FAKE NEWS DETECTION

Do you have the correct information?

Student Name: Medha Rudra
{medha.rudra@colorado.edu}

Introduction:

In the current era dominated by social media, almost 50% of data taken from Facebook is fake. Not so surprisingly, news publishers relied on sources like Facebook for 20% of their publishings. In these uncertain times of the Covid-19 pandemic, it becomes even more crucial to prevent the spread of fake news. It has been reported that approximately 80% of consumers in the United States reported having seen fake news on the coronavirus outbreak so far. These statistics indicate that the spread of fake news is a major problem, globally affecting the economic and political aspects of the people along with their well-being. This calls for the need of a solution that could simplistically classify a news or article as fake or real and I strongly believe that we can leverage machine learning and deep learning models for this task effectively. For this purpose, I have taken up the task to compare and analyze various machine learning models suitable for the task of fake news classification and extended my analysis to deep learning models to determine how useful it would be given the variations of available data.

Summary:

My project is based on Fake News Detection which is a binary classification task to classify a given news article as fake or real based on what we have seen from historical data. For this classification task, I chose (i) Fake News dataset and, (ii) Fake and Real News dataset among the available datasets (like LIAR, COVID-19 Fake News dataset) from Kaggle. I first decided to explore traditional and sophisticated machine learning models for this task. I experimented with the Naive Bayes Classifier, Logistic Regression, and the Passive Aggressive Classifier on the simpler dataset, that is, the Fake News dataset and used two different vectorization techniques, the Count Vectorizer and the TF-IDF Vectorizer to vectorize my data before using it with the machine learning models. Finally, I compared the performance of all these approaches and found that the best performance was shown by the Passive Aggressive Classifier with TF-IDF vectorization.

To further expand on my analysis, I used the Fake and Real News dataset which covers more varied news topics and is more detailed than the Fake News dataset. I chose the best model from my previous experiments, that is, the Passive Aggressive Classifier with TF-IDF vectorization and compared its performance with that of the LSTM (Long Short-Term Memory) deep learning model trained on GloVe embeddings. Through this experiment, I found that both models relatively performed the same because of the limitation of the dataset.

Team Details: I have been working on this project by myself since the beginning of the semester. I don't have other team members. I'm enrolled in group number 5 and named my team as Team

Alpha. Alpha signifies dominance and highest rank.The name Alpha stemmed from my goal to find the best and most efficient model for the task of fake news detection.

Details:

Datasets

Fake News Dataset

The Fake News dataset consists of over 20,000 news articles. For the purpose of my experiments, I divided this dataset into a train/test split of 80:20. The dataset has five attributes: (i) id , which is a unique identifier for each news article, (ii) title of the news article, (iii) author, (iv) text of the article - which could be incomplete and (v) label 0 or 1 where 0 means that the article is reliable and 1 implies that the article is unreliable. The dataset seems to be fairly balanced with 60% positive samples (label 0) and 40% negative samples (label 1). If we look closer into the details of the news articles, we can observe that the articles are mostly related to politics and do not include a variation of topics like contents of world news.

	id	title	author	text	label
0	0	House Dem Aide: We Didn't Even See Comey's Let...	Darrell Lucus	House Dem Aide: We Didn't Even See Comey's Let...	1
1	1	FLYNN: Hillary Clinton, Big Woman on Campus - ...	Daniel J. Flynn	Ever get the feeling your life circles the rou...	0
2	2	Why the Truth Might Get You Fired	Consortiumnews.com	Why the Truth Might Get You Fired October 29, ...	1

Figure 1: Samples from Fake News Dataset

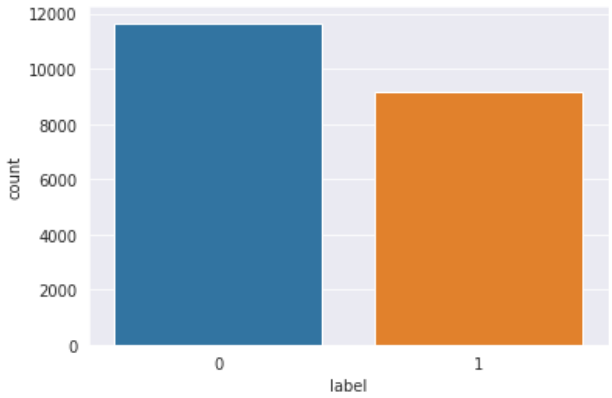


Figure 2: Number of samples (y-axis) in each category (x-axis); Real news (label 0) and Fake news (label1)

