Fake News Detection using Deep Learning and Natural Language Processing

B. Tech. Project Presentation

By

Vishal Sharma

MIS :111903159

Under the Supervision of

Prof. Archana B. Patil

6th May 2023

Overview

Overview

1. Introduction

1.1. Intro

- Fake news, defined as false information deliberately spread with the intent to mislead.
- There is a growing need for effective techniques to combat the spread of fake news.
- Traditional techniques rely on human experts to identify fake news.
- There has been an increasing interest in developing automated techniques to detect fake news.
- Deep Learning enables machines to learn and make decisions without human intervention.
- It has gained popularity due to its ability to extract meaningful features from raw data.

Therefore, it is reasonable to explore the possibility of using deep learning for fake news detection.

1. Introduction

1.2. Motivation

To prevent the spread of misinformation
False information can have harmful effects on individuals and society as a whole.

To preserve the integrity of journalism
Fake news undermines the credibility of news sources and journalists.

To protect democracy
Fake news can be used to manipulate public opinion and influence elections.

2. Literature Review and Research Gaps

2.1. Literature Review

Paper Title	Year	Observation
Supervised Learning for fake news Detection	2019	• Performed experiments using different classifiers such as Naive Bayes, Support vector Machines and Random Forest. • Also evaluated the impact of different features such as lexical and syntactical features in the classification performance.
An ensemble machine learning approach through effective feature extraction to classify fake news.	2021	• Proposed ensemble model comprising of popular ML models such as Decision tree, Random Forest and extra tree classifiers.

2. Literature Review and Research Gaps

2.1. Literature Review

Paper Title	Year	Observations
Fake News detection using Convolutional Neural Networks and Random Forest.	2021	• Proposed model used a hybrid approach using CNN and Random Forest. • One model focusing on image-based news and other on text based news.
Fake News Detection : A hybrid CNN-RNN based deep learning approach.	2021	• Proposed model make use of the ability of CNN to extract local features and of the LSTM to learn long term dependencies.

2. Literature Review and Research Gaps

2.1. Literature Review

Paper Title	Year	Observations
Sentiment Analysis for fake news detection by means of neural networks.	2020	• Proposed a methodology for fake news detection using sentiment analysis and neural networks. • Used bag-of-words approach to extract features from preprocessed news articles and performed sentiment analysis using VADER tool.(Valence Aware Dictionary and Sentiment Reasoner)

2. Literature Review and Research Gaps

2.2. Research Gaps

There has been significant research in developing automatic fake news detection systems. However, these systems still face several gaps and limitations.

- Lack of training data.
- Sensitivity to the source.
- Limited contextual understanding.

3. Problem Statement and Objectives

3.1. Problem Statement

“The goal is to create a model that can accurately and efficiently differentiate between genuine news articles and those that are intentionally misleading or false, in order to prevent the spread of misinformation and promote the dissemination of accurate and reliable information which involves analyzing the content and context of the article, as well as the sources and references cited within it, to determine its credibility and accuracy.”

3. Problem Statement and Objectives

3.2. Objectives

- Accurately identify and classify fake news.
- Improve the efficiency and speed of detection.
- Handle large-scale datasets.
- Reduce the impact of fake news.

4. Methodology

4.1. Data Collection.

- The dataset we use is the ISOT Fake News dataset introduced by ISOT Research Lab.
- The original form of the dataset is two CSV files containing fake and real news respectively.
- The distribution of the data in the training, validation, and test sets is shown in Table 1 for the original combined dataset, which contains 44,898 data points.

Table 4.1: Distribution of Data.

Training	validation	Testing
64%	16%	20%
28734	7184	8980

4. Methodology

4.2. Data Preprocessing

The major objective of this section is to preprocess the input data using NLP techniques in order to set up the subsequent stage of feature extraction. To remove the noise from the news dataset, we constructed a preprocessing pipeline for each statement.
The following components make up the preprocessing pipeline :

- Whitespace was used in place of characters that are not in the range of a to z or A to Z.
- Changed all of the characters to lowercase.
- Lemmatization.
- Stopwords removal.

4. Methodology

4.3. Word Embedding

This step is critical because we must transform the dataset into a format that models can use. We developed a tokenizer to divide the text into individual words and generate token sequences that can be fed into LSTM, Bidirectional LSTM, and CNN-Bi LSTM models.

