

Stance-Based Fake News Identification on Social Media with Hybrid CNN and RNN-LSTM Models

M.D.P.P. Goonathilake, P.P.N.V. Kumara

Abstract— Today, fake news can be readily generated and disseminated via social media platforms. Misinformation and hoaxes propagated via online social media or traditional news media are commonly referred to as fake news. Stance-based fake news is based on the opinions of an audience rather than providing correct facts. This study presents a hybrid model focusing on the CNN model and RNN-LSTM model to identify fake news. A balanced dataset of 216k news items called ‘SherLock-FakeNewsNet’ is explored throughout the study resulting in the proposed hybrid model. The NLTK toolkit is utilized to perform some noise reduction. After that, the Keras tokenizer and pre-processor are used for the tokenization and pre-processing steps. Next, the text data is fitted and represented using pre-trained GloVe word embeddings. CNN model is applied with Conv1D layers and MaxPooling layers to extract higher-level features of the text. Following this, to detect the longer context of the given text and to capture interdependencies among word sequences LSTM units are employed in the RNN model. Dropout layers are carefully chosen to reduce overfitting in the final hybrid model. The proposed hybrid model achieves the highest accuracy rate of 92% by outperforming most of the conventional models today with an average Precision of 91%, Recall of 91%, and F1-Score of 91% using an Adam optimizer and a Binary Cross-Entropy loss function. Based on the five experiments performed in the study, we have presented the results by comparing our proposed hybrid model with five related datasets.

Keywords— Deep Learning, Fake News, Hybrid Approach, Natural Language Processing, Social Media

I. INTRODUCTION

Nowadays almost half of the population worldwide use social media news feeds rather than conventional news sites to check news. As stated in [1] authors examined that about 60% of Americans looked for news on social media news feeds in contrast to 49% in 2012. From the study [2] authors discovered that social media outpaced other news channels as the leading news source. Because of these occurrences, false news can be easily generated and disseminated across many social media platforms. There is no agreed-upon explanation for false news. Commonly, false news is referred to as intentionally creating false information and stories to delude the audience.

Correspondence: MDPP Goonathilake (E-mail: pathumveyron24@gmail.com) Received:16.09.2021, Revised:12.07.2023, Accepted: 01-09-2023

M.D.P.P. Goonathilake and P.P.N.V. Kumara are from Department of Computer Science, General Sir John Kotelawala Defence University, Ratmalana, Sri Lanka. (pathumveyron24@gmail.com, nandana@kdu.ac.lk)

This paper is an extended version of the paper “CNN, RNN-LSTM Based Hybrid Approach to Detect State-of-the-Art Stance-Based Fake News on Social Media” [4] presented at the ICTer 2020.

DOI: <https://doi.org/10.4038/ictcr.v16i3.7234>

© 2023 International Journal on Advances in ICT for Emerging Regions



This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

According to [3] authors explored that, to deceive users, false news consisted of incorrect information generated with vicious purposes. As reported by [5] false information is a serious threat to equality, freedom of speech, and journalism as well. From [6] authors demonstrated that social media sites and gossip sites are the largest sources of escalating sharing of false information or rumours.

Based on the distorted spread on social media many instances have occurred. In the 2016 US presidential election, “Pizzagate” false news was extensively disseminated on Twitter. And it was projected that almost one million tweets were attached to false news “Pizzagate” [7] after the presidential election was finished. In [8] authors examined the social bounce of the circulated news, “Palestinians recognizing Texas as part of Mexico”. Figure 1 represents the social bounce due to the above news.

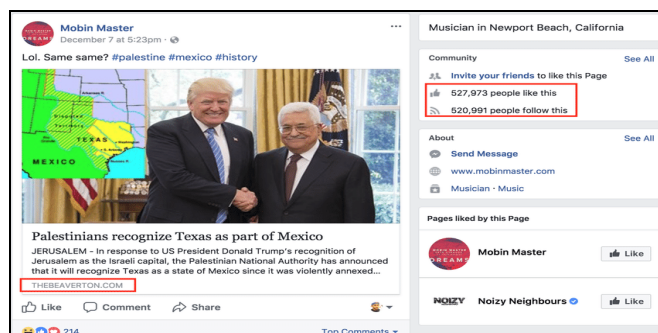


Fig. 1. Palestinians recognizing Texas as part of Mexico

During the Easter Sunday Attacks in Sri Lanka in 2019, the government decided to curtail the access to social media sites as a consequence of false information propagated across social media platforms that led to misunderstandings between religions [9]. This incident evidently proved how social media evolved into a crucial platform for inciting sensitive subjects to the public. Figure 2 represents false news circulated in social media in 2019.



Fig. 2. False news circulated in Sri Lanka in 2019

According to the study [10] authors have found that Facebook has recorded more than 150 million views for the top 100 fake news stories of 2019. Most of the false news stories are related to the category of politics.

Table I points out the top 10 most viewed fake news stories shared on Facebook in 2019.

TABLE I
MOST VIEWED FAKE NEWS STORIES ON FACEBOOK IN 2019

Fake News Headline	Category
Trump’s Grandfather Was A Pimp And Tax Evader; His Father A Member Of The KKK.	Politics
Pelosi Diverts \$2.4 Billion From Social Security To Cover Impeachment Costs.	Politics
Ocasio-Cortez Proposes Nationwide Motorcycle Ban.	Politics
Trump Is Now Trying To Get Mike Pence Impeached.	Politics
Omar Holding Secret Fundraisers With Islamic Groups Tied to Terror.	Politics
BREAKING: Nancy Pelosi’s Son Was Exec At Gas Company That Did Business In Ukraine.	Business
Dems Vote To Enhance Med Care for Illegals Now, Vote Down Vets Waiting 10 Years for Same Service.	Politics
Trump’s Wall Costs Less Than The Obamacare Website, A Quote By Tim Allen.	Politics
NYC Coroner Who Declared The Death Of Jeffrey Epstein A Suicide Made Half A Million Dollars A Year Working For The Clinton Foundation Until 2015.	Politics
Joe Biden Calls Trump Supporters ‘Dregs of Society’.	Politics

As per the above, it is witnessed that false news emerges as one of the most serious dangers to social media platforms. Therefore, it is required to finding an appropriate and proper mechanism that can simply identify false information.

Based on the action and the nature of the news, false news can be classified into various types. The typical forms of false news can be represented as follows.

- Visual-based: The graphical representation of the information is depicted in this type of news.

- User-based: The user profiles and their intended audience-based news are contained in this news type.
- Post-based: Post-based types of news in social media sites are introduced from this news type.
- Network-based: Network-based type of news is focused on some selected groups.
- Knowledge-based: Explanation for an unsolved problem is provided by this news type.
- Style-based: Style-based types of news focus on how to deliver the information in a wrong manner to the audience.
- Stance-based: This type of news includes how the statement is made in a news article.

This study focuses on the last type, which is about identifying false news based on stance. A self-owned dataset called ‘SherLock-FakeNewsNet’ is explored throughout the study along with the proposed hybrid model.

The objectives of this research study mainly focus on:

- Observing the drawbacks of existing research works and related tools.
- Presenting a novel dataset called ‘SherLock-FakeNewsNet’ to research stance-based fake news identification on social media.
- Presenting a novel hybrid model focusing on the CNN model combined with an RNN-LSTM model to identify false information.
- Experiment with the dataset to present the capability and generality of the model over the other classical models.

