Fake News Detection

**Waleed Asghar  
*Dept. of Computer Science*  
 *Hood College*Frederick, MD, USA  
wa5@hood.edu**

*Abstract*—Today's internet is primarily a layered cake of data. The layers never get ridiculed; instead, they are growing exponentially. The news covers a significant proportion of that cake. Considering fake news, it has been there since the advent of the internet. We can define fake news as "fictitious articles that are fabricated deliberately to deceive the readers." Many social media and news outlets publish fake news to get attention or get public views.

The goal is to generate profit by spreading rumors or fake news among people, causing chaos or unrest among the public. Hence, there is a need to identify such article or news and prevent those. The purpose of the work is to come up with a solution that can be utilized by the users to detect and filter out accurately the fake news and false information posted by articles or some other outlets.

In this paper, we highlight an overview of DL techniques' application to fake news detection dataset. We classify and validate the dataset to produce the best fake news detection results. Although many research papers have been published, the application of DL to fake news detection is still an important part considering the growing hybrid network globally: to stimulate further work in this area, we conclude the paper by proposing the most accurate model for the fake news detector.

Keywords—fake news, DL(Deep Learning)), DL techniques, fake news detector

# Introduction

What is DL? As defined by S Marsland, Deep learning (DL) is a branch of Artificial Intelligence that pushes forward the idea that machines can learn by themselves how to solve a specific problem [1]. By leveraging complex mathematical and statistical tools, DL renders machines capable of performing independently intellectual tasks that have been traditionally solved by human beings.[2]

In this paper, our emphasis is on DL techniques. The practical implementation of these techniques will be solely in python and PyTorch library. We will code DL techniques in python on our fake news dataset to select the best algorithm to detect fake news. A fundamental question arises will the deep learning techniques give accuracy and validation results to any human? Hence, for the question to be answered, we will be using three different techniques. Any algorithm can't achieve 100% accuracy until now, but the model with the highest accuracy approximately in the 90s will be close enough to be equivalent to humans' results.

The whole project has been decided to be programmed in python. The reason for selecting python is its simplicity and understandability, no matter how complex it is. Besides, it offers a lot of libraries and modules to support the whole in-depth learning process.

# Dataset Overview

## Word Clouds

The fake news detection will be based on word clouds. There will be two-word clouds: a reliable word cloud and a non-reliable one (real & fake words).

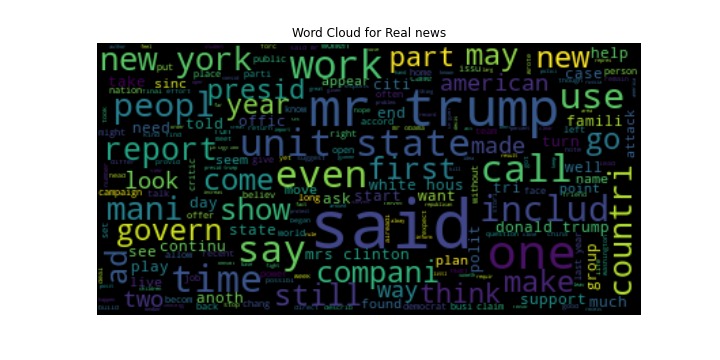


Fig I Word Cloud for Real news

Hence, the algorithm will be classifying the news as real if it finds words in it that present in the word mentioned above the cloud.



Fig II Word Cloud for Fake news

Hence, the classification of fake news will be based on these clouds. These word clouds have been first preprocessed. Processes like stop words, stemming, and punctuation removal have been applied.

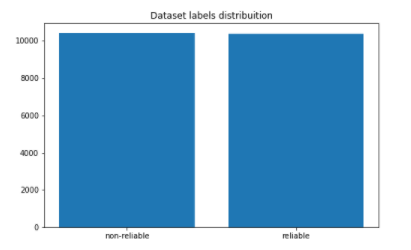


Fig III Reliable & Non-Reliable Word Clouds Graphic Representation

## Training Dataset

The training dataset is taken from Kaggle.[3] Particulars of the dataset have been stated below:

* Id: unique id for a news article
* Title: the title of a news article
* Author: Author of the news article
* Text: the text of the article; could be incomplete
* Label: the label that marks the article as potentially unreliable

# Methodology Overview

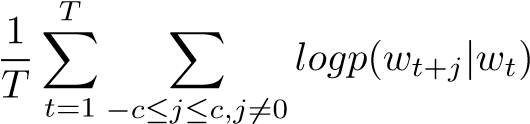
The whole process is divided into four stages. Each stage has been carefully planned and analyzed before being drafted.