- Glove(Global Vectors for Word Representation) Embedding .
- FastText Word Embedding.

4. Methodology

4.3. Word Embedding

4.3.1. Glove(Global Vectors for Word Representation) Embedding

- Consists of Global vectors created by Stanford University.
- Glove files are simple text files in the form of dictionary.
- Four variable available in Glove are 50d,100d,200d and 300d.
- While applying it to words first we need to create Vocabulary Dictionary.
- Then we constructs a co-occurrence matrix that counts the number of times each word appears in the context of every other word in a given corpus.

4. Methodology

4.3. Word Embedding

4.3.2. FastText Word Embedding

- It breaks down each word in a given corpus into its constituent subwords, which are typically character n-grams.
- Each subword is represented as a vector, which is randomly initialized.
- These subword vectors are then combined to create a vector representation for each word in the corpus.
- The resulting vector representation for each word is essentially a weighted average of the subword vectors.

4. Methodology

4.4. Proposed Model

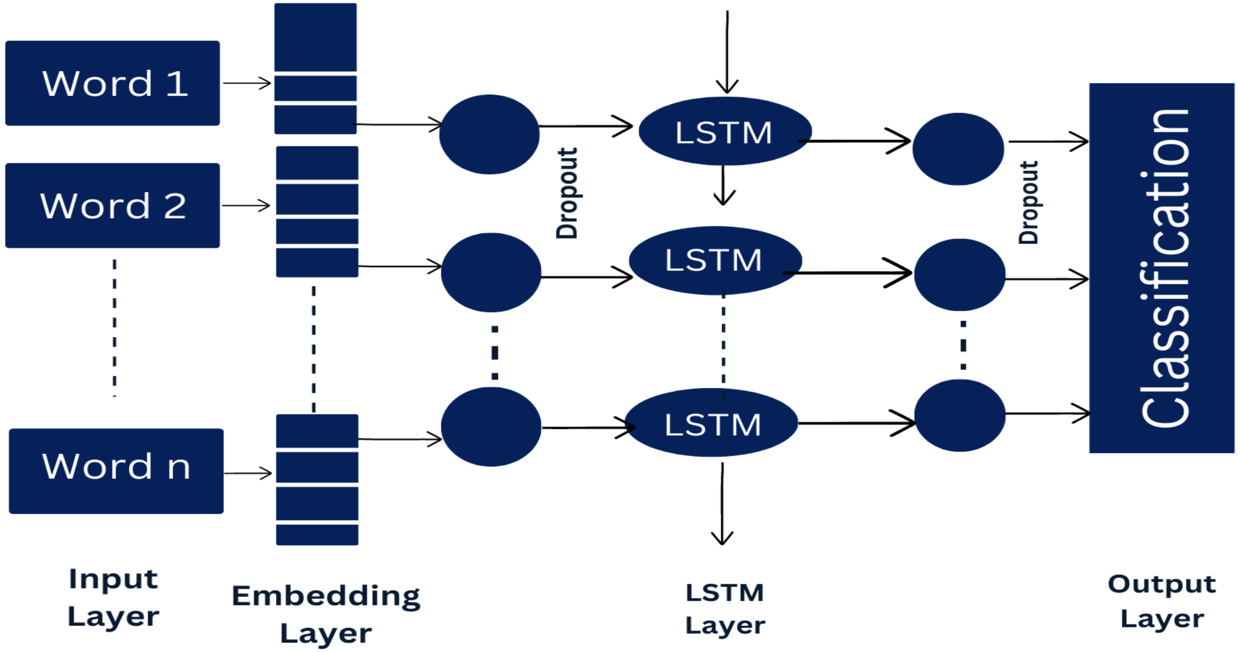
4.4.1. LSTM

- LSTM models consist of multiple LSTM cells, each of which has a set of learnable parameters that are updated during training.
- Each LSTM cell takes as input the current input vector and the output from the previous LSTM cell in the sequence.
- The LSTM cell computes a set of gates that control the flow of information through the cell, including a forget gate, an input gate, and an output gate.
- The forget gate determines which information from the previous hidden state should be forgotten.
- The input gate determines which new information should be added to the current hidden state.
- The output gate determines which information from the current hidden state should be output to the next LSTM cell in the sequence.