The rest of the paper is formed as follows. Section II depicts the related works and tools with a comprehensive review and Section III presents the methodology of the research study. Section IV demonstrates the results of five different experiments and evaluates the discussions based on the results. Finally, Section V represents the conclusion and future work of the research study.

II. LITERATURE REVIEW

A substantial number of researchers have investigated this topic and suggested several methods to overcome this problem. News can be divided into three categories based on its content aspects, namely, style features, textual features, and visual features [11]. Natural Language Processing methods [12] like linguistic features, low-rank textual features and neural textual features are used to extract textual characteristics from news material. Linguistic features are consisted of lexical features and syntactic features. Letters per word, repetition of active words and unique words are a few character-level and word-level elements that are included in lexical features. Syntactic features are subsisted of N-grams and repetition of active words which are included in the sentence level elements [13]. Matrix factorization and tensor factorization are examples of Low-rank textual features. Clustering is frequently adopted as a method in Matrix factorization to study document representation [14]. To learn about the representation of

news article mechanisms like tensor factorization is practised. Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) are among the neural textual features. Word2Vec word-embedding vectors have been utilized by CNN [15]. Long Short-Term Memory (LSTM) neural networks are also employed in RNN [16]. The authors of [17] have proposed a strategy for developing a false news identifier that consisted of Natural Language Toolkits and Textblob, and achieved an accuracy rate of 63%.

Sentiment analysis is one of the significant research areas in Natural Language Processing. It comprises distinct research directions. In [18] authors have proposed an approach based on ensemble learning paradigms focused on a multi-objective weighted voting scheme to strengthen the predictive performance of different classification tasks. The authors also have evaluated the results using different machine learning algorithms. According to [19], authors have presented an ensemble approach for feature selection in sentiment classification. These authors also experimented with a genetic algorithm to accumulate the individual feature lists. From the study [20], the author has proposed a deep learning-based approach with CNN-LSTM architecture to sentiment analysis on product reviews gained from Twitter. Subsequently, the author also has evaluated the predictive performance with different word embeddings and with several weighting functions.

In [21] authors have examined a neural language model and bidirectional LSTM based framework for sarcasm identification on social media. Neural language model introduced by a term weighted based word embedding model with trigrams to represent text documents. Afterwards, three-layer stacked bidirectional LSTM architecture has been presented to identify sarcastic text documents. In [22] author has presented a deep learning-based approach to sarcasm identification on social media. Subsequently, the author also has explored the performance of the approach by comparing topic-enriched word embedding schemes and conventional word embedding schemes.

Through the paper [23], the author has focused on performing sentiment analysis for a massive open online course (MOOC) reviews using text mining and a deep learning-based approach. Later, the author has elaborated on the efficiency of the approach compared with ensemble learning methods and supervised learning methods. Next, from the study [24], the authors have examined a machine learning-based approach to sentiment analysis on student evaluations. Also, these authors have evaluated the results using different machine learning algorithms.

According to [25], the authors have explored the performance of instance selection methods in text sentiment classification based on fifteen benchmark instance selection methods. They have evaluated the instance selection methods by applying a decision tree classifier and radial basis function networks. Keyword extraction is a convenient method in text classification. Authors of [26] have presented five statistical keyword extraction methods by comparing them with different machine learning algorithms and ensemble methods.

In the study [27], the author has focused on different feature engineering schemes for text classification. The author has presented an ensemble classification scheme based on language function analysis. Through the paper [28],

the author has presented a hybrid supervised clustering algorithm to gain a distinct ensemble for text classification. In the study [29], the authors have examined a hybrid ensemble pruning approach for sentiment classification. And the authors have evaluated the approach by comparing it with ensemble methods and evolutionary algorithms.

From the study [30], the authors have proposed an ensemble scheme to identify satirical Turkish news articles with a deep learning-based approach. And the authors also have presented different methods like linguistic and psychological feature sets to extract features. The authors of [31] concentrated on deep neural networks to identify false news. And the authors investigated the significance of a few deep learning network architectures, including, Convolutional Neural Networks (CNN), Gated Recurrent Unit Networks (GRU) and Long Short-Term Memory Networks (LSTM) etc.

Following that, the authors have discussed the impacts of merging different neural networks, including, Convolutional Neural Networks with Gated Recurrent Unit (CNN-GRU) and Convolutional Neural Networks with Long Short-Term Memory (CNN-LSTM). Furthermore, the authors have examined the impact of word embedding characteristics in Deep Neural Networks (DNN). From the study [32], the authors have proposed a technique called TRACEMINER to handle the LSTM-RNN model in social networks and demonstrated some good accuracy for classification-related datasets.

Table II exhibits selected studies and the techniques which have been developed under this topic. Technologies and algorithms that are applied to each tool are represented with a small description of each tool. Almost every tool is accustomed to web scraping and web crawling as the main technologies for the data collection part. Furthermore, some of the tools are equipped with word-embedding techniques like GloVe and Word2vec as well. A few of the tools are based on several deep learning mechanisms like CNN, LSTM and auto-encoders etc. Some of the tools used built-in APIs and algorithms to identify false information on social media.

TABLE II
RELATED TOOLS

Title	Applied Technologies/Algorithm	Description
Hoaxy [33]	Data Scraping, Web Scraping, Twitter API, Web Syndication, RSS Parser	An online tool that can collect and track misinformation.
FakeNewsTracker [34]	LSTM, Auto-Encoder, Web Scraping, Web Crawler, Word Cloud Representation	A tool that can collect, detect and visualize misinformation on social media.
dEFEND [35]	Deep Learning Model, Web Crawler, Hoaxy API, Web Scraping, dEFEND Algorithm	A fake news detection system that shows a news propagation network.

News Verify [36]	Sentiment Analysis, Feature Extraction, Web Scraping, Web Crawling, Twitter API	A real-time news verification system.
FAKE NEWS PATTERN DETECTOR [37]	CNN, Word-Embeddings, Word2vec, Feature Extraction, Fact Extraction	A deep learning network that detects the patterns in fake news.
B.S. Detector [38]	Web Scraping, Web Syndication, Twitter API, Sentiment Analysis, Feature Extraction	A browser extension to search for unreliable sources.
SherLock [39]	Web Scraping, CNN, LSTM, Word-Embeddings, News API	A fact-checking mobile platform.

As discussed in the literature review section, a substantial number of researchers suggest various approaches to conduct the research. However, after reviewing the existing mechanisms, techniques and related works, the idea of combining Natural Language Processing techniques with several Deep Neural Networks together and building a hybrid neural network to initiate the research, emerged, because combining several Deep Neural Networks has given some highly accurate results as seen from the related research works.

Natural Language Processing techniques such as tokenization were applied to represent words in tokens. Methods like stop word removal were used to remove stop words, and several techniques were employed to clean the dataset. Word embeddings were applied to map words with vectors of real numbers. A hybrid approach was chosen mainly focusing on CNNs to extract the higher-level local characteristics from the input text. Subsequently, RNN-LSTMs were chosen to capture the long-term dependencies of the word sequences. Finally, Dropout layers were carefully selected to prevent overfitting and Dense layers were carefully introduced to enhance the overall efficiency of the hybrid model. From section III, the methodology of the research study widely elaborates the above-discussed methods that are applied to each phase of the research study.