Fake and Real News Dataset

This dataset consists of about 40,000 data samples (news articles) with about 21,500 samples in True.csv (true news articles) and about 18,500 samples in Fake.csv (fake news articles). For my experiments, I took 80% of the dataset for training and the remaining 20% for testing. The Fake and Real News dataset has four attributes: (i) title of the news article, (ii) text of the article, (iii) subject on which the article is based, (iv) date. To make the data processing simpler, I combine the samples from True.csv and Fake.csv and added a column for class label indicating if the article is fake or real. This dataset seems to be more balanced than the Fake News dataset with 53% of the

samples being real news and 47% being fake news. The Fake and Real News dataset additionally has a lot more variation in topics of news articles like left-news, world news, politics, and so on.

	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	WASHINGTON (Reuters) - The head of a conservat...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	WASHINGTON (Reuters) - Transgender people will...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	WASHINGTON (Reuters) - The special counsel inv...	politicsNews	December 31, 2017

Figure 3: Examples of True Articles

	title	text	subject	date
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017

Figure 4: Examples of Fake Articles

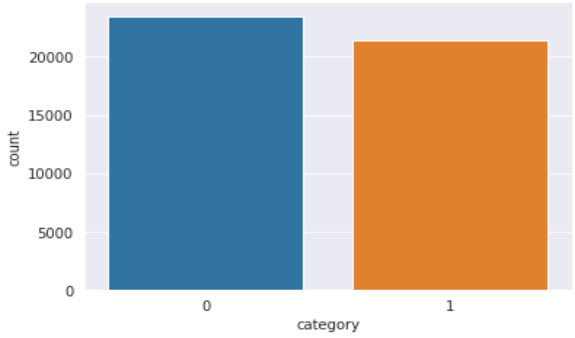


Figure 5: Count of samples in each category, Real news (label 1) and Fake news (label 0).

```
df.subject.value_counts()

politicsNews         11272
worldnews            10145
News                  9050
politics              6841
left-news             4459
Government News       1570
US_News                783
Middle-east            778
Name: subject, dtype: int64
```

Figure 6: Variety in the topics covered by the news articles

Methods

Algorithms

Naïve Bayes Classifier

The Naïve Bayes Classifier takes a probabilistic learning approach. It is based on Bayes' Theorem which states that:

$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right)P(A)}{P(B)}$, where A and B are events and P(B) is not 0.

It makes a naïve assumption that the contribution of each feature to the outcome is independent of each other and equal. In my experiments, I have used the Multinomial Naïve Bayes Classifier where each feature vector represents the frequencies with which certain events have been generated by a multinomial distribution.

Logistic Regression

It is a supervised machine learning algorithm which is commonly used for classification tasks. The target variable, y, can have only discrete values for a given set of feature input X. The logistic function (also called the Sigmoid function) helps in predicting the probability of a given sample

belonging to a particular class or category. The sigmoid function outputs a value between 0 and 1 for the prediction and can be represented as: $sigmoid(z) = \frac{1}{1+e^{-z}}$

Passive Aggressive Classifier

The Passive Aggressive Classifier is an online machine learning algorithm. It takes an aggressive approach by updating the weights for a sample for which the prediction was incorrect, otherwise it doesn't take any action and remains passive. Since it's an online learning algorithm, it processes one sample at a time to update its weights and moves on, and never sees the same data sample again. This is in contrast to batch learning where the entire dataset is processed at once. We can leverage this advantage of online learning algorithms like this to process datasets of immense size, which would have otherwise been computationally infeasible to process. Hence, Passive Aggressive Classifiers are very useful in processing datasets from social media like Facebook, WhatsApp and Twitter where new set of data gets added at very frequent intervals leaving us with a lot of data to process. If we want to read data from these sources dynamically, it would be ideal to use an online learning algorithm for this task like the Passive Aggressive Classifier.