4. Methodology

4.4. Proposed Model

4.4.1. LSTM



4. Methodology

4.4. Proposed Model

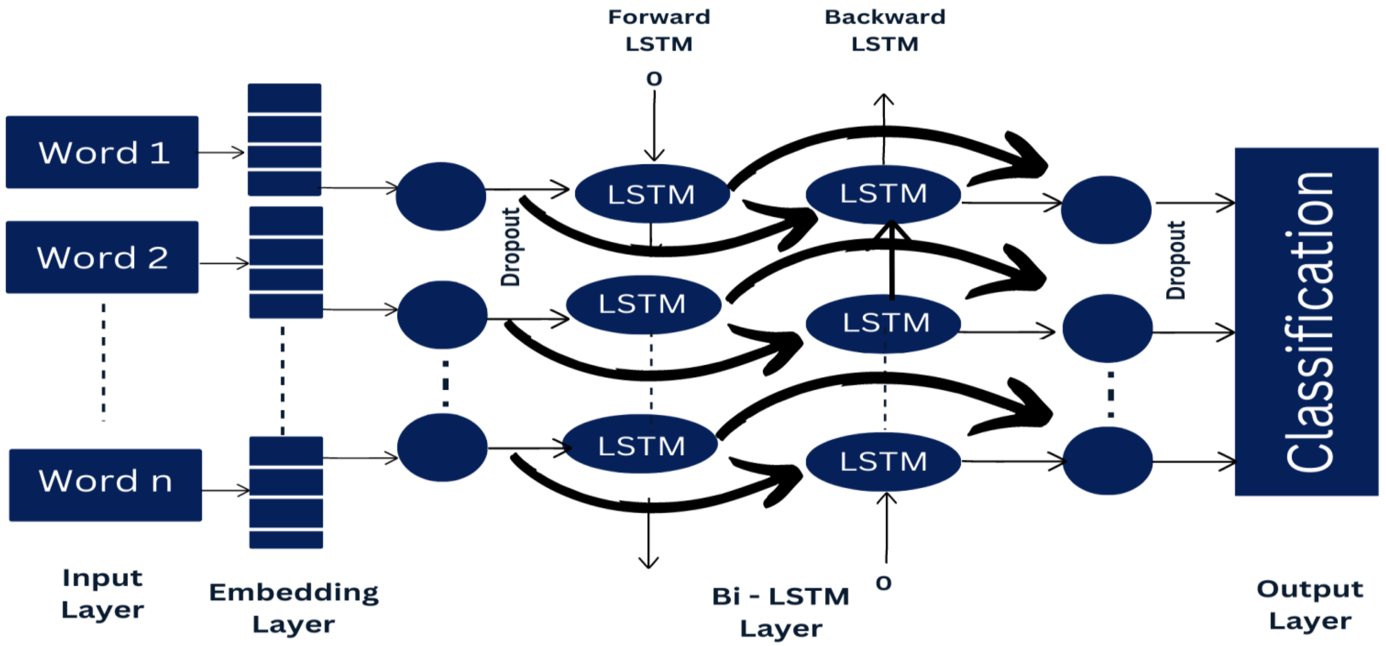
4.4.2. Bi-LSTM

- A bidirectional LSTM network is a type of neural network that processes input data in both the forward and backward directions.
- During the forward pass, the BiLSTM processes the input sequence in the normal forward direction.
- During the backward pass, it processes the same sequence in reverse order.
- The final output of the BiLSTM model is obtained by concatenating the hidden states of the forward and backward LSTM networks at each time step.

4. Methodology

4.4. Proposed Model

4.4.2. Bi-LSTM



4. Methodology

4.4. Proposed Model

4.4.3. CNN-Bi-LSTM

The CNN-Bi LSTM model is a type of deep learning model used for detecting fake news. This model is a combination of two powerful neural network architectures: Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi LSTM) network.

My model is composed of different layers :

- Convolutional Neural Network (CNN) Layer.
- Max Pooling Layer.
- Bidirectional Long Short-Term Memory (Bi LSTM) Layer.
- Fully Connected Layer.

4.4. Proposed Model

4.4.3. CNN-Bi-LSTM

Convolutional Neural Network(CNN) Layer :

The first layer of the model is the CNN layer. Here is how CNN layer works in model :

- Input Data.
- Convolutional Layer.
- Non-Linear Activation.

Max Pooling Layer :

After the CNN layer, the output is passed through a max-pooling layer. This layer reduces the spatial size of the feature map while retaining the most important features.