III. METHODOLOGY

To solve the stance-based fake news problem, this research study is directed with the hybrid approach. Natural Language Processing techniques such as stop word removal, tokenization and word embeddings were used with the NLTK toolkit in the first phase of the hybrid approach. As for the second phase of the hybrid approach, sequence of deep neural networks was applied such as Convolutional Neural Networks with Convolution layers, and Maxpooling layers and Recurrent Neural Networks with Long Short-Term Memory layers. Finally, Dropout layers were carefully selected to prevent overfitting, and Dense layers were carefully introduced to enhance the overall efficiency of the hybrid model. The research study was initiated by acquiring a dataset.

A. Finalizing a Method for Collecting a Dataset

HTML parsers and web parsers had been employed as methods by researchers [40] to collect data from different sources. However, for this study, a free web scraper called

Scrapy [41] was acknowledged to gather news articles from various sources.

1) Why Scrapy:

Scrapy is a free library developed by Scrapinghub to extract data from websites. A wide range of web-scraping and web-crawling mechanisms are used in Scrapy to work on data mining, automated testing and monitoring purposes. Scrapy is used as the data collection framework using the web scraping method. And Scrapy also provides rich support for building scalable large crawling projects. Spiders and selectors use Scrapy to go through the websites and extract data from websites. Scrapy can handle the requests asynchronously, and it uses an auto-throttling mechanism to speed up the process.

Global news articles were assembled from websites such as bbc.com, politifact.com and snopes.com from the year 2018 to 2020. From government news sources such as news.lk, local news articles were collected. Furthermore, fact-checking news sources and gossip sites such as AFP Sri Lanka, afpnews.com and gossipcop.com were investigated to gather articles about false news from the year 2018 to 2020. All the various techniques above resulted in a dataset of 216,682 news articles. Table III represents the individual instances of two classes.

TABLE III
COLLECTED DATASET INDIVIDUAL INSTANCES

News Type	Count
Real News	108,341
Fake News	108,341

2) Why Choose Dataset with Exact Balance:

Choosing a balanced dataset is critical in research since it can assist eliminating bias in data processing and interpretation. A balanced dataset has an equal or nearly equal amount of samples from each class or group under consideration. This can assist in ensuring that the statistical models utilized are not bias to one group over another and that the findings are representative of the total population. As datasets included in this stance-based fake news detection area are not balanced, it was decided to use a well-balanced dataset.

3) How Labelled the Dataset:

News articles that are captured from credible news sources like bbc.com which are identified and guaranteed as real news from fact-checking websites like AFP Fact Check and Snopes were given the reliable label as 1, whereas news gathered from untrustworthy sources like gossip sites which are identified and guaranteed as fake news from the above-mentioned fact-checking websites were given the reliable label as 0. Figure 3 represents the portrait of the captured dataset by reflecting two classes of the dataset and how news sources are labelled giving the scores of 0 and 1.

	news_headline	reliable
0	Here Is A Comprehensive And Up To Date List O...	0
1	Robin Roberts and Michael Strahan spark feud r...	0
2	HILLARY Œā, -ā, çS ANTI-TRUMP MUSLIM DAD Claims ...	0
3	Factbox: Contenders for key jobs in Trump's ad...	1
4	Here's Why the Internet Is Freaking Out Over T...	1

Fig. 3. Portrait of the captured dataset

B. Constructing SherLock-FakeNewsNet Dataset

Next, before applying Natural Language Processing techniques, it was decided to clean the dataset. The NLTK toolkit [42] was utilized to remove punctuations, special characters and stop words from the text. Following that, to remove the noisy texts, square brackets and URLs the BeautifulSoup tool was employed.

C. Envisioning SherLock-FakeNewsNet Dataset

Subsequently, WordNet [43] was used to visualize the SherLock-FakeNewsNet dataset. Wordcloud format visualized the frequency of the words in the corpus. Figure 4 represents the portrait of the WordNet visualization for real text which is labelled as 1.

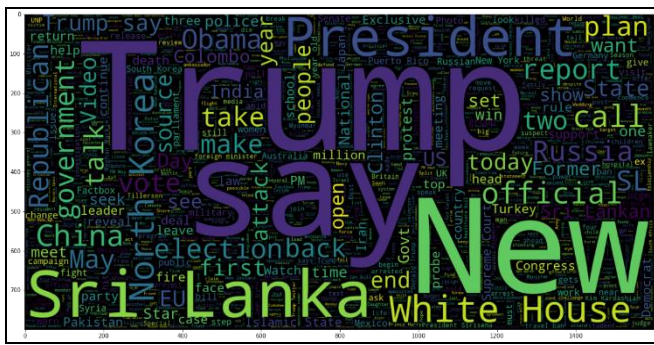


Fig. 4. Wordcloud for real text (label-1)

Figure 5 represents the portrait of the WordNet visualization for fake text which is labelled as 0.

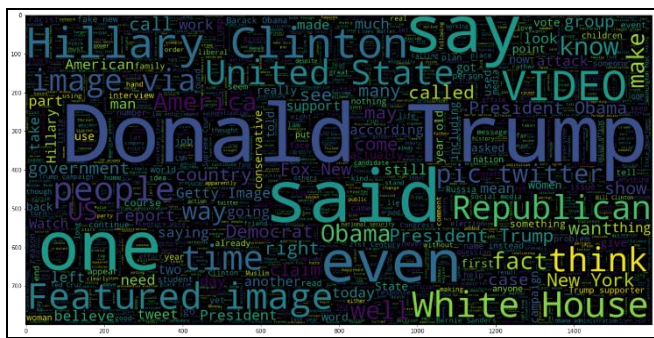


Fig. 5. Wordcloud for fake text (label-0)

D. Applying Natural Language Processing Techniques

To tokenize the dataset Keras tokenizer [44] was utilized. After that, to pre-process the tokenized dataset Keras pre-processing techniques such as text and sequence processing were applied. Finally, the text data was fitted and represented using global vectors of GloVe [45] word embeddings.

1) Tokenization:

Tokens are the fundamental forming blocks of NLP. Word Tokenization is the most used approach in NLP. In Tokenization, words are represented in tokens. Here, NLTK Tokenizer and Keras inbuilt Tokenizer were used to tokenize the text. ToktokTokenizer is a basic tokenizer that was used here which is intended to be quick and efficient, making it well-suited for large-scale NLP applications. It can tokenize text rapidly and accurately since it uses just simple rules based on whitespace and punctuation.

2) Stop Word Removal:

Stop words like “the”, “is”, “in” and “for” should be removed from the dataset to decrease the dataset size and decrease the time to train the model. Here, the same NLTK toolkit was used to remove stop words and punctuation from the text.

3) Word-Embeddings:

In NLP, words are matched to real numbers of vectors by mathematically embedding them into a continuous vector space with many dimensions per word. Word2Vec and GloVe have pre-trained Word-Embeddings for many languages including English. Here, GloVe word embeddings were used to map words to vectors.

4) Why GloVe Word-Embeddings:

GloVe stands for Global Vectors for Word Representation. It is formed on the matrix factorization techniques on the word-count matrix. GloVe word embeddings are used to easily represent the words in vectors in a large number of contexts. In real-world scenarios, it takes a long time to convert words into vectors. Owing to this, pre-trained word embeddings were used to accomplish this task. In GloVe, a large matrix of co-occurrence information is created, and it is based on the overall statistics of the word co-occurrence.

