A BENCHMARK STUDY ON MACHINE LEARNING METHODS FOR FAKE NEWS DETECTION

Neel Kanbar(202018024), MSc. Data Science,DAIICT Ahmedabad

Arnav Desai(202018031), MSc. Data Science,DAIICT Ahmedabad

Aneri Joshi(202018032), MSc.Data Science,DAIICT Ahmedabad

Pankti Fadia(202018045), MSc.Data Science,DAIICT Ahmedabad

Abstract :

Fake news and hoaxes have been there since before the advent of the Internet. The widely accepted definition of fake news is : "Fictitious Articles Deliberately Fabricated to Deceive Readers". Social media and news outlets publish fake news to increase readership as a part of Psychological Warfare. Social media platforms in their current state are extremely powerful and useful for their ability to allow users to discuss and share ideas and debate over issues such as democracy, education, and health also, with a negative perspective for creating biased opinion, manipulating mindsets and spreading Satire or Absurdity.

In Our project, Fake News Detection, we have implemented Supervised Machine Learning Algorithms to classify text. We have used TFIDF Vectorization - Using vectorisation of the news text and then analysing the tokens of words with our dataset, Machine Learning Models : Passive Aggressive Classifier, Random Forest Classifier and Logistic regression, Confusion Matrix and F1 Score for Accuracy Check. We have implemented Flask to give an Application Layer to our Project for detecting whether the news is fake or real.

I. Introduction :

The idea of fake news is not a novel concept. Notably, the idea has been in existence even before the emergence of the Internet as publishers used false and misleading information to further their interests. Tech companies such as Google, Facebook, and Twitter have attempted to address this particular concern. However, these efforts have hardly contributed towards solving the problem as the organizations have resorted to denying the individuals associated with such sites the revenue that they would have realized from the increased traffic. Users, on the other hand, continue to deal with sites containing false information and whose involvement tends to affect the reader's ability to engage with actual news.

There has been a rapid increase in the spread of fake news in the last decade, most prominently observed in the 2016 U.S. Elections, U.S and Iran World War III trending on Social Media like Twitter in early 2020s. One recent case is the spread of novel coronavirus, where fake reports spread over the Internet about the origin, nature, and behavior of the virus. The situation worsened as more people read about the fake contents online. Such proliferation of sharing articles online that do not confirm to facts has led to many problems not just limited to politics but covering various other domains such as Sports, Health and Science. One such are affected by fake news is the Financial Markets where a rumor disastrous consequences and may bring the market to a halt.

II. Methods :

In Our project we have implemented many Libraries, Classifiers, Metrics and Vectorization Techniques.

1. **Libraries :**

- **Pandas :**

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

We have used Pandas for reading and processing CSV file in our project.

- **Sklearn :**

Scikit-learn is a machine learning library for Python. It features various algorithms like support vector machines, random forests, and k-neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy .

We have used Sklearn for implementing models and calculating accuracy scores.

- **Matplotlib :**

Matplotlib is a visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays.

We have used matplotlib for plotting graphs and data visualization.

- **Mlxtend :**

Mlxtend (machine learning extensions) is a Python library of useful tools for the day-to-day data science tasks

We have used mlxtend for plotting for confusion matrices.

- **Collections :**

Collections in Python are containers that are used to store collections of data, for example, list, dict, set, tuple etc. These are built-in collections. Several modules have been developed that provide additional data structures to store collections of data.

We have used collections for counting labels.

- **NLTK :**

Natural Language Toolkit(NLTK) is a leading platform for building Python programs to work with human language data.

We have used nltk for pre-processing and tokenizing the data.

- **Pickle :**

Pickle in Python is primarily used in serializing and deserializing a Python object structure. In other words, it's the process of converting a Python object into a byte stream to store it in a file/database, maintain program state across sessions, or transport data over the network.

We have used pickle for saving and retrieving machine learning models.

- **Flask :**

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application. This web application can be some web pages, a blog, a wiki or go as big as a web-based calendar application or a commercial website.

We have used FLASK for creating Web-Application.

2. Data Preprocessing :

- **TFIDF :**

TF-IDF is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. This is done by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents. It converts TF-IDF is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. This is done by multiplying two metrics: how many times a word appears in a document, and the inverse document frequency of the word across a set of documents.