4. Methodology

4.4. Proposed Model

4.4.3. CNN-Bi-LSTM

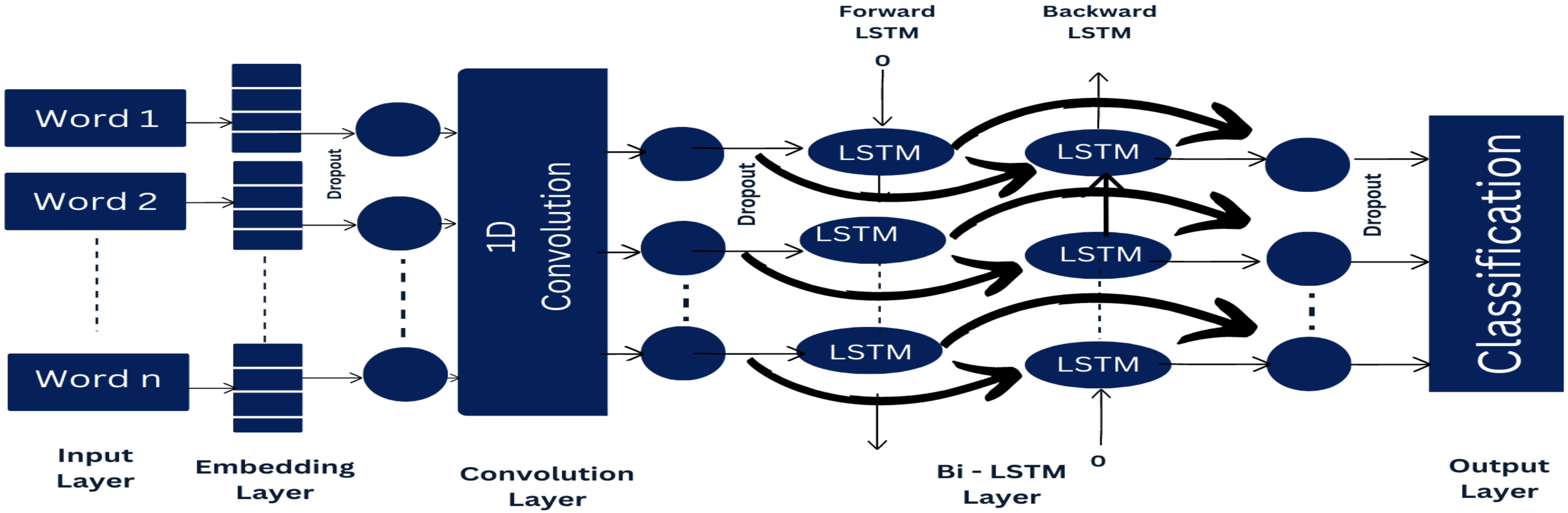
Bidirectional Long Short-Term Memory (Bi LSTM) Layer :
The next layer in the model is the Bi LSTM layer. This layer is responsible for capturing the temporal information in the input data. This allows the network to capture the context and dependencies between the words in the input text.

Fully Connected Layer :
The output of the Bi LSTM layer is then passed through a fully connected layer. This layer is responsible for mapping the features extracted from the input data to a binary classification output (fake news or real news).

4. Methodology

4.4. Proposed Model

4.4.3. CNN-Bi-LSTM



4. Methodology

4.5. Model Evaluation

We used accuracy as an evaluation metric to measure the effectiveness of our model on the test dataset. The accuracy metric is commonly used to evaluate classification models, as it indicates the model's ability to make correct predictions.

$$Accuracy = \frac{|TP| + |TN|}{|TP| + |TN| + |FP| + |FN|}$$

5. Experimental Setup

5.1. Software Requirements

a) Python3

(b) Google Colab or Jupyter Notebook

(c) Word Cloud

(d) Numpy

(e) Pandas

(f) Seaborn

(g) Matplotlib

(h) Sklearn(Scikit Learn)

(i) Natural Language Toolkit (NLTK)

(j) TensorFlow

(k) Keras

(l) Other Python Libraries

5. Experimental Setup

5.2. Hardware Requirements

(a) Laptop or computer with CPU having processor i3

(b) Minimum of 4 GB RAM is required.

(c) The Operating system that supports deep learning framework

5. Experimental Setup

5.3. Dataset

Data	Training Set		Validation Set		Testing Set	
	Original	Imbalanced	Original	Imbalanced	Original	Imbalanced
True	13644	13644	3449	3449	4324	4324
Fake	15090	1509	3735	374	4656	466

The distribution of genuine and fake data for both Original and Imbalanced datasets in the training, validation and testing sets is depicted in Figure.

