Fake News Detection using Machine Learning

Abstract - *This paper investigates machine-learning-based Fake news detection. In this work, we have collected a dataset of news in 2016 from Kaggle where we will find the tendency of news to be fake using statistical tools and predict whether the data is real or fake based on Machine learning algorithms.*

☐ *Introduction*

In Bangladesh, rumors, and disinformation are nothing new. The stories of cartridges greased with beef and pig fat, which led to the Sepoy Rebellion of 1857, are one of the oldest yet most prevalent rumors that people grow up reading in history books. In this decade, more and more of our lives are spent interacting online through social media platforms, individuals are increasingly seeking out and consuming news from social media outlets rather than traditional news organizations. There have been numerous examples in Bangladesh over the previous decade where social media misuse has become a security risk. There has been an increasing number of attacks against minority populations prompted by defamation on social media and related falsehoods, ranging from the incident in Ramu (2012) to Bhola (2019). Additionally, false child abduction rumors sparked mob attacks on 30 persons in 2019 resulting in the deaths of eight people. Moreover, during Covid-19, Bangladesh, like the rest of the world, was subjected to widespread coronavirus disinformation. To achieve certain objectives, the media may distort knowledge in a variety of ways. As a result, news reports are produced that are either partially true or entirely fraudulent. As a result, fake news has become a global crisis. Many scientists believe that artificial intelligence and machine learning could be used to combat fake news.

The purpose of the study is to see how machine learning models and specific methodologies function for this type of problem when given a labeled news dataset and to support (or not) the idea of employing AI for fake news identification. The distinction between this article and others on similar topics is that in this one, Logistic Regression was employed expressly for false news identification; also, the constructed system was evaluated on a relatively current data set, allowing for a comparison of its performance in recent data.

☐ *Data Collection*

Online news can be obtained from a variety of sources including social media, search engines, websites, unauthorized news portals, unauthorized pages, or some unauthorized and unrated news agencies that try to make a profit just by selling fake news. On the internet, there are a few publicly available websites for databases for identifying fake news. In our project, we used a database from Kaggle. With social media, search engines, ads, homepages, and many other sources online news can be collected. Manually evaluating the validity of news, on the other hand, is a complex operation that usually necessitates domain experts to perform critical analysis of claims, supplementary evidence, context, and reporting from reliable sources. Expert journalists, fact-checking websites, industry detectors, and crowd-sourced workers are all common news data sources with annotations. However, there is no exact benchmark for the dataset for fake news detection problems. We gathered our dataset from Kaggle. Below is a description of the dataset we used for our project:

i) the dataset we used is in CSV format

ii) we used 12 columns for collecting data

iii) we took both fake and real news in this dataset

☐ *Data PreProcessing*

Online news data is highly unstructured data. The majority of them are informal communications containing inaccuracies, vulgarity, poor grammar, and lots of mistakes. In order to improve performance and dependability, approaches for utilizing resources to make informed selections must be developed. Before predictive modeling may be utilized to gain greater insights, the data must first be cleaned. To do this, some basic pre-processing was done on the News training dataset. The steps are following:

Data Cleansing:

i) In this dataset, there are a lot of values that are not present. Which are known as null values. Null values are the absence of data. To solve this, we replaced the empty dataset or the null values with an empty string. An empty string is a string with zero length or no character.

ii) We merged the author's name and news title together. So, we can see which article does not belong to any author, and identifying will be easier.

iii) We replaced the target variables 'fake' and 'real' with 1 and 0.

iv) We replaced all the capital letters with small letters for better readability and understanding.

v) We separated the label names from the dataset as we do not need the labels for our work. We will work on the values only.

Stemming:

Stemming helps reduce a word to its stem form. It treats related words in the same way. It takes the words to their original form or base form by removing the suffixes. For example, if the word is 'badly', the suffix part 'ly' will be removed from the word and only 'bad' will be taken. It minimizes the number of words in the database, but the actual words are frequently overlooked. Note: Words with the same stem are treated as synonyms by some search engines.

Vectorizing:

Vectorization is the process where we convert our text data to an integer format. By doing this, machine learning algorithms can understand our data.