LSTM (Long Short-Term Memory)

The architecture of Artificial Neural Networks is inspired from the biological neurons. It has a layered structure with a combination of algorithms working in a complex structure. Recurrent Neural Networks (RNN) are a type of artificial neural network that are capable of learning a sequence of data in such a way that each sample of data can be considered dependent on the previous samples. RNN models are able to remember the previous inputs over a span of time. Each RNN layer saves the from the output of the previous time step and feeds it back to the model thus building a recurrent connection. This helps RNNs remember information from the past and is useful to process a sequence of data.

RNN architectures sometimes run into the problem of "Vanishing Gradient". Gradients are multiplied by the factor by which the weight contributes to the error at each time step. This happens during backpropagation and impacts the gradient in a way that the effect is immense and gets propagated. For very small weights, the gradients shrink exponentially to the extent that the network stops training further due to negligible weight updation. For large weight, gradients grow exponentially. This, eventually, may result in underflow or overflow errors. To solve this problem, also called the vanishing gradient problem, we use gated RNNs.

An LSTM is a special type of gated RNN which consists of input, output and forget gates which determine which information is relevant to be kept or forgotten in the memory cell. This helps in controlling the flow of signals between respective states in the RNN architecture.

Vectorization Techniques

In order to feed text data to the machine learning algorithms, it is important to convert it to a form that the algorithm can process. The machine learning algorithms can only process numerical data and hence we use vectorization techniques that help us map textual data to real numbers. The vectorization techniques used in this experiment are as follows.

Count Vectorizer
This technique transforms textual data to numerical data by considering the frequency of each of the words in the entire text. The count vectorizer works by creating a matrix with each column being represented by unique words and each row is represented by text sample. The value in each cell of the matrix is the count or frequency of the unique words in the text sample.

TF-IDF Vectorizer
TF-IDF is short for Term Frequency Inverse Document Frequency. The term frequency is the count of the occurrences of a particular word in a document. The inverse document frequency utilizes the computation of number documents in the corpus compared to the frequency of a word over all the documents. Overall, TF-IDF checks the relevance of each word in the text.

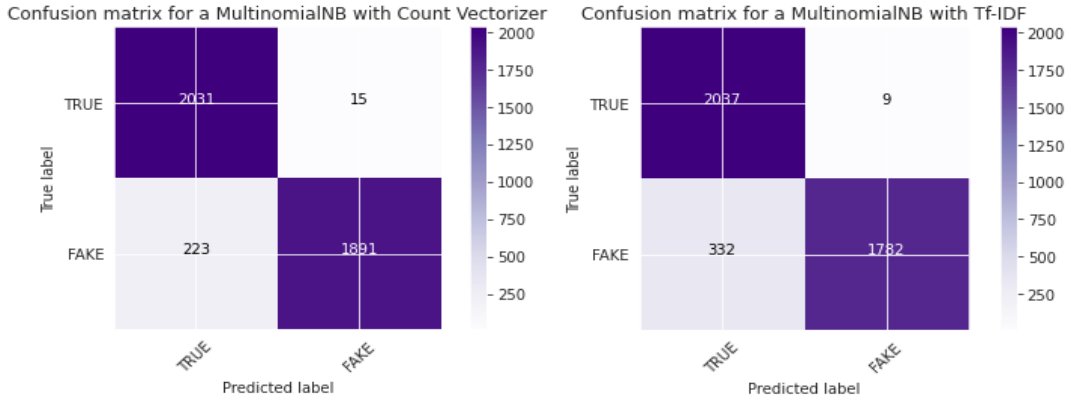
GloVe Embeddings
GloVe or Global Vectors for Word Embeddings is an unsupervised learning algorithm developed by Stanford used for obtaining vector representation of textual data. Training is performed on aggregated global word-word co-occurrence statistics from a text corpus. This results in representations that are linear substructures of the word vector space.