5. Experimental Setup

5.4. Hyper-parameter Settings

(a) Embedding Dimension d = 100

(b) Dropout = 0.5

(c) Convolutional filters = 16

(d) Conv1D Activation function – Relu

(e) Conv1D kernel size = 3

(f) LSTM units = 32

(g) Activation function – Sigmoid

(h) Regularizer = 0.01

(i) Optimizer – Adam

(j) Loss function - Binary crossentropy

(k) Epochs = 15

(l) Batch size = 32

6. Results

Model	Accuracy
LSTM	0.9697
Bi-LSTM	0.9689
CNN-Bi-LSTM	0.9748

Table 6.1: Performance on the Balanced Dataset(Using GloVe Embedding)

Model	Accuracy
LSTM	0.9798
Bi-LSTM	0.9790
CNN-Bi-LSTM	0.9823

Table 6.2: Performance on the Imbalanced Dataset(Using GloVe Embedding)

6. Results

Model	Accuracy
LSTM	0.9636
Bi-LSTM	0.9672
CNN-Bi-LSTM	0.9680

Table 6.3: Performance on the Balanced Dataset(Using FastText Embedding)

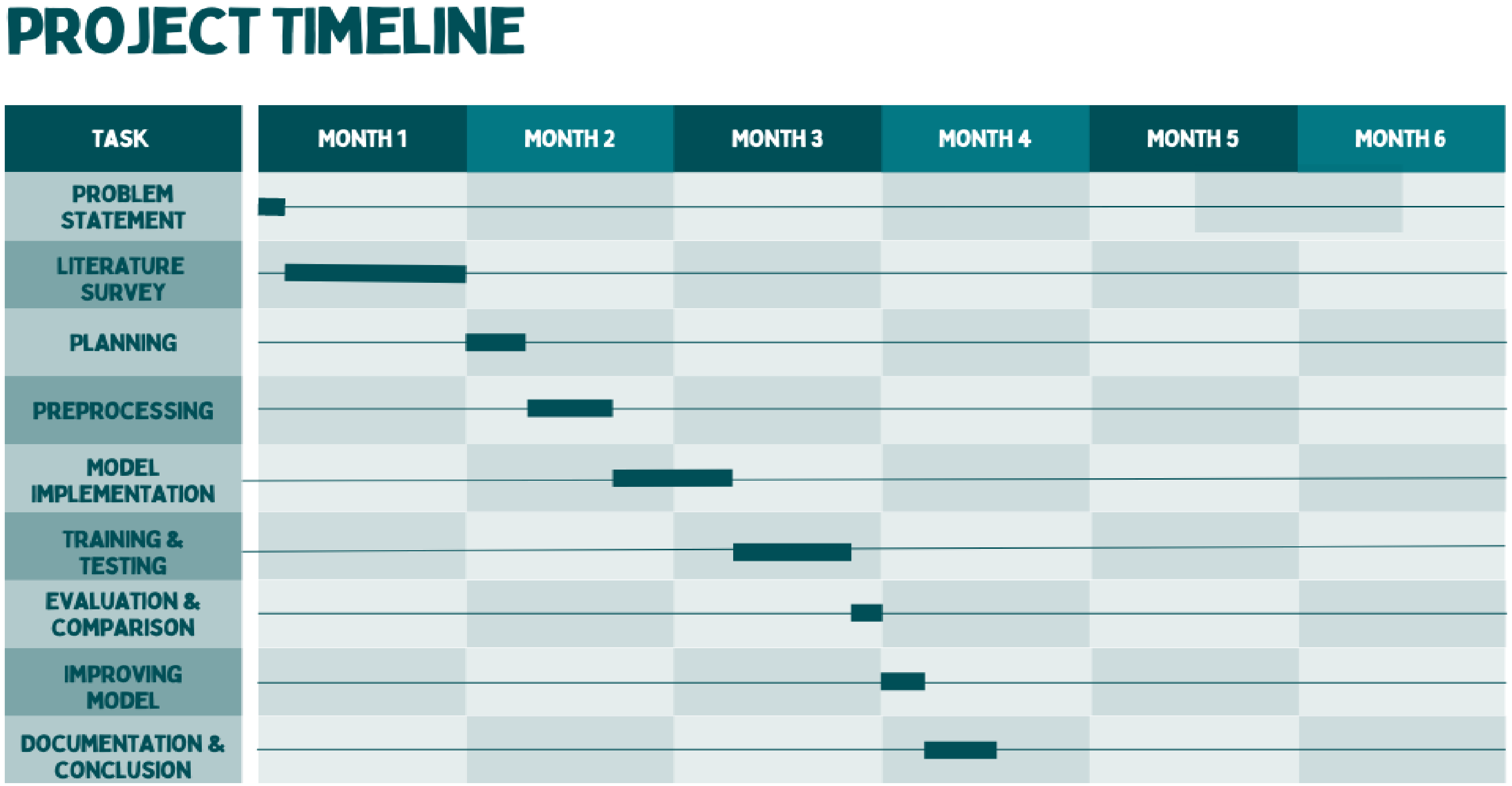
Model	Accuracy
LSTM	0.9754
Bi-LSTM	0.9708
CNN-Bi-LSTM	0.9740