5) Why GloVe over Word2Vec:

GloVe has an advantage over Word2Vec as it considers the global context of words in a corpus rather than simply the local context around each word. GloVe does this by first building a co-occurrence matrix, which assesses how frequently each pair of words co-occurs in a corpus, and then factoring this matrix to generate the word embeddings. GloVe's global method enables it to record more complicated word associations, such as synonymy and antonymy. GloVe is also more computationally efficient than Word2Vec since it does not require a huge number of iterations over the training corpus. This facilitates training on a huge corpus and may result in shorter training periods.

E. Dividing the Dataset to Train and Test Sets

The proposed hybrid neural network was trained by using 90% of the dataset and retaining 10% for testing the resulting model.

1) Why 90:10 Train Test Split Ratio:

This ratio has gained popularity because it achieves a reasonable compromise between having enough data to efficiently train the model and having enough data to reliably assess the model's performance. The model may learn from more instances and develop a better understanding of the underlying patterns in the data with a bigger training set (90%). This can lead to improved model performance when tested on fresh, previously unknown data. A smaller testing set (10%), on the other hand, can assist guarantee that the model's performance is tested on a broad range of cases that it has never seen before. This can aid in identifying potential overfitting concerns when the model performs well on training data, but, badly on fresh data.

F. Proposed Hybrid Neural Network Architecture

The keras sequential neural network model is chosen as the architecture for the proposed hybrid approach. The first layer of the chosen keras sequential model consisted of an Embedding layer with the parameters such as embedding size and max features which are defined in Table V. Then, to remove a few contexts the Dropout layer is applied. After that, to extract higher-level characteristics, Conv1D layers and MaxPooling layers are added. Following that, to catch the context of the given input text and to capture interdependencies among word sequences two Long-Short-Term-Memory layers are utilized. In the last phases, Dropout layers are carefully selected to reduce overfitting and Dense layers are carefully selected to improve the efficacy of the hybrid model. Two activation functions Relu in Conv1D layers and Sigmoid in Dense layers are employed with the proposed hybrid model. With that, the proposed hybrid model experiments with various loss functions and optimizers. After evaluating the hyperparameter tuning Adam is examined as the proper optimizer, and the Binary Cross-Entropy loss function is explored as the proper loss function which is stated in Table IX. The structure of the suggested hybrid neural network is presented in the following tables. Table IV represents the structure of the hybrid neural network with layer type, output shape and parameters.

TABLE IV
STRUCTURE OF THE HYBRID NEURAL NETWORK

Layer Type	Output Shape	Parameters
Embedding	(None, None, 100)	1000000
Dropout	(None, None, 100)	0
Conv1D	(None, None, 128)	64128
MaxPooling1D	(None, None, 128)	0
LSTM	(None, None, 20)	11920
LSTM	(None, 20)	3280
Dropout	(None, 20)	0
Dense	(None, 512)	10752
Dropout	(None, 512)	0
Dense	(None, 256)	131328
Dense	(None, 1)	257

Table V represents the details of the defined parameters of the hybrid neural network with batch size, embedding size, max features, max length, and epochs.

TABLE V
DETAILS OF THE DEFINED PARAMETERS

Defined Parameter	Size of Parameter
Batch	256
Embedding	100
Max_Features	10000
Max_Length	300
Epochs	10

1) Why Convolutional Neural Network:

Convolutional Neural Networks (CNNs) are a type of neural network that was originally developed for computer vision tasks, such as image classification and object recognition. However, they have also been successfully applied to Natural Language Processing (NLP) tasks, such as text classification, sentiment analysis, and language modelling. One of the key strengths of CNNs is their ability to capture local structure within an input. In NLP, this can correspond to capturing n-grams, or sequences of adjacent words, which can be important for understanding the meaning of a sentence or document. Another significant feature of CNNs is their capacity to exchange parameters across many areas of the input. This can lower the number of parameters needed to train the model while also assisting the model in generalizing to new data. CNNs may also be resilient to variations in input length, which can be very advantageous for NLP tasks where the length of the input might vary greatly. CNNs can benefit from pre-training on vast amounts of unlabelled data, which can assist enhance their performance on downstream NLP tasks. Overall, while traditional NLP methods such as Bag-of-Words and n-gram models have been effective for many tasks, CNNs can offer a powerful and flexible alternative that can capture local structure, share parameters, and generalize well to new data. Figure 6 represents the detailed process and deep neural network architecture of the proposed hybrid approach.

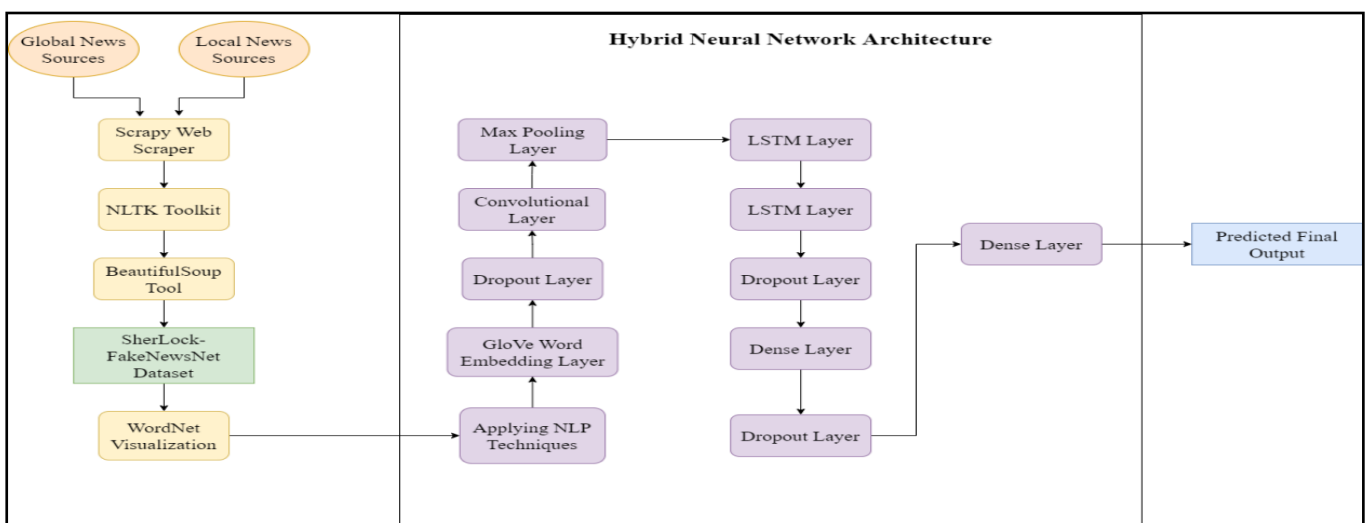


Fig. 6: Detailed methodology architecture

2) Why Recurrent Neural Network:

Recurrent Neural Networks (RNNs) are a type of neural network that has been widely used in Natural Language Processing (NLP) tasks, such as language modelling, machine translation, sentiment analysis, and speech recognition. RNNs are made to analyze sequential data, like language, where a word's meaning depends on the context of the words that came before it. The meaning of a sentence or document might rely on the context of the words that came before it, making them well-suited for NLP tasks. RNNs have a "memory" that enables them to recognize input dependencies over a lengthy period. This is especially helpful for NLP tasks as the meaning of a sentence might vary depending on its whole input history. For NLP tasks where the input length might vary significantly, the ability of RNNs to handle variable-length inputs is crucial. Overall, RNNs are a powerful tool for NLP tasks that involve sequential data. They can handle variable-length inputs, capture long-term dependencies, and can be pre-trained on large amounts of data, making them a versatile and effective choice for a wide range of NLP tasks.