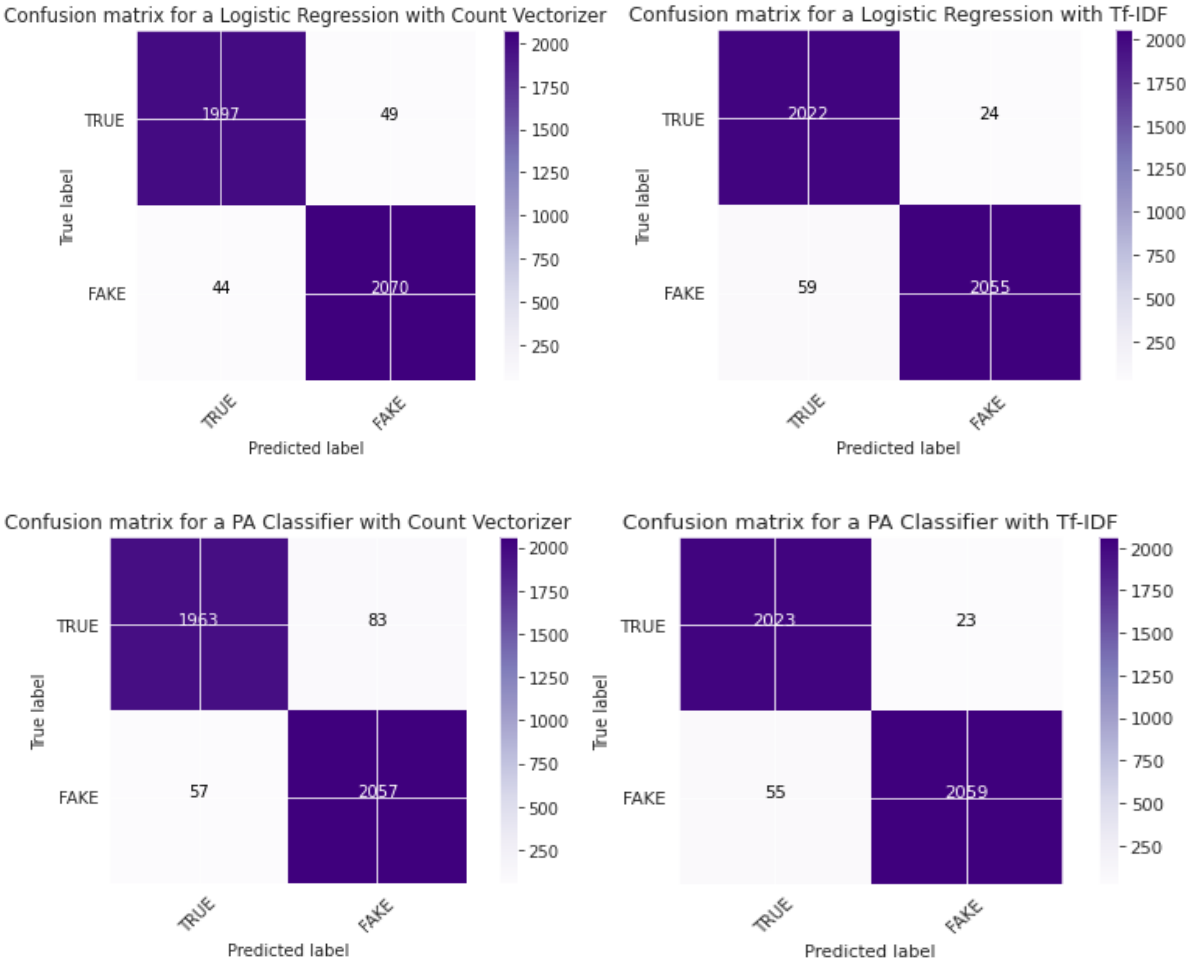
Experiments, Results and Analysis

Comparison of Machine Learning Models

Experiments:
The dataset chosen for this experiment is the Fake News Dataset from Kaggle. The dataset is first pre-processed to handle missing values. So that we don't lose a lot of information from missing values, I combined the text with the author information and title information which helped in obtaining richer feature set. I was initially applying stemming using the Porter Stemmer algorithm, but it did not make any significant changes to the results so I directly vectorized my data. Two vectorization techniques were used: (i) Count Vectorizer and (ii) TF-IDF Vectorizer. The vectorized data was then passed to three different machine learning models: (a) Naïve Bayes Classifier, (b) Logistic Regression model, and (c) Passive Aggressive Classifier. Finally, I used the test data to validate and compare each of these models and computed the accuracy and confusion matrix.