We have used vectorisation of the news text and then analysed the tokens of words with our dataset.

3. Models :

- **PassiveAggressiveClassifier :**

PassiveAggressive algorithms are generally used for large-scale learning. It is one of the few ‘online-learning algorithms‘. In online machine learning algorithms, the input data comes in sequential order and the machine learning model is updated step-by-step, as opposed to batch learning, where the entire training dataset is used at once. This is very useful in situations where there is a huge amount of data and it is computationally infeasible to train the entire dataset because of the sheer size of the data. We can simply say that an online-learning algorithm will get a training example, update the classifier, and then throw away the example.

- **RandomForestClassifier :**

Random forest has nearly the same hyperparameters as a decision tree or a bagging classifier. Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features.

- **Logistic Regression :**

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist.

4. Model Evaluation :

Model evaluation aims to estimate the generalization accuracy of a model. Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work.

- **Accuracy_Score :**

It has been used for evaluation of the accuracy score of a binary model.

$$\texttt{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

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

- **F1_Score :**

The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

- **Confusion Matrix:**

A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

		Predicted class: P	Predicted class: N
Actual Class	P	True Positives (TP)	False Negatives (FN)
	N	False Positives (FP)	True Negatives (TN)

III. Experimental Results :

In our Fake News detection, we have implemented 3 models for the classification whether the news is fake or real. And further we have calculated accuracy score, F1 score and plotted confusion matrix for each model. By implementing these models, we have compared the accuracy score of each model via Data Visualization.

A. Accuracy Score :

- **Accuracy of Passive Aggressive Classifier:** 93.69%
- **Accuracy of Logistic Regression:** 91.55%
- **Accuracy of Random Forest Classifier:** 90.45%

B. F1 Score :

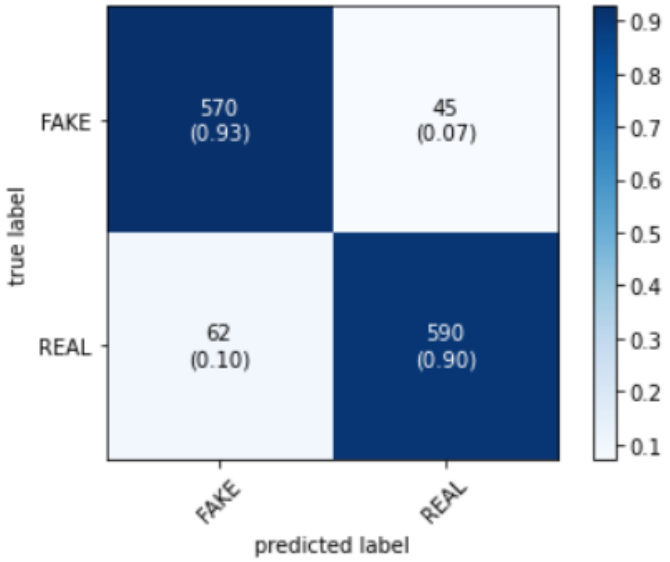
- F1 score of Logistic Regression:
(0.9159139677265499,0.9155485398579322,0.915566431124211, None)

- F1 score of Passive Aggressive Classifier:
(0.9369051844936325,0.9368587213891081,0.9368445435242649, None)

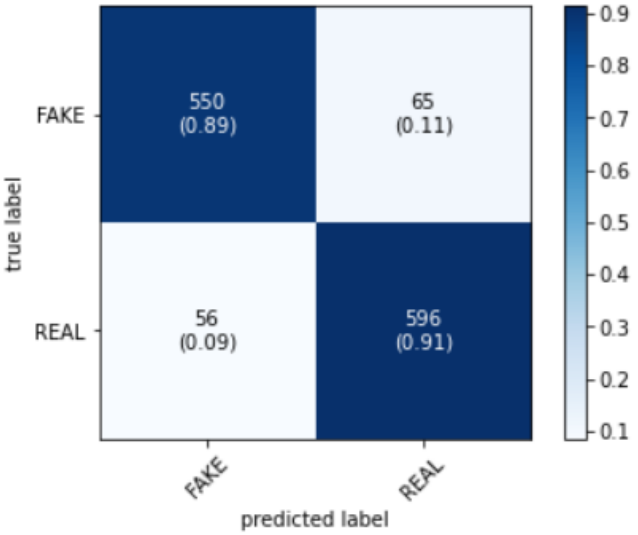
- F1 score of Random Forest Classifier:
(0.9045409151733013,0.904498816101026,0.9044741540775675, None)