3) Why Long Short-Term Memory Neural Network:

LSTM (Long Short-Term Memory) is a type of Recurrent Neural Network (RNN) that has been widely used in Natural Language Processing (NLP) tasks, such as language modelling, machine translation, sentiment analysis, and speech recognition. In NLP applications where the meaning of a phrase might rely on the complete history of the input, the ability of LSTMs to manage long-term dependencies is crucial. In order to capture long-term dependencies without being impacted by the vanishing gradient problem, LSTMs use a memory cell that may selectively add or delete information over time. The ability of LSTMs - can handle variable-length inputs - is crucial for NLP applications where the input length might change significantly. LSTMs are also robust to noise and missing data, making them well-suited for NLP tasks where the input may be noisy or incomplete. Overall, LSTMs are a powerful tool for NLP tasks that involve sequential data with long-term dependencies. They can handle variable-length inputs, are robust to noise, and can be pre-trained on large amounts of data, making them a versatile and effective choice for a wide range of NLP tasks.

IV. RESULTS AND EVALUATION

By dividing the dataset into 90% train size and 10% test size the proposed hybrid neural network is trained. For training and testing sets accuracy rates of 94% and 91.37% are accomplished respectively with Adam optimizer and Binary-Cross Entropy loss function after 10 epochs. The proposed hybrid model is trained using Google Colab and Kaggle Kernels. Following that, Kaggle TPUs and TensorFlow TPUs are used to run the model.

1) Why TensorFlow:

TensorFlow is an open-source machine learning library that can do various tasks related to deep learning. TensorFlow can run on multiple CPUs, GPUs and TPUs. TensorFlow helps to build machine learning and deep learning models easily with high-level APIs like Keras with eager execution. Also, it helps to train and deploy models in the cloud, on the device, on-prem and in the browser.

2) Why Google Colab and Kaggle Kernels:

Google Colab and Kaggle Kernels are used as the platforms to run the proposed deep learning model. TensorFlow TPUs and Kaggle TPUs are used to execute the model. Execution time is more than one hour when using TPUs rather than GPUs, because, TPUs are more powerful and faster than GPUs and CPUs. Kaggle Kernel and Google Colab kernel both support Jupyter notebooks, the notebooks can be saved easily with Kaggle and Google Colab cloud-hosted environments. Kaggle and Google Colab provide many in-built datasets for data science-related tasks and those datasets can be easily integrated into the projects as well.

A. Comparing Results with Different Models

First, to do some experiments two different models namely, Convolutional Neural Network (CNN) model and the Recurrent Neural Network (RNN) model with Long Short-Term Memory (LSTM) model are selected. After that, the proposed hybrid model was chosen to experiment with the dataset. From these three models, the proposed hybrid model accomplished the highest rate of accuracy which is 92%.

Table VI indicates the results for different models. It further elaborates on the accuracy and loss values of each model.

TABLE VI
RESULTS FOR DIFFERENT MODELS

Comparison of Results	Model		
	CNN Model	RNN Model	Hybrid Model
Accuracy	0.9031	0.9061	0.9204
Loss	0.2261	0.1589	0.1889

B. Evaluation Results of the Proposed Hybrid Model

To evaluate the performance of the proposed hybrid model, compute the Precision, Recall and F1-Score. Precision scores of 0.92 and 0.91, Recall scores of 0.90 and 0.93 and F1-Scores of 0.91 and 0.92 are acquired for fake and not fake classes.

Table VII displays the Precision, Recall, and F1-Score results as well as the testing set's Accuracy, Macro Avg and Weighted Avg.

TABLE VII
EVALUATION RESULTS OF THE PROPOSED HYBRID MODEL

	Precision	Recall	F1-Score	Support
Fake	0.92	0.90	0.91	10443
Not Fake	0.91	0.93	0.92	11226
Accuracy			0.91	21669
Macro Avg	0.91	0.91	0.91	21669
Weighted Avg	0.91	0.91	0.91	21669

According to Table VII, the Recall scores of 0.90 and 0.93 are achieved when assessing the results. It improves the subtlety of the proposed hybrid model.

F1-score is the weighted average of Precision and Recall. The results in table VII demonstrate that scores of Precision and Recall are relatively similar. As a result, the F1-Scores of 0.91 and 0.92 for fake and not fake sets exhibit the proposed hybrid model achieves a good percentage.

In scikit-learn, the Confusion matrix [46] is adapted to generate the classification report.

Table VIII demonstrates the confusion matrix of the testing dataset. In the testing dataset, the confusion matrix examines four distinct combinations.

TABLE VIII
TESTING DATASET CONFUSION MATRIX

	Positive (1)	Negative (0)
Positive (1)	9404	1039
Negative (0)	831	10395

C. Evaluation Results of Training and Testing Sets

Following the execution of the neural network, the Matplotlib [47] library is chosen to produce graphs and display the data. Matplotlib is the ideal tool to represent the outcomes of neural networks. By dividing the dataset into 90% train size and 10% test size the proposed hybrid neural network is trained. After executing the proposed hybrid model, figure 7 and figure 8 demonstrate the Matplotlib reflection of training and testing sets. The accuracy and loss scores of 0.9204 and 0.1889, 0.9137 and 0.2147 are achieved for the training set and the testing set respectively.

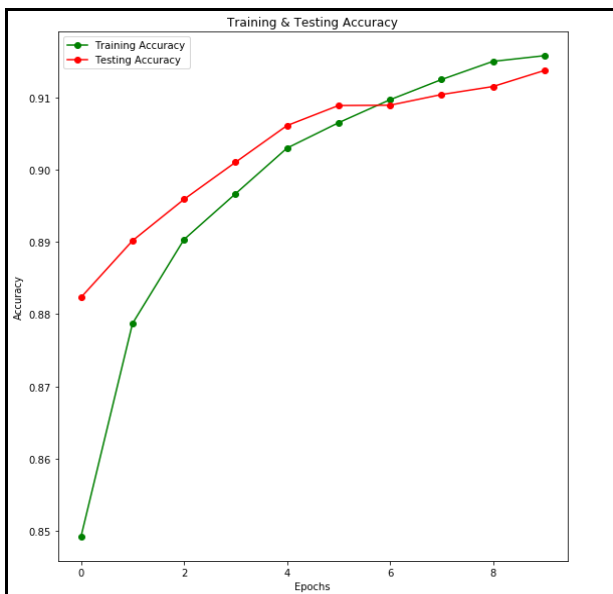


Fig. 7. Matplotlib reflection of results for training and testing accuracy

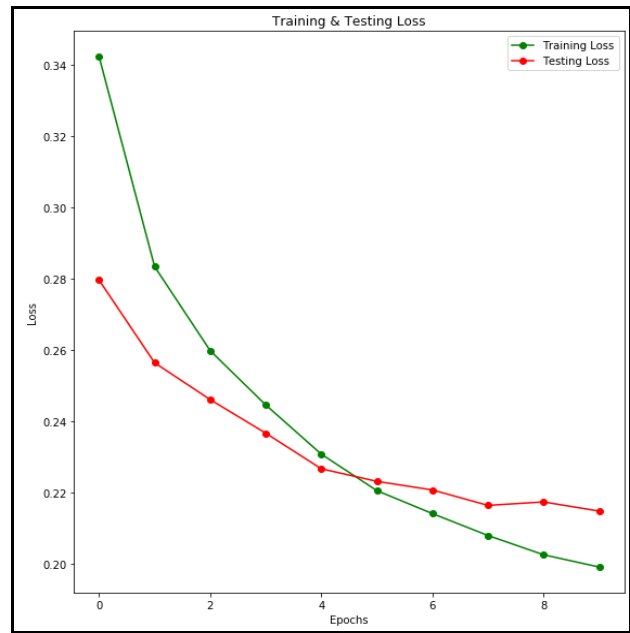


Fig. 8. Matplotlib reflection of results for training and testing loss

D. Hyperparameter Tuning for the Hybrid Model

Hyperparameter tuning for the proposed hybrid model is achieved after 10 epochs with the loss function as Binary Cross-Entropy. The above loss function has been chosen due to the two classes of the ‘SherLock-FakeNewsNet’ dataset. Table IX represents the results of different optimizers.