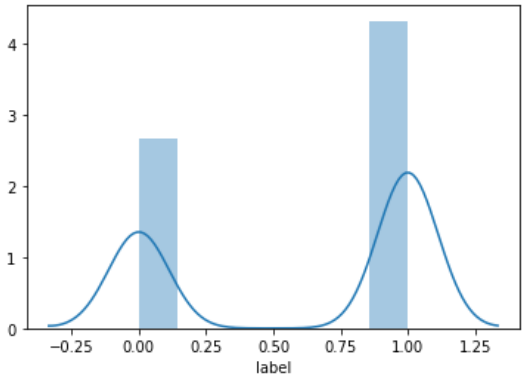
☐ *Data Exploration and Analysis*

In this part, we will explore our data and get to know some answers to questions like "What kind of data has the tendency of being fake? Are articles with no author fake?"

To predict fake news, it is necessary to know the dataset very well. In our dataset, we have 12 columns and 2094 rows. The 12 attributes are author, published, Title, Text, language, site_url, main_img_url, type, label, title_without_stopwords, text_without_stopwords, hasImage. As we are trying to predict if a news is fake or real so we will take the attribute "Label" as our target variable.

Characteristics of Fake News:
Their sources are not genuine most of the time. Fake news particularly does not have any author name mentioned and has less detailed information.

In our data exploration part, we will see if the previously mentioned characteristics are correct for our dataset or not and analyze those data for our future ML modeling part.

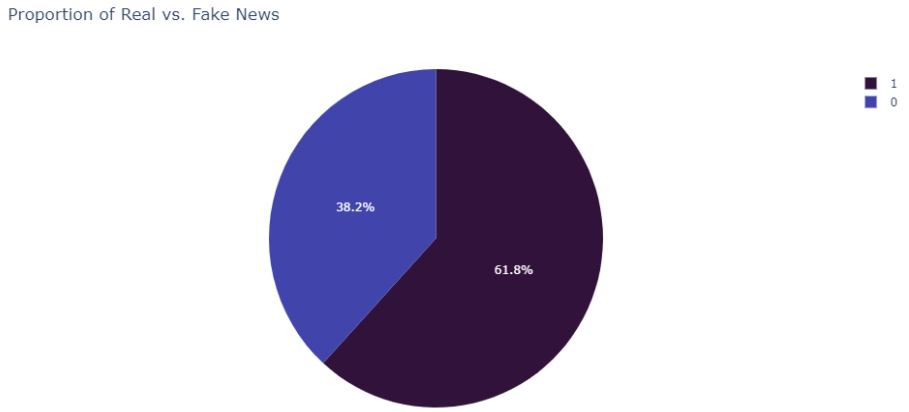


Skewness: -0.485877
Kurtosis: -1.765611

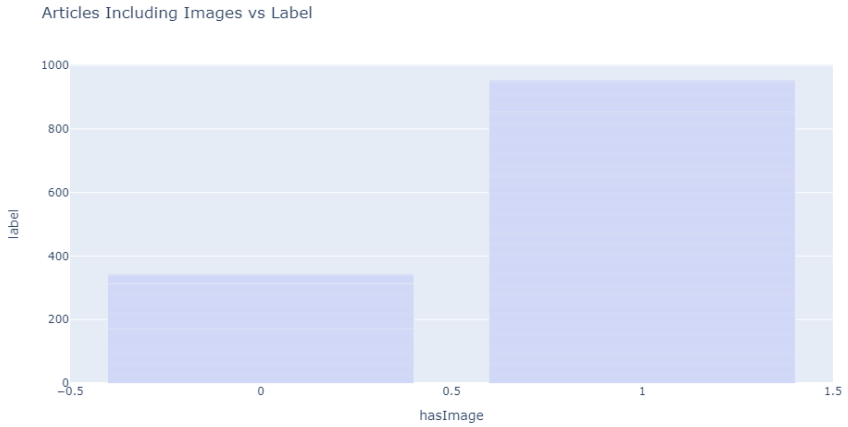
The distribution of our dataset is left-skewed and the distribution with a negative kurtosis value indicates that the distribution has lighter tails than the normal distribution.

