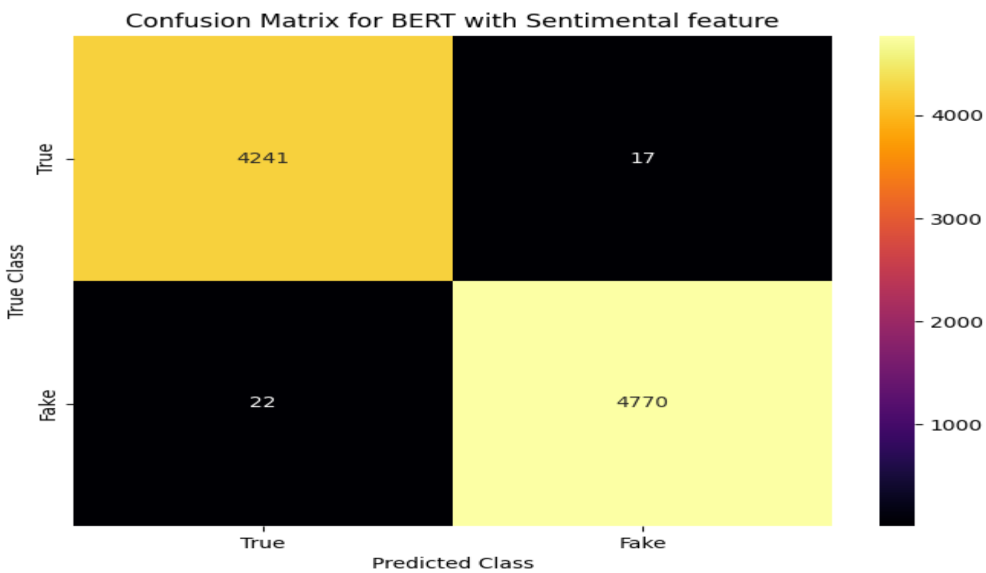

Fig 11. Confusion Matrix for BERT with Sentiment Feature

In the comparison, "BERT with Sentiment Features" stands out as the best-performing algorithm. The bar chart given as figure 11 provides a comparative analysis of three algorithms: LSTM, BERT, and BERT with sentimental features, in their performance on a news classification task. All three models exhibit high testing accuracy, with BERT with sentimental features achieving the highest accuracy of 98%. LSTM follows closely with 97% accuracy, while BERT without sentimental features achieves 97.1%. In terms of precision, BERT with sentimental features maintains its lead with a score of 98%, indicating a high rate of true positive predictions. LSTM and BERT without sentimental features achieve similar precision scores of 97%. For recall, BERT with sentimental features continues to outperform the other models, achieving a score of 97.2%. LSTM and BERT without sentimental features demonstrate comparable recall scores of 97%. The F1-score, which balances precision and recall, also favors BERT with sentimental features, with a score of 98%. LSTM and BERT without sentimental features achieve similar F1-scores of 97%. Fig12. It is achieved a remarkable 98% testing accuracy, 98% precision, 98% recall, and a notable 98% F1-score. These results highlight its ability to accurately predict positives while maintaining balance, making it the preferred choice based on these metrics.