TABLE IX
RESULTS FOR DIFFERENT OPTIMIZERS

Optimizer	Accuracy	Loss
Adadelta	0.7229	0.5602
Adagrad	0.8560	0.3278
Adam	0.9204	0.1889
RMSprop	0.9161	0.1969
SGD	0.8560	0.3366

By observing results displayed in Table IX, Adam is chosen as the best optimizer that fit with the other hyperparameters of the suggested hybrid neural network.

E. Experimental Results of the Hybrid Model

Some related datasets in the stance-based fake news detection area were chosen to experiment with the proposed hybrid model using the same experimental setup used for ‘SherLock-FakeNewsNet’ dataset.

1) Experiment 01 Results:

As for the first experiment, the proposed hybrid model is checked and evaluated with the ‘FakeNewsNet’ [48] dataset and following Table X represents the results of the experiment.

TABLE X
EXPERIMENT 01 RESULTS FROM THE PROPOSED HYBRID MODEL

	Precision	Recall	F1-Score	Support
Fake	0.91	0.83	0.87	2958
Not Fake	0.81	0.90	0.85	3405
Accuracy			0.87	6363
Macro Avg	0.86	0.87	0.86	6363
Weighted Avg	0.87	0.87	0.87	6363

The precision score of 0.86 indicates that out of all the news articles that were classified as fake, 86% were fake. The recall score of 0.88 indicates that out of all the fake news articles in the dataset, the model correctly identified 88% of them. The F1-Score of 0.87 represents the weighted average of the precision and recall scores, taking both into account. The support value of 6363 indicates the number of samples in the dataset used to calculate these metrics. The accuracy of 0.87 indicates the proportion of correctly classified samples out of all the samples in the dataset.

2) Experiment 02 Results:

As for the second experiment, the proposed hybrid model is checked and evaluated with the "FA-KES" [49] dataset and Table XI represents the results of the experiment.

TABLE XI
EXPERIMENT 02 RESULTS FROM THE PROPOSED HYBRID MODEL

	Precision	Recall	F1-Score	Support
Fake	0.84	0.92	0.88	938
Not Fake	0.90	0.92	0.91	1042
Accuracy			0.88	1980
Macro Avg	0.87	0.92	0.89	1980
Weighted Avg	0.88	0.88	0.88	1980

The precision score of 0.87 indicates that out of all the news articles that were classified as fake, 87% were fake. The recall score of 0.92 indicates that out of all the fake news articles in the dataset, the model correctly identified 92% of them. The F1-Score of 0.89 represents the weighted average of the precision and recall scores, taking both into account. The support value of 1980 indicates the number of samples in the dataset used to calculate these metrics. The accuracy of 0.88 indicates the proportion of correctly classified samples out of all the samples in the dataset.

3) Experiment 03 Results:

As for the third experiment, the proposed hybrid model is checked and evaluated with the "LIAR" [50] dataset and Table XII represents the results of the experiment.

TABLE XII
EXPERIMENT 03 RESULTS FROM THE PROPOSED HYBRID MODEL

	Precision	Recall	F1-Score	Support
Fake	0.80	0.87	0.83	212
Not Fake	0.90	0.85	0.87	1072
Accuracy			0.86	1284
Macro Avg	0.85	0.86	0.85	1284
Weighted Avg	0.87	0.86	0.86	1284

The precision score of 0.88 indicates that out of all the news articles that were classified as fake, 88% were fake. The recall score of 0.84 indicates that out of all the fake news articles in the dataset, the model correctly identified 84% of them. The F1-Score of 0.86 represents the weighted average of the precision and recall scores, taking both into account. The support value of 1284 indicates the number of samples in the dataset used to calculate these metrics.

4) Experiment 04 Results:

As for the fourth experiment, the proposed hybrid model is checked and evaluated with the "ISOT" [51] dataset and Table XIII represents the results of the experiment.

TABLE XIII
EXPERIMENT 04 RESULTS FROM THE PROPOSED HYBRID MODEL

	Precision	Recall	F1-Score	Support
Fake	0.81	0.87	0.84	258
Not Fake	0.92	0.86	0.89	1611
Accuracy			0.87	1869
Macro Avg	0.86	0.86	0.86	1869
Weighted Avg	0.88	0.87	0.87	1869

The precision score of 0.89 indicates that out of all the news articles that were classified as fake, 89% were fake. The recall score of 0.85 indicates that out of all the fake news articles in the dataset, the model correctly identified 85% of them. The F1-Score of 0.87 represents the weighted average of the precision and recall scores, taking both into account. The support value of 1869 indicates the number of samples in the dataset used to calculate these metrics.

5) Experiment 05 Results:

As for the fifth experiment, the proposed hybrid model is checked and evaluated with the "FNC-1" [52] dataset and Table XIV represents the results of the experiment.

TABLE XIV
EXPERIMENT 05 RESULTS FROM THE PROPOSED HYBRID MODEL

	Precision	Recall	F1-Score	Support
Fake	0.88	0.89	0.88	21417
Not Fake	0.86	0.85	0.85	19045
Accuracy			0.87	40462
Macro Avg	0.87	0.87	0.87	40462
Weighted Avg	0.87	0.87	0.87	40462

The precision score of 0.87 indicates that out of all the news articles that were classified as fake, 87% were fake. The recall score of 0.87 indicates that out of all the fake news articles in the dataset, the model correctly identified 87% of them. The F1-Score of 0.87 represents the weighted average of the precision and recall scores, taking both into account. The support value of 40462 indicates the number of samples in the dataset used to calculate these metrics.

Based on the results obtained from the five experiments conducted on different datasets, it can be concluded that the proposed hybrid CNN and RNN-LSTM model is effective for detecting fake news. The precision scores ranged from 0.86 to 0.89, indicating that the model was able to correctly identify a high percentage of fake news articles. The recall scores ranged from 0.84 to 0.92, indicating that the model was able to correctly identify a high percentage of the actual fake news articles in the dataset. The F1-Scores ranged from 0.86 to 0.89, which indicates that the model was able to balance both precision and recall in identifying fake news articles. The overall accuracy ranged from 0.87 to 0.88, indicating that the model was able to correctly classify a high percentage of samples in the dataset. Therefore, it can be concluded that the proposed hybrid CNN and RNN-LSTM model is a reliable and effective approach for detecting fake news in various datasets. However, further research can be done to explore the potential of this model on more diverse and larger datasets.