	author	published	title	text	language	site_url	main_img_url	type	label	title_without_stopwords	text_without_stopwords
count	2094	2094	2094	2094	2094	2094	2094	2094	2094.000000	2094	2094
unique	490	2004	1782	1941	5	68	1228	8	NaN	1780	1937
top	No Author	2016-10-30T13:00:00.000+02:00	no title		english	wnd.com	No Image URL	bs	NaN	title	
freq	505	8	186	45	2016	100	466	601	NaN	187	49
mean	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.617956	NaN	NaN
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.486003	NaN	NaN
min	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.000000	NaN	NaN
25%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.000000	NaN	NaN
50%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.000000	NaN	NaN
75%	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.000000	NaN	NaN
max	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.000000	NaN	NaN

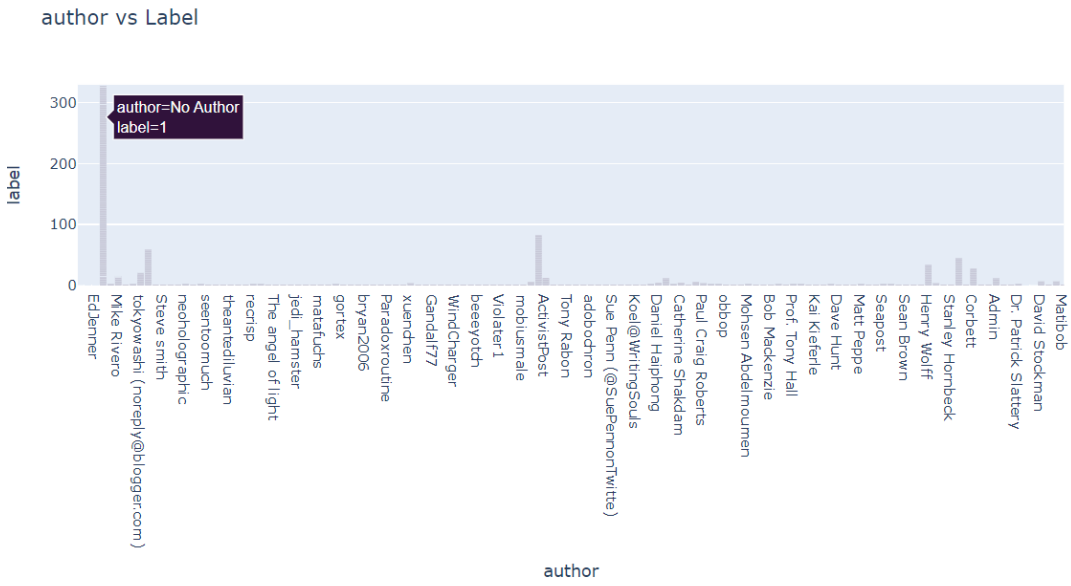
In the table, we can see that "No Author" and "No title" are the top elements in the author and title column.



In our dataset, the proportion of real vs fake news is 38.2% and 61.8% respectively which shows that there might be more fake news available online than actual news.

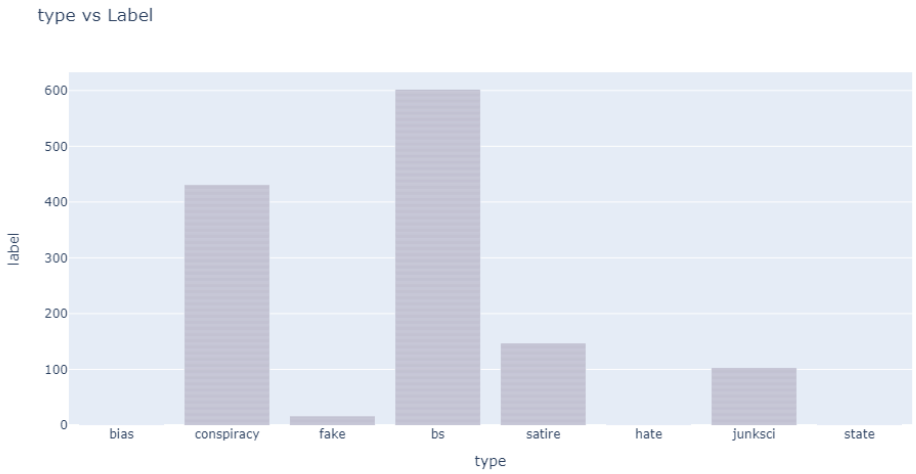


Moreover, if we compare "Label" with "hasImage", we can see that the news with the image is less likely to be fake than news with no image in our dataset.