## Stage-I:

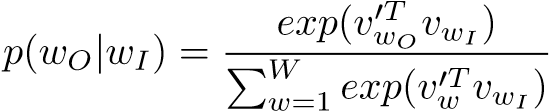
Firstly, word compounds will be translated and stored. Two categories of the word compounds will help us classify whether the information/news is reliable or not.

To feed textual data to a machine, it needs to be converted to a digital stream of binary numbers. Hence, the one-hot encoding of the input data following a dictionary for which each word has only one bit high, representing its unique translation in the dictionary, is translated into. The preprocessing following python libraries have been used pandas, nltk, genism (word2vec), Keras for tokenization, NumPy, and sci-kit learn. After the data is preprocessed, it's given the shape of the data frame. Afterward, it was converted to be in vector form to feed to a Deep Neural Network.

For the word2vec conversion, the skip-gram model is used. This model learns the vector representation from the raw data using the similarity between several words based on their context. The Skip-gram model's objective is to learn word representations that are useful for predicting nearby words in a document. Formally, given a sequence of training words/sentence *w*1, *w*2, *w*3, . . . , *wT* , the objective of the Skipgram model is to increase the average log probability defined as:

 (1)

where c is the size of training context (which can be a function of the center word *wt*). The Skip-gram formulation defines p(*wt*+*j* —*wt*) using the softmax function:

 (2)

where *vw* and are the "input" and "output" vector representations of w, and W is the number of words in the vocabulary.

## Stage-II:

To proceed forward, we had selected three neural networks. These have been listed below

### RCNN (Recurrent Based Convolutional Neural Network):

RCNN are a class of artificial neural network. According to the wiki definition, it is defined as the algorithm that connects the network can refer to last states and can therefore process arbitrary sequences of input [4]

### LSTM (Long Short Term Memory Neural Network):

LSTM is primarily a type of RCNN. Hence, it is defined as networks that are capable of learning order dependence in sequence prediction problems. [5]

### Bi\_LSTM (Bi-directional Long Short Term Memory):

Bidirectional recurrent neural networks are just putting two independent RNNs together. This structure allows the networks to have both backward and forward information about the sequence at every time step [6]

Hence, we will train the dataset using all of these techniques separately and check for their validity and accuracy. Python being a vast language, has aided in all types of modules. Python libraries used for implementing these libraries include genism and PyTorch primarily. The reason for selecting these RNNs was that they are best related to the type of textual classification we require. They are easy to apply to the dataset to train and, in return, get the results.

## Stage – III:

In the third stage, we will implement all these DL techniques. Implementation will help us to identify which of the above-listed technique gives more accurate and validated results.

## Stage -IV:

Hence, the results will be analyzed. After a careful analysis of the results, a conclusion will be drawn, which is the most suitable technique for applying fake news detection?

## Overview of the whole Process:

Fig IV Stages of Methodology

As researched, the general fake news detection process has been found as such. It clearly explains how the data is taken in and then preprocessed; afterward, the model is trained. Once the model is trained, it can easily classify any set of input data as Fake or Truthful News, which is the primary and only objective of this paper.

Fig V Generic Fake News Detection Outlook

# Result Analysis

The result of the DL techniques is satisfactory in all aspects. Typically, it is practically impossible to achieve 100 percent accuracy or validation, but accuracy and validation of about 90 percent usually are acceptable in all aspects. The results have been shared in the form of the tables

This first table is the representation of results from the LSTM technique using learning rates 0.005 and 0.0005

|  |  |  |
| --- | --- | --- |
| **LSTM** | | |
| Learning Rate | 0.005 | 0.0005 |
| Validation Accuracy | 94 | 84 |
| Training Accuracy | 97 | 85 |
| Valid Loss | 0.029 | 0.39 |
| Train Loss | 0.091 | 0.2999 |

TABLE I: LSTM Results

The above table shows that training accuracy is 97 %, and similarly, validation accuracy is 94 %, which is a good proportion with almost very little loss in both cases.

|  |  |  |
| --- | --- | --- |
| **Bi - LSTM** | | |
| Learning Rate | 0.005 | 0.0005 |
| Validation Accuracy | 94.5 | 85 |
| Training Accuracy | 98 | 86 |
| Valid Loss | 0.023 | 0.379 |
| Train Loss | 0.098 | 0.362 |

TABLE II: Bi - LSTM Results

The above table shows the training accuracy to be 98 % and validation accuracy 94.5 at a learning rate of 0.005. Similarly, valid loss and train loss is also minimal and acceptable.