V. CONCLUSION AND FUTURE WORK

According to the results obtained from the three different models namely, CNN model, RNN model, and Hybrid model, it can be concluded that after 10 epochs with the Adam optimizer and Binary Cross-Entropy loss function from the proposed hybrid neural network architecture, 92% accuracy was achieved when compared with the available datasets and research works carried out in the stance-based fake news detection. As a result, it can be concluded that the proposed Hybrid approach which is based on CNN, and RNN-LSTM models achieve the highest accuracy rate of 92% by outperforming most of the conventional models today.

Based on the results of five experiments, which are conducted from the proposed hybrid model, we can conclude that this hybrid approach yields the highest Precision, Recall and F1-Score, because different types of messages such as lengthy and short messages also included in the training dataset. In addition, the testing dataset also

included a variety of messages. The hybrid approach chose mainly focusing on CNNs to extract the higher-level characteristics from the input text. Subsequently, RNN-LSTMs chose to capture the long-term dependencies of the word sequences. Finally, Dropout layers are carefully selected to reduce overfitting, and Dense layers are carefully selected to enhance the efficiency of the hybrid model.

From the tables, Table X, Table XII, Table XIII, and Table XIV, based on the results of the five experiments conducted on different datasets, it can be concluded that the proposed hybrid CNN and RNN-LSTM model performs well in detecting fake news articles. The precision scores ranging from 0.86 to 0.89 indicate that a high percentage of articles classified as fake by the model were fake. The recall scores ranging from 0.84 to 0.92 indicate that the model correctly identified a high percentage of fake news articles present in the dataset. The F1-Scores ranging from 0.86 to 0.89 represent the weighted average of the precision and recall scores, taking both into account. Additionally, the accuracy scores ranging from 0.87 to 0.88 indicate that the model correctly classified a high proportion of samples in the dataset. Overall, the results suggest that the proposed hybrid model can generalize well across different datasets and perform effectively in detecting fake news articles.

Since this paper considered only the mechanisms of Deep Learning and the mechanisms of Natural Language Processing to identify stance-based fake news, it is an open area to apply different techniques from supervised learning, semi-supervised learning, and unsupervised learning to identify stance-based fake news.

In terms of future developments, the proposed hybrid model will experiment with new methods such as Transformers, Auto-Encoders, Federated Learning APIs and Graph Neural Networks. Finally, this research study will be continued as a public Kaggle Notebook to perform new experiments from time to time.

VI. DATA AVAILABILITY

The dataset can be found at:

<https://www.kaggle.com/pathumveyron24/sherlockfakenewsnetorigins>

ACKNOWLEDGMENT

The researchers of this work would like to appreciate many persons who helped in numerous ways to form this work a success.

REFERENCES

- [1] S. Afroz, M. Brennan, and R. Greenstadt, "Detecting Hoaxes, Frauds, and Deception in Writing Style Online," in *2012 IEEE Symposium on Security and Privacy*, May 2012, pp. 461–475, doi: 10.1109/SP.2012.34.
- [2] H. Allcott and M. Gentzkow, "Social Media and Fake News in the 2016 Election," 2017, doi: 10.1257/JEP.31.2.211.
- [3] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake News Detection on Social Media: A Data Mining Perspective," *SIGKDD Explor News*, vol. 19, no. 1, pp. 22–36, Sep. 2017, doi: 10.1145/3137597.3137600.
- [4] M. D. P. P. Goonathilake and P. P. N. V. Kumara, "CNN, RNN-LSTM Based Hybrid Approach to Detect State-of-the-Art Stance-Based Fake News on Social Media," 2020 20th International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, Sri Lanka, 2020, pp. 23–28, doi: 10.1109/ICTer51097.2020.9325477.