Table 6.4: Performance on the Imbalanced Dataset(Using FastText Embedding)

7. Conclusion

- I developed a hybrid deep learning model using CNN and Bi-LSTM by utilizing the ISOT fake news Dataset.
- To prepare the text data for input into these models, I employed different word embedding techniques like Glove(Global Vectors for Word Representation) Embedding and FastText Word Embedding.
- I compared CNN-Bi-LSTM with LSTM and Bi-LSTM.
- Our experimental results showed that all three models performed well in detecting fake news, with CNN-BiLSTM outperforming the basic LSTM model.

8. Timeline


PROJECT TIMELINE
TASK
MONTH 1
MONTH 2
MONTH 3
MONTH 4
MONTH 5
MONTH 6
PROBLEM STATEMENT
LITERATURE SURVEY
PLANNING
PREPROCESSING
MODEL IMPLEMENTATION
TRAINING & TESTING
EVALUATION & COMPARISON
IMPROVING MODEL
DOCUMENTATION & CONCLUSION

9. References

[1] Goldani, M. H., Momtazi, S., Safabakhsh, R. (2021). Detecting fake news with capsule neural networks. Applied Soft Computing, 101, 106991

[2] Hakak, S., Alazab, M., Khan, S., Gadekallu, T. R., Maddikunta, P. K. R., Khan, W. Z. (2021). An ensemble machine learning approach through effective feature extraction to classify fake news. Future Generation Computer Systems, 117, 47-58.

[3] Kula, S., Chora´s, M., Kozik, R., Ksieniewicz, P., Wo´zniak, M. (2020, June). Sentiment analysis for fake news detection by means of neural networks. In International Conference on Computational Science (pp. 653-666). Springer, Cham.

[4] Jamal Abdul Nasir, Osama Subhani Khan , Iraklis Varlamis (2021, April).Fake news detection: A hybrid CNN-RNN based deep learning approach.In International Journal of Information Management Data Insights.Volume 1 issue 1.

[5] I. Kadek Sastrawan, I.P.A. Bayupati, Dewa Made Sri Arsa (2022, sep). Detection of fake news using deep learning CNN–RNN based methods. In ICT express. Volume 8, Issue 3,(pg no. 396-408).

9. References

[6] Kaliyar, R. K., Singh, N. (2019, July). Misinformation Detection on Online Social Media-A Survey. In 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE.

[7] Hitesh Narayan Soneji Sughosh Sudhanvan (2021, March). Fake News Detection Using Convolutional Neural Networks and Random Forest—A Hybrid Approach.In Cybernetics, Cognition and Machine Learning Applications pp 349–359.

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

[9] Ahmed, H., Traore, I., Saad, S. (2018). Detecting opinion spams and fake news using text classification. Security and Privacy, 1(1), e9.

[10] Reis, J. C., Correia, A., Murai, F., Veloso, A., Benevenuto, F. (2019). Supervised learning for fake news detection. IEEE Intelligent Systems, 34(2), 76-81.

9. References

[11] https://www.uvic.ca/engineering/ece/isot/datasets/fake-news/index.php.

[12] https://scikitlearn.org/stable/modules/generated/sklearn.utils.shuffle.html.

[13] https://www.geeksforgeeks.org/pre-trained-word-embedding-using-glove-in-nlp-models.

Thank You