Results:





Analysis:

The Fake News Dataset which seemed to be fairly balanced performed with a good accuracy even on the Naïve Bayes Classifier. This is probably due the fact that the dataset under consideration mostly has only political news and is also geographically restricted to the news of the United States. Therefore, the model is easily able to learn the data even on limited training due to the bias in the dataset. The Logistic Regression model and the Passive Aggressive Classifier show comparable performance of about 98-99% accuracy. However, the Passive Aggressive Classifier still performs marginally better in terms of accuracy and makes more sense for a dataset that could be dynamically drawn from social media websites leading to a huge amount of data samples to process. This is because the Passive Aggressive Classifier is an online algorithm and does not process all the data at once, so it is computationally feasible to think about this approach for the task of fake news detection on social media.

Machine Learning models Vs Deep Learning models

Experiments:

For this experiment, I chose the Fake and Real News Dataset (from Kaggle) which is a more balanced dataset as compared to Fake News Dataset and has more variations in topics being covered by the news articles. For the machine learning, I considered the model developed for the previous experiment, the Passive Aggressive Classifier with TF-IDF vectorization. That is because it performed with the best results in the previous experiment. For the deep learning model, I vectorized the data by creating embeddings using GloVe. The embeddings were then passed to an LSTM model with two LSTM layers and two Dense layers with the final layer having sigmoid activation so that the output ranges between 0 and 1 that would help us predict the class, fake or real, of the news article. Both models were trained for 10 epochs with the same train/test split.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 250, 100)          13043100

 lstm (LSTM)                 (None, 250, 128)          117248

 lstm_1 (LSTM)               (None, 64)                49408

 dense (Dense)               (None, 32)                2080

 dense_1 (Dense)             (None, 1)                 33

=================================================================
Total params: 13,211,869
Trainable params: 168,769
Non-trainable params: 13,043,100
_________________________________________________________________
```

Figure 7: LSTM Model

Results:

Results obtained for LSTM Model:

```
              precision    recall  f1-score   support

           0       0.64      1.00      0.78      5858
           1       1.00      0.39      0.56      5367

    accuracy                           0.71     11225
   macro avg       0.82      0.69      0.67     11225
weighted avg       0.81      0.71      0.67     11225

