# Fake News Detection Using Python And Machine Learning

## ABSTRACT

Fake news is a topic that has been discussed for quite some time. Prior to the internet era, it was mostly distributed through yellow journalism, with a focus on sensational news such as crime, rumors, accidents, and amusing news. To rescue the life of people from these fake news propagation, detection of fake news at an early stage becomes the most crucial step. People unknowingly propagate fake news and become a part of fake news propagation. Various techniques exist to detect fake news in social media, among which neural networks have shown effective results. For this research, a deep learning based approach has been used to differentiate false news from the original ones. A “LSTM neural network” has been used to build the proposed model. Besides the neural network, a Word2Vec word embedding has been used for vector representation of textual words. Also, for feature extraction or vectorization, tokenization techniques have been used. The comparative analysis of multiple fake news detection techniques is analyzed. The results of the proposed model have been evaluated using accuracy metrics. The model outperformed by achieving 99.43% of accuracy.

## 1) Methodology

Our goal is to train our model to correctly predict whether a piece of news is true or false.The dataset in all contains about 40,000 articles among which includes both fake news as well as real news. The false news data and actual news data is separated into two different datasets, each with approximately 20,000 articles.

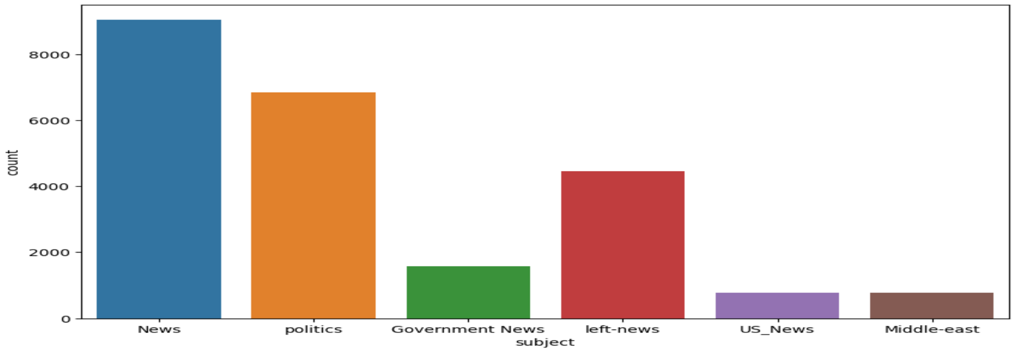


**Fig 1:** **Topics trending in each category of fake news and real news.**

### 1.1 Data visualization, preprocessing and cleaning

The dataset is characterized into two classes, one labeled as a true category and the second class labeled as a false category. Data visualization helps us comprehend what relative data means by displaying data in a visual context, such as maps or graphs. This makes it easier to spot trends, patterns, and outliers in large data sets by making the data more natural to analyze for the human mind. The dataset is classified into two categories that are fake news and original news. The first category is the true news category represented by class '1' and the second category is the fake news category represented by class '0'.

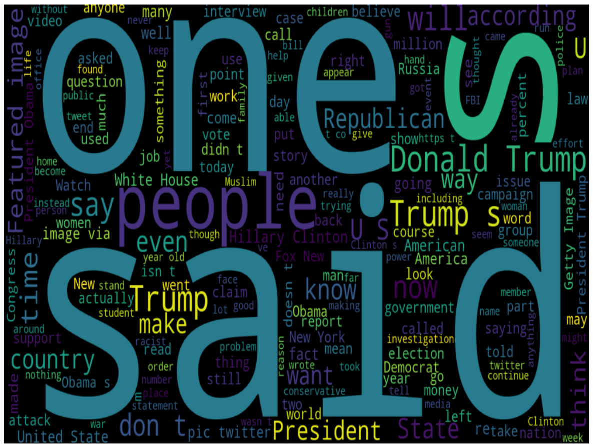


Fig 2: Word cloud for fake news dataset.

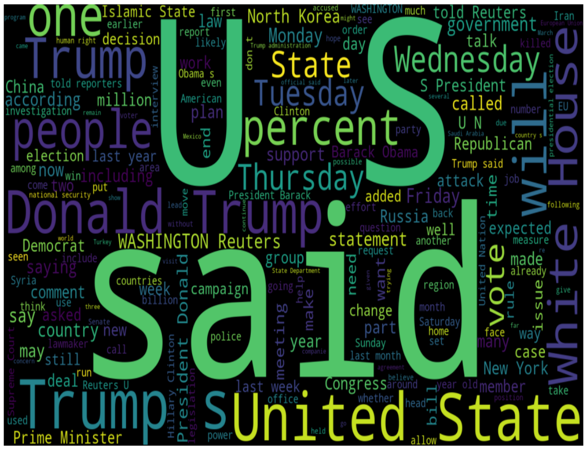


Fig 3: Word cloud for true news dataset.

In data cleaning we remove all the special characters and punctuations except alphanumeric characters and white spaces. We use Regular expression patterns to match non-alphanumeric characters and whitespace.

### 1.2 Tokenization & Vectorization

Tokenization is the process of breaking down a larger chunk of text into smaller units called tokens. We did the tokenization process by using tokenizer which is imported from keras.

Now we have to convert this text data into numerical. This technique is called vectorization. Vectorization refers to the process of converting textual data into a numerical representation that can be processed by machine learning algorithms.Word embedding is a specific type of word representation that aims to capture the semantic and syntactic relationships between words. It maps words from a vocabulary to dense vector representations in a continuous vector space. Word2Vec is a popular technique in NLP to represent words as dense vectors, capturing semantic relationships between words.