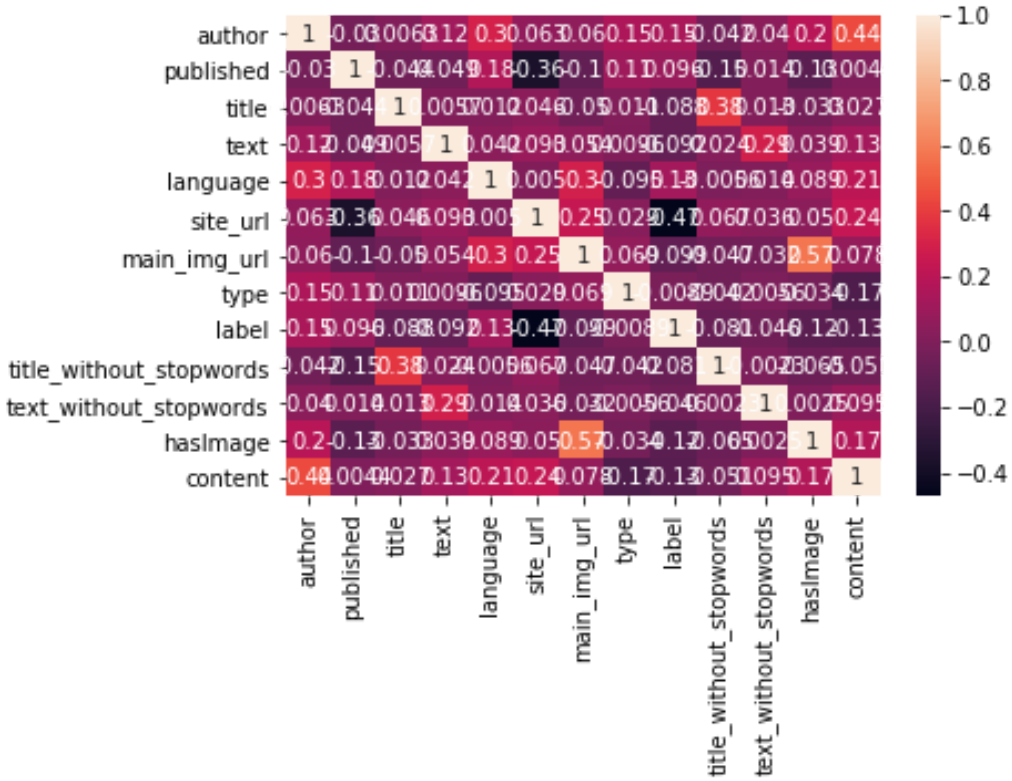


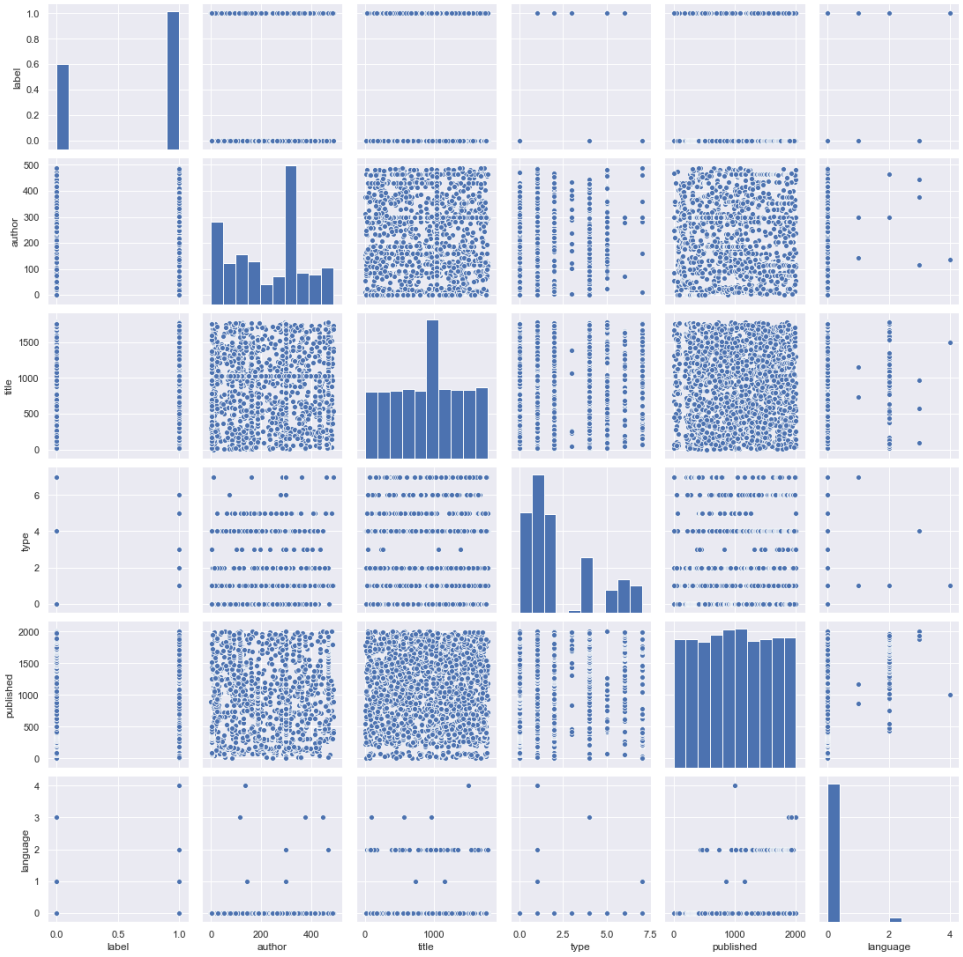
In the author bar chart, we get the highest amount of fake news when there is no author mentioned in an article. This matches our characteristics of Fake News.