Precision =  0.9975996159385502
Recall =  0.38718092043972424
```

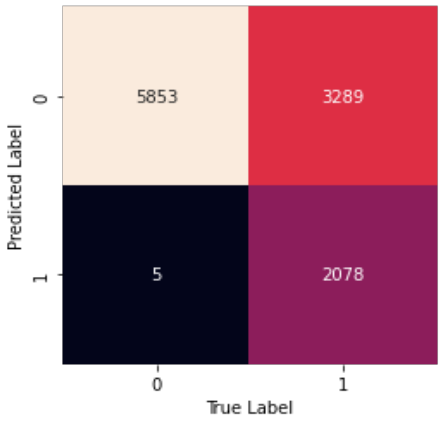
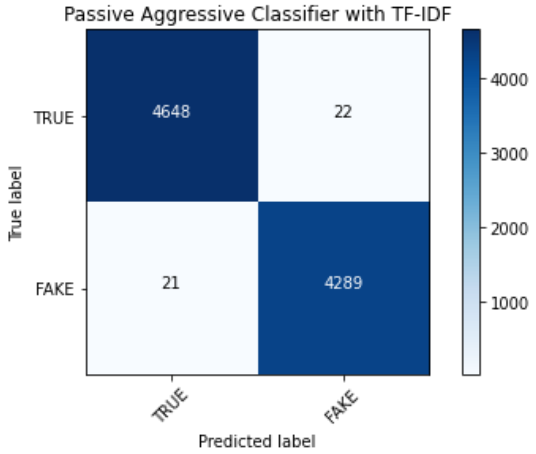


Figure 8: Confusion Matrix for LSTM *Figure 9: Accuracy Metrics for LSTM*

Results obtained for Passive Aggressive Classifier:



Analysis:

For this experiment, I observed that the machine learning model, that is, the Passive Aggressive Classifier showed comparable performance to the deep learning model, that is, the LSTM model. This can be due to the fact that machine learning algorithms tend to outperform deep learning models in the fewer data samples. Deep learning algorithms need a significantly large dataset to learn all the patterns and features of the data properly, given their complex architecture and the immense number of parameters the model needs to learn. For a more complex and detailed dataset like the LIAR dataset, where it would be useful for the model to learn more features and details of the data in order to make correct predictions, traditional machine learning models like the Passive Aggressive Classifiers would not suffice. That is why the current state-of-the-arts for the task of Fake News Detection is the BERT model.

About My Goals:

In my midterm reports I mentioned that I planned to implement a Passive Aggressive Classifier in addition to the other machine learning models like Logistic Regression and Naïve Bayes Classifier and compare these models. I achieved this goal and found that the best performance was obtained through the Passive Aggressive Classifier. I also mentioned that I would like to extend this project to various other dataset than the Fake News dataset and I could implement the Passive Aggressive Classifier and an LSTM model to experiment on a new dataset – Fake and Real News Dataset. I wanted to fine tune my model for better performance and to achieve this goal, I experimented with various hyperparameters for my LSTM model and Passive Aggressive Classifier to observe if I could get better results. Apart from this, one major goal for me was to read and research on the current works taking place for the task of Fake News Detection. While I was conducting this research, I found that BERT models are the state-of-the-arts for this task. Another far-fetched goal for me was to try incorporating coreference resolution into this task and experiment with that concept. But due to my health issues and time constraints I couldn't incorporate Coreference Resolution to my work. However, I got the chance to read some papers on Coreference Resolution like Dan Jurafsky's work[16] that applies Discourse Coherence Theory to coreference resolution and another paper by Barhom[15] on Cross Document Coreference Resolution.

Code:

Comparison of Machine Learning Models:
Google Colab Link:
https://colab.research.google.com/drive/12WNo7Lq6EH6mfHBNbPT3ahrzzI5L_4Sd?usp=sharing

Machine Learning models Vs Deep Learning models:
Google Colab Link:
https://colab.research.google.com/drive/1H4f3GXSo-f3rblZJdJc1unQPEo6-cwSM?usp=sharing

References:

[1] https://www.sciencedirect.com/science/article/pii/S1084804521001326
[2] https://www.statista.com/topics/3251/fake-news/
[3] https://jmlr.csail.mit.edu/papers/volume7/crammer06a/crammer06a.pdf
[4] https://www.kaggle.com/code/namithadeshpande027/fake-and-real-news/data
[5] https://www.kaggle.com/competitions/fake-news/data
[6] https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html
[7] https://www.geeksforgeeks.org/understanding-logistic-regression/
[8] https://www.geeksforgeeks.org/passive-aggressive-classifiers/
[9] Deep Learning For NLP And Speech Recognition – Kamath
[10] Deep Learning Assignment, Problem Set 4 – Medha Rudra
[11] https://www.tutorialspoint.com/time_series/time_series_lstm_model.htm
[12] https://www.geeksforgeeks.org/using-countvectorizer-to-extracting-features-from-text/
[13] https://www.geeksforgeeks.org/understanding-tf-idf-term-frequency-inverse-document-frequency/
[14] https://nlp.stanford.edu/projects/glove/
[15] Revisiting Joint Modeling of Cross-document Entity and Event Coreference Resolution. Shany Barhom, Vered Shwartz, Alon Eirew, Michael Bugert, Nils Reimers and Ido Dagan. ACL 2019.
[16] Focus on what matters: Applying Discourse Coherence Theory to Cross Document Coreference. William Held, Dan Iter, Dan Jurafsky. EMNLP 2021.