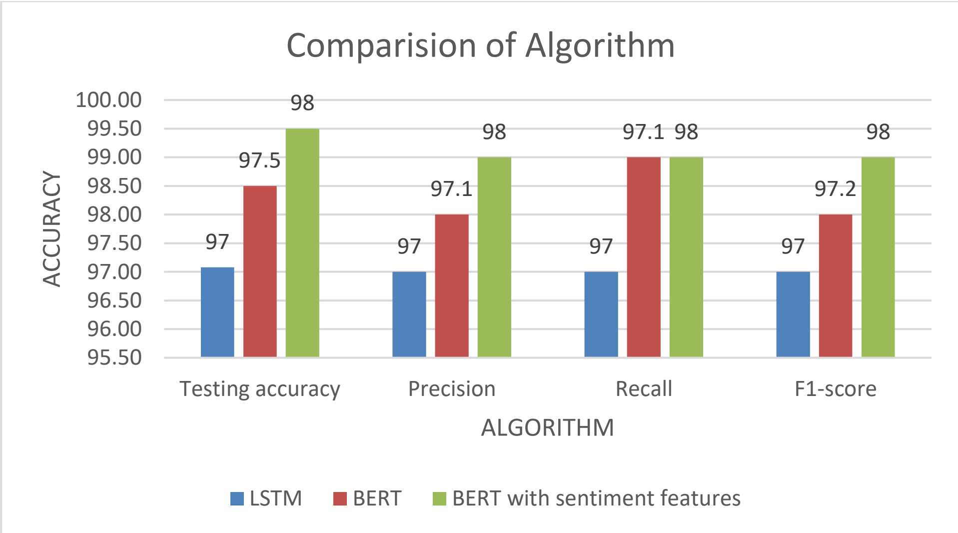

Fig 12. Final Comparison

Initially, this work has been started by incorporating BiLSTM and CNN along with word embedding approach, later it was further enhanced by incorporating 1DCNN in place of CNN, to improve its performance in terms of its accuracy. Currently, this work has been proposed by deploying hybrid model with the help of the combination of BERT model with Sentimental feature. Hence it achieves the highest accuracy by 98%.

Table 6. Comparison analysis between previous work and proposed work

S.No	**Technique Used**	**Accuracy (%)**
1	BilSTM +CNN with Word2Vec and Glove	79
2	BilSTM +1DCNN with Word2Vec and Glove	82
3	**Hybrid model (BERT with Sentimental feature)**	**98**

5. Conclusion and Future Work

The evaluation of three different models indicates that BERT-Sentiment Analysis exhibits the most significant progress in the model. This topology consistently outperforms all metrics in the algorithm, surpassing LSTM in terms of performance. Although LSTM performs well, BERT stands out with superior performance. The future suggests the potential discovery of more advanced algorithms, with global data server updates aiding in distinguishing between genuine and fake content on major global platforms. LSTM's primary strength lies in effectively segregating fake and true data, achieving a testing accuracy of 97%. However, its proficiency in interpreting data for final output segregation is minimal. In contrast, BERT analysis, when assessed, demonstrates a testing accuracy of 98%. A further advancement is achieved by combining BERT and LSTM, adding sentimental analysis, resulting in a testing accuracy of 98%. This model offers improved hierarchical development for data segregation and model refinement. The F1 score shows promise in BERT analysis, particularly in the sentimental analysis of the BERT combination. However, the algorithmic model has certain limitations, occasionally struggling to identify fake information. Overcoming coding complexity in the structural component of the feature system remains a concern in this methodology.

Future research will focus on enhancing the robustness of the models by integrating additional data augmentation techniques, domain adaptation methods, and advanced pretraining strategies for diverse datasets. Fine-tuning existing models, exploring hybrid architectures that combine BERT with other deep learning techniques, and leveraging ensemble approaches can further boost performance for both sentiment analysis and fake news detection. Additionally, real-time deployment scenarios and the inclusion of multi-modal data (e.g., images or videos) will be explored to improve practical applications. A deeper emphasis on explainable AI techniques will ensure greater transparency and interpretability of predictions.Considering this proposed approach of BERT and LSTM with sentimental analysis, future work should focus on fine-tuning this hybrid model for even more precise data segregation. The study indicates a positive trend in hierarchical model development, and efforts should be directed towards optimizing the input features to overcome the coding complexities. Furthermore, attention to refining the algorithm's F1 score, particularly in sentimental analysis, will contribute to a more comprehensive evaluation of model performance. As the field of fake news detection evolves, continuous efforts should be made to address limitations, improve accuracy, and ensure the adaptability of the algorithm to emerging challenges in the dynamic landscape of information authenticity.

REFERENCES

[1] Ali, A.M. *et al.* (2022) 'Deep Ensemble Fake News Detection Model Using Sequential Deep Learning Technique', *Sensors*, 22(18), p. 6970. Available at: https://doi.org/10.3390/s22186970.

[2] Alonso, M.A. *et al.* (2021) 'Sentiment Analysis for Fake News Detection', *Electronics*, 10(11), p. 1348. Available at: https://doi.org/10.3390/electronics10111348.

[3] Bhutani, B. *et al.* (2019) 'Fake News Detection Using Sentiment Analysis', (August).