Moreover, when news is fake most of the time it has no title name is shown in the graph.



In the "type" attribute there are 8 types of unique values and "bs" is the most responsible element for being fake news followed by "conspiracy".





Here we have our correlation matrix and scatterplot, which shows which attribute is closely related to our target variable "label". So, we choose, author, title, text_without_stopwords against "label" to build models in our next step.

Finally, from the data exploration, we get to know that in our dataset ***articles with "No author", "No title", "No Image" and "bs" types are more likely to be fake****. Additionally, due to the close relationship with the target variable "label", author, title, text_without_stopwords are chosen as content to build our models in the next step.*

☐ *Model Building and Evaluation:*

Algorithm:

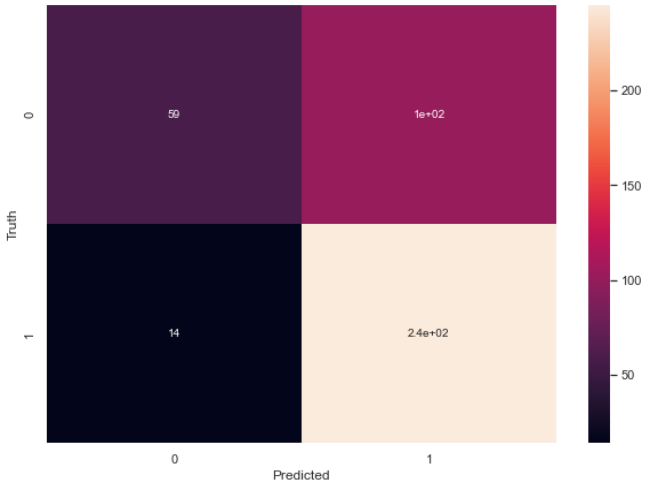
This section deals with training the classifiers. To anticipate the text's class, various classifiers were studied. We looked at four distinct machine-learning algorithms: Logistic regression, K-Nearest Neighbors (KNN), Random Forest, and Naïve Bayes Classifier. We evaluate the performance of algorithms for fake news detection problems and here confusion matrix has been used.

A confusion matrix is a table that shows how well a classification model (or "classifier") performs on a set of test data for which the true values are known. It enables the visualization of an algorithm's performance. A confusion matrix is a summary of classification problem prediction outcomes. The number of correct and imprecise predictions is totaled and broken down by class using count values.

Logistic Regression:

Logistic Regression is one of the basic and popular algorithms to solve a classification problem. It's a classification algorithm, not a regression algorithm. It's used to estimate discrete values (like 0/1, yes/no, and true/false) based on a set of independent variables (s). In simple terms, it fits data to a logit function to determine the probability of an event occurring. We performed hyperparameters to acquire the best results for each dataset, and we evaluated numerous values before obtaining the highest accuracies.