- [5] I. Katsirea, “‘Fake news’: reconsidering the value of untruthful expression in the face of regulatory uncertainty,” *J. Media Law*, vol. 10, no. 2, pp. 159–188, Jul. 2018, doi: 10.1080/17577632.2019.1573569.
- [6] The Verge, “Your short attention span could help fake news spread”, 2020.[Online].Available:https://www.theverge.com/2017/6/26/15875488/fake-news-viral-hoaxes-bots-information-overload-twitter-facebook-social-media. [Accessed: 12-Sep-2021].
- [7] Conspiracy Theories, “The ‘Pizzagate’ Conspiracy Theory - Conspiracy Theories”, 2021. [Online]. Available: https://conspiracytheories.in/the-pizzagate-conspiracy-theory/. [Accessed: 12-Sep-2021].
- [8] S. B. Parikh and P. K. Atrey, “Media-Rich Fake News Detection: A Survey,” in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, Apr. 2018, pp. 436–441, doi: 10.1109/MIPR.2018.00093.
- [9] Nytimes.com, ‘Sri Lanka Blocks Social Media, Fearing More Violence (Published 2019)’, 2021. [Online]. Available: https://www.nytimes.com/2019/04/21/world/asia/sri-lanka-social-media.html.[Accessed:11-Sep-2021].
- [10] L. Papadopoulos, L. Papadopoulos, C. McFadden and C. McFadden, “These Are the 10 Most-viewed Fake News Stories of 2019 on Facebook”, *Interestingengineering.com*, 2021. [Online]. Available: https://interestingengineering.com/these-are-the-10-most-viewed-fake-news-stories-of-2019-on-facebook. [Accessed: 12-Sep-2021].
- [11] K. Shu *et al.*, *Detecting Fake News on Social Media*. Morgan & Claypool, 2019.
- [12] R. Oshikawa, J. Qian, and W. Y. Wang, “A Survey on Natural Language Processing for Fake News Detection,” *ArXiv*, vol. abs/1811.00770, 2018.
- [13] J. Fürnkranz, “A Study Using n-gram Features for Text Categorization”, *Citeseerx.ist.psu.edu*,2021.[Online].Available:http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.131.3126. [Accessed: 12-Sep-2021].
- [14] Dl.acm.org, ‘Document clustering based on non-negative matrix factorization | Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval’,2021.[Online].Available:https://dl.acm.org/citation.cfm?id=860485. [Accessed: 12-Sep-2021].
- [15] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and Their Compositionality,” in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, USA, 2013, pp. 3111–3119, Accessed: Sep. 12, 2021. [Online]. Available: http://dl.acm.org/citation.cfm?id=2999792.2999959.
- [16] H. Karimi and J. Tang, “Learning Hierarchical Discourse-level Structure for Fake News Detection,” *ArXiv Prepr. ArXiv190307389*, 2019.
- [17] Ieeexplore.ieee.org, ‘Classifying Fake News Articles Using Natural Language Processing to Identify In-Article Attribution as a Supervised Learning Estimator - IEEE Conference Publication’, 2021.[Online].Available:https://ieeexplore.ieee.org/document/866553. [Accessed: 12-Sep-2021].
- [18] ‘A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification | Expert Systems with Applications: An International Journal’, *Dl.acm.org*,2021.[Online].Available:https://dl.acm.org/doi/10.1016/j.eswa.2016.06.005. [Accessed: 12-Sep-2021].
- [19] ‘A feature selection model based on genetic rank aggregation for text sentiment classification - Aytuğ Onan, Serdar Korukoğlu, 2017’, *SAGEJournals*,2021.[Online].Available:https://journals.sagepub.com/doi/10.1177/0165551515613226. [Accessed: 12-Sep-2021].
- [20] A. Onan, “Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks”, 2021.
- [21] ‘A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification’, *Ieeexplore.ieee.org*,2021.[Online].Available:https://ieeexplore.ieee.org/document/9316208. [Accessed: 12-Sep-2021].
- [22] A. Onan, “Topic-Enriched Word Embeddings for Sarcasm Identification,” in *Software Engineering Methods in Intelligent Algorithms*, Cham, 2019, pp. 293–304.
- [23] A. ONAN, "Sentiment analysis on massive open online course evaluations: A text mining and deep learning approach", 2021.
- [24] M. A. Toçoğlu and A. Onan, “Sentiment Analysis on Students’ Evaluation of Higher Educational Institutions,” in *Intelligent and Fuzzy Techniques: Smart and Innovative Solutions*, Cham, 2021, pp. 1693–1700.
- [25] A. Onan and S. Korukoğlu, “Exploring Performance of Instance Selection Methods in Text Sentiment Classification,” in *Artificial Intelligence Perspectives in Intelligent Systems*, Cham, 2016, pp. 167–179.
- [26] ‘Ensemble of keyword extraction methods and classifiers in text classification | Expert Systems with Applications: An International Journal’, *Dl.acm.org*, 2021. [Online]. Available: https://dl.acm.org/doi/10.1016/j.eswa.2016.03.045. [Accessed: 12-Sep-2021].
- [27] ‘An ensemble scheme based on language function analysis and feature engineering for text genre classification - Aytuğ Onan, 2018’, *SAGEJournals*,2021.[Online].Available:https://doi.org/10.1177/0165551516677911. [Accessed: 12-Sep-2021].
- [28] ‘Hybrid supervised clustering based ensemble scheme for text classification | Emerald Insight’, *Emerald.com*,2021. [Online]. Available: https://www.emerald.com/insight/content/doi/10.1108/K-10-2016-0300/full/html. [Accessed: 12-Sep-2021].
- [29] ‘A hybrid ensemble pruning approach based on consensus clustering and multi-objective evolutionary algorithm for sentiment classification | Information Processing and Management: an International Journal’, *Dl.acm.org*, 2021. [Online]. Available: https://dl.acm.org/doi/10.1016/j.ipm.2017.02.008. [Accessed: 12-Sep-2021].
- [30] A. Onan and M. A. Tocoglu, “Satire identification in Turkish news articles based on ensemble of classifiers,” p. 21.
- [31] R. K. Kaliyar, “Fake News Detection Using A Deep Neural Network,” in *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, Dec. 2018, pp. 1–7, doi: 10.1109/CCAA.2018.8777343.
- [32] L. Wu and H. Liu, “Tracing Fake-News Footprints: Characterizing Social Media Messages by How They Propagate,” in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, New York, NY, USA, 2018, pp. 637–645, doi: 10.1145/3159652.3159677.
- [33] C. Shao, G. L. Ciampaglia, A. Flammini, and F. Menczer, “Hoaxy: A Platform for Tracking Online Misinformation,” in *Proceedings of the 25th International Conference Companion on World Wide Web*, Republic and Canton of Geneva, Switzerland, 2016, pp. 745–750, doi: 10.1145/2872518.2890098.
- [34] K. Shu, D. Mahudeswaran, and H. Liu, “FakeNewsTracker: a tool for fake news collection, detection, and visualization,” *Comput. Math. Organ. Theory*, vol. 25, no. 1, pp. 60–71, Mar. 2019, doi: 10.1007/s10588-018-09280-3.
- [35] L. Cui, K. Shu, S. Wang, D. Lee and H. Liu, “dEFEND”, *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2021. Available: 10.1145/3357384.3357862 [Accessed: 12-Sep-2021].
- [36] “Real-Time News Certification System on Sina Weibo.” https://dl.acm.org/citation.cfm?id=2742571 (accessed Sep. 12, 2021).
- [37] “Fake News Patterns Detector.” http://fakenews.mit.edu/ (accessed Sep. 12, 2021).
- [38] N. J. Conroy, V. L. Rubin, and Y. Chen, “Automatic deception detection: Methods for finding fake news,” *Proc. Assoc. Inf. Sci. Technol.*, vol. 52, no. 1, pp. 1–4, 2015, doi: 10.1002/pr2.2015.145052010082.
- [39] M. Goonathilake and P. Kumara, "SherLock: A CNN, RNN-LSTM Based Mobile Platform for Fact-Checking on Social Media", *Ir.kdu.ac.lk*, 2021. [Online]. Available: http://ir.kdu.ac.lk/handle/345/2901. [Accessed: 12-Sep-2021].
- [40] KLTN Perera, PPNV Kumara, and B Hettige, “Intelligent News Reader.” 10th Annual International Research Conference of General Sir John Kotelawala Defence University, Mar. 2018, [Online]. Available:https://www.researchgate.net/publication/323986168_Intelligent_News_Reader.
- [41] Scrapy.org, ‘Scrapy | A Fast and Powerful Scraping and Web Crawling Framework’,2021. [Online]. Available: https://scrapy.org/. [Accessed: 12-Sep-2021].
- [42] Nltk.org, ‘Natural Language Toolkit — NLTK 3.6.2 documentation’, 2021. [Online]. Available: https://www.nltk.org/. [Accessed: 12-Sep-2021].
- [43] Nltk.org, ‘WordNet Interface’, 2021. [Online]. Available: https://www.nltk.org/howto/wordnet.html.[Accessed: 12-Sep-2021]
- [44] TensorFlow, ‘The Sequential model |TensorFlow Core’, 2021. [Online].Available:https://www.tensorflow.org/guide/keras. [Accessed: 12-Sep-2021].
- [45] J. Pennington, “GloVe: Global Vectors for Word Representation”, *Nlp.stanford.edu*, 2021. [Online]. Available: https://nlp.stanford.edu/projects/glove/. [Accessed: 12-Sep-2021].
- [46] Scikit-learn.org, ‘Confusion matrix — scikit-learn 0.24 documentation’,2021.[Online].Available:https://scikitlearn.org/stable

- [/auto_examples/model_selection/plot_confusion_matrix.html](#). [Accessed: 12-Sep-2021].
- [47] Matplotlib.org, 'Matplotlib: Python plotting — Matplotlib 3.4.3 documentation', 2021. [Online]. Available: <https://matplotlib.org/>. [Accessed: 12-Sep-2021].
- [48] KaiDMML, "KaiDMML/FakeNewsNet: This is a dataset for fake news detection research," *GitHub*. [Online]. Available: <https://github.com/KaiDMML/FakeNewsNet>. [Accessed: 22-Mar-2023].
- [49] F. K. A. Salem, R. A. Feel, S. Elbassuoni, M. Jaber, and M. Farah, "FA-Kes: A fake news dataset around the Syrian War," *Proceedings of the International AAAI Conference on Web and Social Media*. [Online]. Available: <https://ojs.aaai.org/index.php/ICWSM/article/view/3254>. [Accessed: 22-Mar-2023].
- [50] "Papers with code - liar dataset," *Dataset | Papers With Code*. [Online]. Available: <https://paperswithcode.com/dataset/liar#:~:text=Liar%20is%20a%20publicly%20available, fact%2Dchecking%20research%20as%20well>. [Accessed: 22-Mar-2023].
- [51] "Papers with code - isot fake news dataset dataset," *Dataset | Papers With Code*. [Online]. Available: <https://paperswithcode.com/dataset/isot-fake-news-dataset>. [Accessed: 22-Mar-2023].
- [52] "Papers with code - FNC-1 dataset," *FNC-1 Dataset | Papers With Code*. [Online]. Available: <https://paperswithcode.com/dataset/fnc-1>. [Accessed: 22-Mar-2023].