Now we trying to truncate all the news which have length more than 1000 words.The pad_sequence function is a utility function typically used in natural language processing (NLP) tasks to pad sequences to a fixed length here pad_sequence truncate all the text greater than 1000 words and add '0' to the text with < 1000 words.

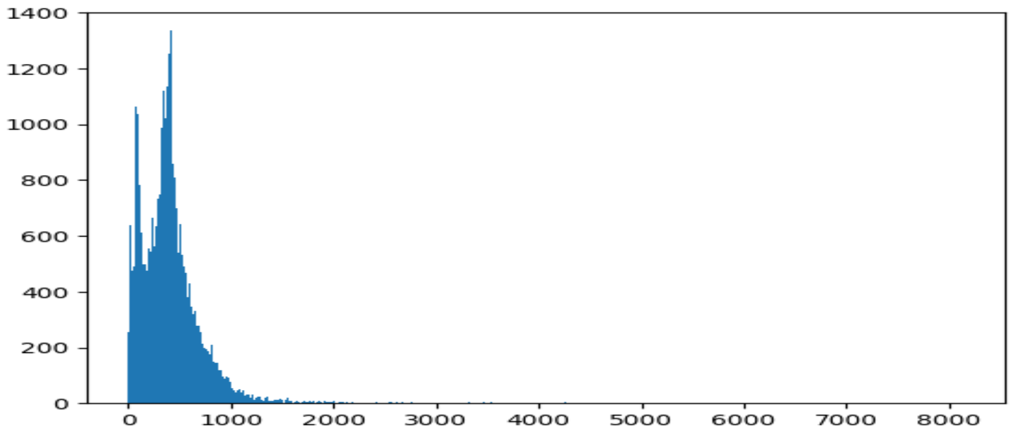


**Fig 4: Histogram which represents the number of words.**

### 1.3 Neural network: deep learning LSTM model

Long Short-Term Memory (LSTM ) is a type of recurrent neural network ( RNN) designed to understand and analyze sequences of data, such as text or time series. It overcomes the vanishing gradient problem of traditional RNNs, enabling it to capture long-term dependencies . LSTMs have internal memory cells that can remember and forget information, allowing them to learn and predict patterns within the data. They are widely used in tasks like language modeling, speech recognition, and sentiment analysis.

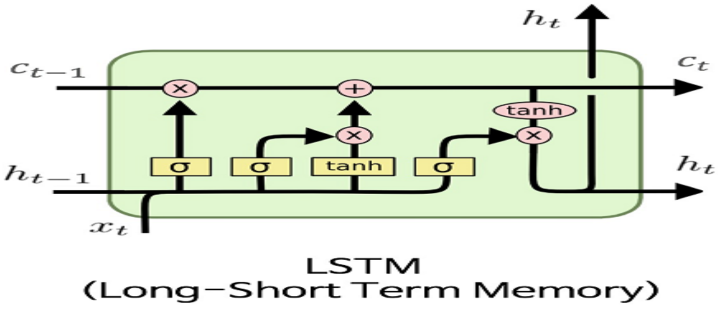
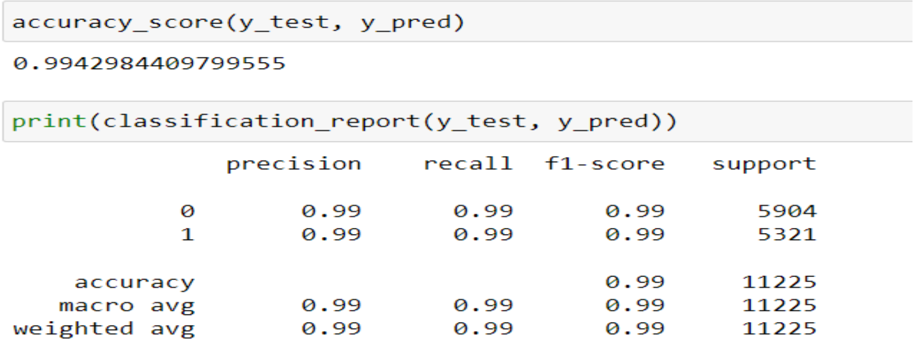


**Fig 5: Architecture of LSTM.**

- ➔ A sigmoid function produces values between 0 and 1, the weights connecting the input to these nodes can be taught to produce values near to zero, allowing certain input values to be “switched off.”
- ➔ Finally, there’s a tanh squashing function at the output layer, whose output is controlled by an output gate. This gate decides which values from the cell $ht$ are actually allowed as an output.

## 2) Results

The model has now been trained to recognize fake news as well as real news. The proposed model is evaluated based on accuracy, precision, recall, F1-score and support metrics.



```
accuracy_score(y_test, y_pred)
```

```
0.9942984409799555
```

```
print(classification_report(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       0.99      0.99      0.99      5904
           1       0.99      0.99      0.99      5321

    accuracy                           0.99     11225
   macro avg       0.99      0.99      0.99     11225
weighted avg       0.99      0.99      0.99     11225
```

**Fig 6 : Results of the fake news model.**

## 3) Conclusion

The focal point of our project lies in differentiating and detecting fake news and original or real news. The proposed model used a deep learning framework, which makes use of neural networks and the long short-term memory architecture.

Finally, the model is programmed to distinguish between true and false news. The evaluation metric of the proposed model is accuracy. Our proposed model achieved 99.43% accuracy.

Team : Tech Knights
MRCET