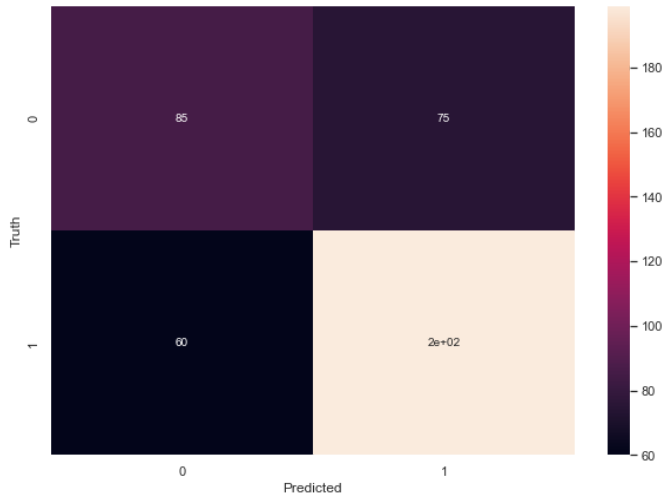
The matrix from this algorithm we generated:



	precision	recall	f1-score	support
0	0.81	0.37	0.51	160
1	0.71	0.95	0.81	259
accuracy			0.73	419
macro avg	0.76	0.66	0.66	419
weighted avg	0.75	0.73	0.69	419

KNN :

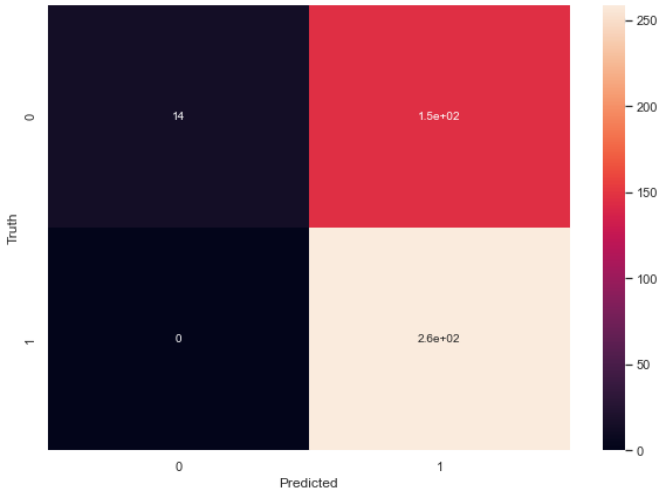
KNN is a supervised machine learning model that does not require a dependent variable to predict a certain data outcome. We give the model enough training data and let it determine which neighborhood a data point belongs to. The KNN model calculates the distance between a new data point and its nearest neighbors, and the value of K calculates the majority of its neighbors' votes; if K is 1, the new data point is assigned to the class with the shortest distance.
The matrix from this algorithm we generated:



	precision	recall	f1-score	support
0	0.59	0.53	0.56	160
1	0.73	0.77	0.75	259
accuracy			0.68	419
macro avg	0.66	0.65	0.65	419
weighted avg	0.67	0.68	0.67	419

Naïve Bayes Classifier:

This classification technique is based on the Bayes theorem, which assumes that the presence of a particular feature in a class is independent of the presence of any other feature. It provides a way for calculating the posterior probability.

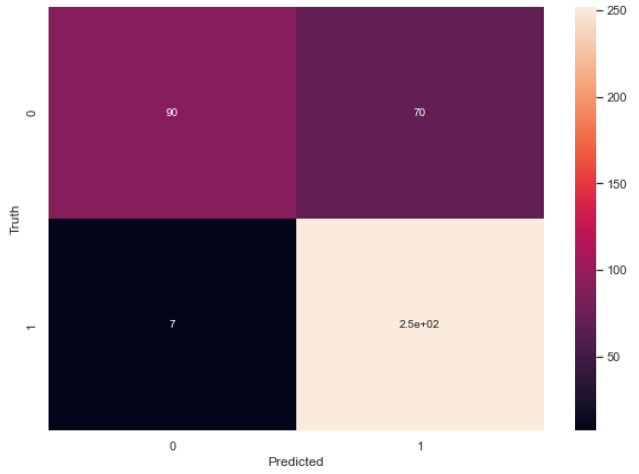


	precision	recall	f1-score	support
0	1.00	0.09	0.16	160
1	0.64	1.00	0.78	259
accuracy			0.65	419
macro avg	0.82	0.54	0.47	419
weighted avg	0.78	0.65	0.54	419