[4] Chaubey, P.K. *et al.* (2022) 'Sentiment Analysis of Image with Text Caption using Deep Learning Techniques', *Computational Intelligence and Neuroscience*. Edited by V. Kumar, 2022, pp. 1–11. Available at: https://doi.org/10.1155/2022/3612433.

[5] Chen, Q., Zhang, W. and Lou, Y. (2020) 'Forecasting Stock Prices Using a Hybrid Deep Learning Model Integrating Attention Mechanism, Multi-Layer Perceptron, and Bidirectional Long-Short Term Memory Neural Network', *IEEE Access*, 8, pp. 117365–117376. Available at: https://doi.org/10.1109/ACCESS.2020.3004284.

[6] Dufraisse, E. *et al.* (2023) 'MAD-TSC: A Multilingual Aligned News Dataset for Target-dependent Sentiment Classification', in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 8286–8305. Available at: https://doi.org/10.18653/v1/2023.acl-long.461.

[7] Galli, A. *et al.* (2022) 'A comprehensive Benchmark for fake news detection', *Journal of Intelligent Information Systems*, 59(1), pp. 237–261. Available at: https://doi.org/10.1007/s10844-021-00646-9.

[8] Ju, X. *et al.* (2021) 'Joint Multi-modal Aspect-Sentiment Analysis with Auxiliary Cross-modal Relation Detection', in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 4395–4405. Available at: https://doi.org/10.18653/v1/2021.emnlp-main.360.

[9] Yang Z., Yang D., Dyer C., He X., Smola A., Hovy E. Hierarchical attention networks for document classification; Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; San Diego, CA, USA. 12–17 June 2016; Stroudsburg, PA, USA: ACL; 2016. pp. 1480–1489.

[10] Bojanowski P., Grave E., Joulin A., Mikolov T. Enriching word vectors with subword information. *Trans. Assoc. Comput. Linguist.* 2017;**5**:135–146. doi: 10.1162/tacl_a_00051.

[11] Pennington J., Socher R., Manning C.D. Glove: Global vectors for word representation; Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing; Doha, Qatar. 25–29 October 2014; Stroudsburg, PA, USA: ACL; 2014. pp. 1532–1543.

[12] Ortis A., Farinella G. M., Battiato S. Survey on visual sentiment analysis. *IET Image Processing* . 2020;14(8):1440–1456. doi: 10.1049/iet-ipr.2019.1270.

[13] Khambra G., Shukla P. Novel machine learning applications on fly ash based concrete: an overview. *Materials Today Proceedings* . 2021 doi: 10.1016/j.matpr.2021.07.262.

[14] Jain N., Shanti Rathore Dr., Shukla P. K. Designing efficient optimum reduced order IIR filter for smoothening EEG motion artifacts signals. *Design Engineering* . 2021;**2021**:5080–5101.

[15] Chen P., Sun Z., Bing L., Yang W. Recurrent Attention Network on Memory for Aspect Sentiment Analysis; Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing; Copenhagen, Denmark. 9–11 September 2017; Stroudsburg, PA, USA: ACL; 2017. pp. 452–461.

[16] Orestes Appel, Francisco Chiclana, Jenny Carter, and Hamido Fujita. 2016. A hybrid approach to the sentiment analysis problem at the sentence level. Knowl.-Based Syst., 108:110–124.

[17] Peng Chen, Bing Xu, Muyun Yang, and Sheng Li. 2016. Clause sentiment identification based on convolutional neural network with context embedding.In Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), 2016 12th International Conference on, pages 1532–1538.

[18] Lingjia Deng and Janyce Wiebe. 2015. Joint prediction for entity/event-level sentiment analysis using probabilistic soft logic models. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing.

Dr. M.Anusha is Assistant Professor and Head in Computer Applications in our college. She has 12 years of experience in teaching. Her specialization is soft computing . She has published 2 patents, and 17research articles in reputed journals in her expertise. She is guiding 4 ph.D research scholars.She has served as a resource person in many technical programmes. She is currently working on 2 projects(1 completed).

I am Leelavathi Ph.D Full-Time Research Scholar in Department of Computer Science, National College, Trichiruappalli. Currently doing my research topic is Sentiment Analytics. Research interests include data analytics, Natural Language Processing and Deep Learning.