C. Confusion Matrix :

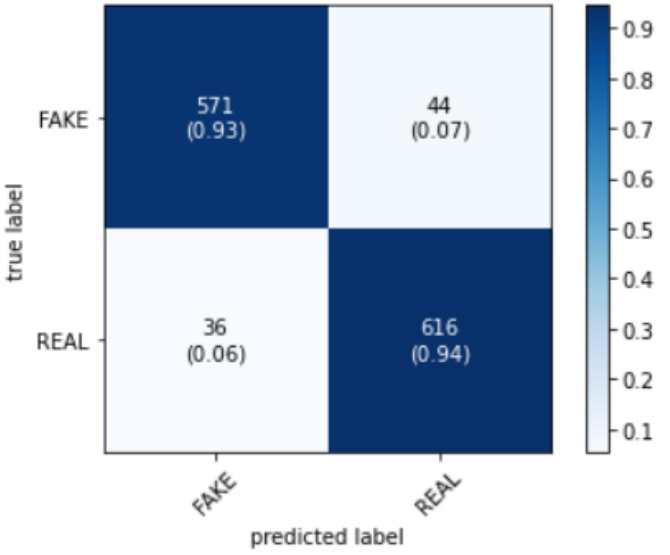
- **Confusion matrix of Logistic Regression** :



- **Confusion Matrix of Random Forest Classifier :**

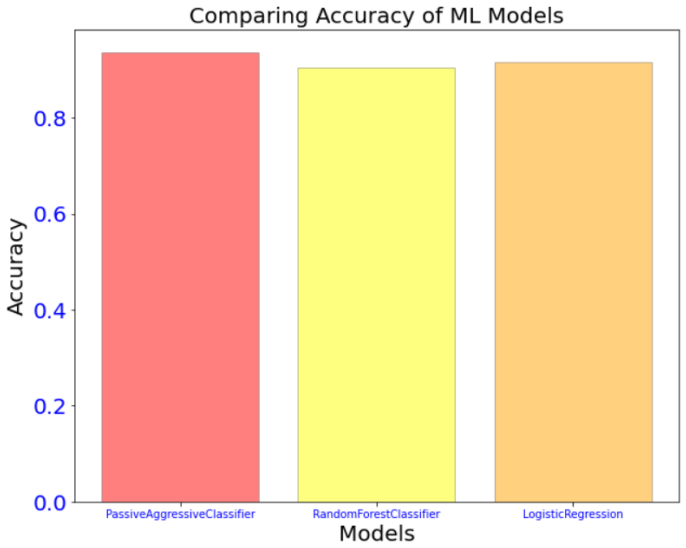


- **Confusion Matrix of PassiveAggressiveClassifier :**



D. Model Comparison :

We have predicted whether the news is fake or real using supervised learning algorithms using three models - PassiveAggressiveClassifier , RandomForestClassifier and Logistic Regression . We calculated accuracy score for each and lastly compared these models and concluded that PassiveAggressiveClassifier has the best accuracy score and is best suitable among the three models.



IV. Conclusion :

It is significant to find the accuracy of news which is available on the internet. In this research, we discussed the problem of classifying fake news articles using machine learning models. As fake news is a widely spreaded problem across the globe , many models and solutions are provided by many researchers. We have tried implementing some of built-in libraries and techniques and created fake news detection problem solving methods.

In this project , we have first preprocessed data with help of TF-IDF. Then we used three classification models - PassiveAggressiveClassifier , RandomForestClassifier and Logistic Regression. We got above 90% positive result when we checked the accuracy of all three models. Among all three , we got the highest accuracy of PassiveAggressiveClassifier which is 93.69% and also is the best suited model among all three in these cases.

References

- https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html
- **https://scikit-learn.org/stable/supervised_learning.html**
- **http://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html**
- https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html
- http://rasbt.github.io/mlxtend/user_guide/plotting/plot_confusion_matrix/
- https://arxiv.org/pdf/1906.11126v2.pdf
- https://matheo.uliege.be/bitstream/2268.2/8416/1/s134450_fake_news_detection_using_machine_learning.pdf