Random Forest

Random forest (RF) is a supervised learning model that is an enhanced version of decision trees (DT). RF is made up of a huge number of decision trees that work together to predict the outcome of a class, with the final prediction based on the class that obtained the most votes. Due to little correlation among trees, the error rate in the random forest is low when compared to other models. Our random forest model was trained with various parameters, such as varying numbers of estimators, in a grid search to find the optimal model that can accurately predict the outcome. There are multiple algorithms to decide a split in a decision tree based on the problem of regression or classification.

The matrix from this algorithm we generated:



	precision	recall	f1-score	support
0	0.93	0.56	0.70	160
1	0.78	0.97	0.87	259
accuracy			0.82	419
macro avg	0.86	0.77	0.78	419
weighted avg	0.84	0.82	0.80	419

Implementation Steps:

In our static part, we implemented all the algorithms, and here are the steps how these are implemented:

- *In the first step, we extracted features from the already preprocessed dataset. These features are; training data, testing data, and predicted scores for both training and testing data.*
- *Here, we have built the classifiers for analyzing fake news. We have used Naive-Bayes, Logistic Regression, KNN, and Random forest classifiers from sklearn. The extracted features which we processed were used in all of the classifiers.*
- *After fitting the model, we compared the f1 score, Accuracy, Precision, and Recall to check the confusion matrix.*
- *Two best-performing models were selected for the fake news classification after we fitted all the classifiers.*
- *We have performed a gridsearchCV method where we get the accuracy score by comparing both test and training data with the predicted data and finally we chose the best performing parameters for these classifiers.*
- *Finally, the model that was chosen, which was Random Forest Classification, was used for fake news detection with the probability of truth.*

☐ *Results*

We learn from data exploration that articles with "No author," "No title," "No Image," and "bs" types are more likely to be fraudulent in our dataset. After implementing the machine learning algorithm, the accuracy of each classifier is estimated. We observed that only one classifier had accuracy above 80% which is random forest. In all the cases we split into 20% test data and the remaining 80% is training data.

Classifiers	***Accuracy***	***Precision***	***Recall***
Logistic Regression	*0.72*	*0.81*	*0.37*
Naïve Bayes Classifier	*0.65*	*1.00*	*0.64*
KNN (k-nearest neighbors)	*0.67*	*0.59*	*0.53*
Random Forest	*0.82*	*0.93*	*0.78*

The model's positive predictive value (precision) represents the appropriate text among the repossessed text documents, whereas sensitivity (recall) represents the fraction of the total number of related text documents that were actually retrieved. As a result, there is also a graph that defines the comparison of these supervised learning algorithms. It can be estimated based on the accuracy and random forest is the one with higher accuracy.

Deployment:
In this part a random news from our test data to compare and then we stored it in a variable. And we submit the news in our prediction function to see whether the predictive system goes wrong or it gives us an accurate result.

☐ *Conclusion*

In this century people look through anything with a choice of virtual world and news is unlike this. When people start reading news from a particular site that person will be deceived into two different states. First,he might want to check if the news is real or not and secondly, they would start believing that their perceptions about a particular topic are true as assumed. In order to solve this state, we have developed a prediction model which can analyze news and at the same time, it will show whether the people are passing their valuable time on a true story or not. The

main objective of the classification of news is a complex task even when using classifier techniques because the input data is in text format and the news has a large number of characteristics that must be considered. This complex issue was addressed in our paper using classifiers that achieved an accuracy of 67% for KNN, 72% for Logistic Regression, 65% for Naive Bayes, and 82.33% for Random Forest. As a result, if a user feeds a specific news article or its headline into our model, there is an 82% chance that it will be classified to its true nature. So finally, we can say that this project might be expanded into a practical application that can take any input, regardless of language, and determine whether it is fake or real.

☐ *References*

[1] https://www.kaggle.com/datasets/ruchi798/source-based-news-classification