|  |  |  |
| --- | --- | --- |
| **RCNN** | | |
| Learning Rate | 0.05 | 0.005 |
| Validation Accuracy | 95.6 | 94.7 |
| Training Accuracy | 97 | 94 |
| Valid Loss | 0.05 | 0.09 |
| Train Loss | 0.088 | 0.109 |

TABLE III: RCNN Results

The above table shows the result of the RCNN algorithm. The results are equally satisfying when considered with the other two DL techniques

We have considered that data results which are obtained at the learning rate of 0.005 from all the DL techniques

Fig VI Accuracy Results

Hence, Fig IV shows that RCNN has an immense value for validation accuracy. But Bi – LSTM has an enormous value for training accuracy. However, LSTM has the smallest value in both cases. Therefore, this makes us confident in defining LSTM as not a better technique for Fake News Detection. This leaves us with Bi -LSTM, and RCNN. Both techniques are competing with each other.

Fig VII Loss Results

In Fig V. most less validation loss is of Bi – LSTM. Similarly, most less train loss is of LSTM. The graph clearly shows that RCNN has values large than others making it clear that RCNN is failing. The LSTM and Bi – LSTM are both competing techniques regarding the data results

# CONCLUSION

Considering all of aspects, we have concluded that since validation accuracy is prime suspect to declare the best DL technique for the fake news detection, After analysis of results we come to conclusion that RCNN provides the most validation accuracy results but similarly it also provide most validation loss. On the other hand, Bi-LSTM provides almost same validation accuracy as RCNN with less validation loss as compared to RCNN so Bi-LSTM would be the suitable choice.

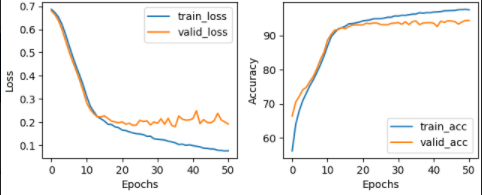


Fig VIII Curve Plot for Bi-LSTM Accuracy and Loss results (Epochs)

The curve plot for Bi-LSTM at learning rate 0.005 shows a significant decrease in the validation loss and a similarly significant increase in the excellent accuracy that justifies our conclusion.

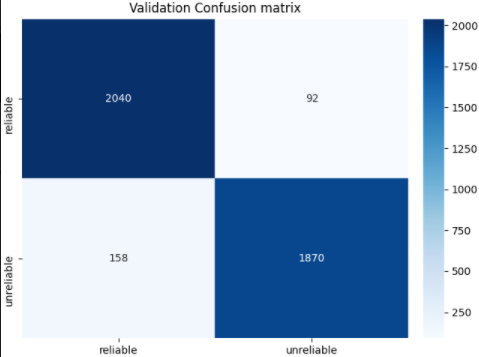


Fig IX Confusion Matrix of Validation (Bi-LSTM)

The validation confusion matrix results also are very much in alignment with the conclusion results. The real positive result makes it pretty much straightforward.

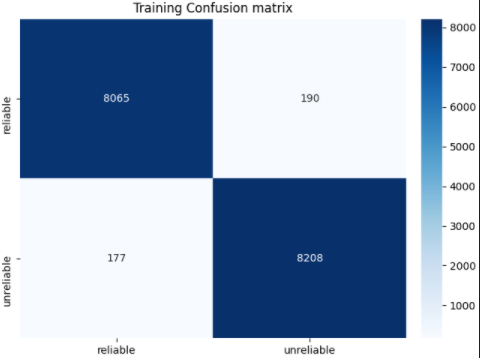


Fig X Confusion Matrix of Training (Bi-LSTM)

The actual positive value in the confusion matrix of Training is 8065, which is a significant value. And unreliable value is 8208, which is also an enormous value and is justifies the conclusion.

Considering Bi – LSTM is the appropriate choice since it provides almost the same validation accuracy, but in addition to that, it results in giving the least validity loss, which is also an essential aspect to investigate.

##### References

1. S. Marsland, Machine learning: an algorithmic perspective. CRC Press, 2015.
2. An Overview on Application of Machine Learning Techniques in Optical Networks Francesco Musumeci, Member, IEEE, Cristina Rottondi, Member, IEEE.
3. <https://www.kaggle.com/c/fake-news/data>
4. <https://wiki.tum.de/display/lfdv/Recurrent+Neural+Networks+-+Combination+of+RNN+and+CNN>
5. <https://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/>
6. <https://medium.com/@raghavaggarwal0089/bi-lstm-bc3d68da8bd0>