**FAKE NEWS DETECTION USING MACHINE LEARNING**

**1.INTRODUCTION**

The advent of the World Wide Web and the rapid adoption of social media platforms (such as Facebook and Twitter) paved the way for information dissemination that has never been witnessed in the human history before. Besides other use cases, news outlets benefitted from the widespread use of social media platforms by providing updated news in near real time to its subscribers. The news media evolved from newspapers, tabloids, and magazines to a digital form such as online news platforms, blogs, social media feeds, and other digital media formats [1]. It became easier for consumers to acquire the latest news at their fingertips. Facebook referrals account for 70% of traffic to news websites [2]. These 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. However, such platforms are also used with a negative perspective by certain entities commonly for monetary gain [3, 4] and in other cases for creating biased opinions, manipulating mindsets, and spreading satire or absurdity. The phenomenon is commonly known as fake news.

As an increasing amount of our lives is spent interacting online through social media platforms, more and more people tend to hunt out and consume news from social media instead of traditional news organizations.[1] The explanations for this alteration in consumption behaviours are inherent within the nature of those social media platforms: (i) it's often more timely and fewer expensive to consume news on social media compared with traditional journalism , like newspapers or television; and (ii) it's easier to further share, discuss , and discuss the news with friends or other readers on social media. For instance, 62 percent of U.S. adults get news on social media in 2016, while in 2012; only 49 percent reported seeing news on social media [1]. It had been also found that social media now outperforms television because the major news source. Despite the benefits provided by social media, the standard of stories on social media is less than traditional news organizations. However, because it's inexpensive to supply news online and far faster and easier to propagate through social media, large volumes of faux news, i.e., those news articles with intentionally false information, are produced online for a spread of purposes, like financial and political gain. it had been estimated that over 1 million tweets are associated with fake news “Pizza gate" by the top of the presidential election. Given the prevalence of this new phenomenon, “Fake news" was even named the word of the year by the Macquarie dictionary in 2016 [2]. The extensive spread of faux news can have a significant negative impact on individuals and society. First, fake news can shatter the authenticity equilibrium of the news ecosystem for instance; it's evident that the most popular fake news was even more outspread on Facebook than the most accepted genuine mainstream news during the U.S. 2016 presidential election. Second, fake news intentionally persuades consumers to simply accept biased or false beliefs. Fake news is typically manipulated by propagandists to convey political messages or influence for instance, some report shows that Russia has created fake accounts and social bots to spread false stories. Third, fake news changes the way people interpret and answer real news, for instance,

some fake news was just created to trigger people's distrust and make them confused; impeding their abilities to differentiate what's true from what's not. To assist mitigate the negative effects caused by fake news (both to profit the general public and therefore the news ecosystem). It's crucial that we build up methods to automatically detect fake news broadcast on social media [3].

Internet and social media have made the access to the news information much easier and comfortable [2]. Often Internet users can pursue the events of their concern in online form, and increased number of the mobile devices makes this process even easier. But with great possibilities come great challenges. Mass media have an enormous influence on the society, and because it often happens, there's someone who wants to require advantage of this fact. Sometimes to realize some goals mass-media may manipulate the knowledge in several ways. This result in producing of the news articles that isn’t completely true or maybe completely false. There even exist many websites that produce fake news almost exclusively. They intentionally publish hoaxes, half-truths, propaganda and disinformation asserting to be real news – often using social media to drive web traffic and magnify their effect. The most goals of faux news websites are to affect the general public opinion on certain matters (mostly political). Samples of such websites could also be found in Ukraine, United States of America, Germany, China and much of other countries [4]. Thus, fake news may be a global issue also as a worldwide challenge. Many scientists believe that fake news issue could also be addressed by means of machine learning and AI [5]. There’s a reason for that: recently AI algorithms have begun to work far better on many classification problems (image recognition, voice detection then on) because hardware is cheaper and larger datasets are available. There are several influential articles about automatic deception detection. In [6] the authors provide a general overview of the available techniques for the matter. In [7] the authors describe their method for fake news detection supported the feedback for the precise news within the micro blogs. In [8] the authors actually develop two systems for deception detection supported support vector machines and Naive Bayes classifier (this method is employed within the system described during this paper as well) respectively. They collect the info by means of asking people to directly provide true or false information on several topics – abortion, execution and friendship. The accuracy of the detection achieved by the system is around 70%. This text describes an easy fake news detection method supported one among the synthetic intelligence algorithms – naïve Bayes

classifier, Random Forest and Logistic Regression. The goal of the research is to look at how these particular methods work for this particular problem given a manually labelled news dataset and to support (or not) the thought of using AI for fake news detection. The difference between these article and articles on the similar topics is that during this paper Logistic Regression was specifically used for fake news detection; also, the developed system was tested on a comparatively new data set, which gave a chance to gauge its performance on a recent data.

1. Characteristics of Fake News:

They often have grammatical mistakes. They are often emotionally coloured. They often try to affect readers’ opinion on some topics. Their content is not always true. They often use attention seeking words and news format and click baits. They are too good to be true. Their sources are not genuine most of the times [9].

1.1 **Our Contributions**

In the current fake news corpus, there have been multiple instances where both supervised and unsupervised learning algorithms are used to classify text [20, 21]. However, most of the literature focuses on specific datasets or domains, most prominently the politics domain [10, 19, 21]. Therefore, the algorithm trained works best on a particular type of article’s domain and does not achieve optimal results when exposed to articles from other domains. Since articles from different domains have a unique textual structure, it is difficult to train a generic algorithm that works best on all particular news domains. In this paper, we propose a solution to the fake news detection problem using the machine learning ensemble approach. Our study explores different textual properties that could be used to distinguish fake contents from real. By using those properties, we train a combination of different machine learning algorithms using various ensemble methods that are not thoroughly explored in the current literature. The ensemble learners have proven to be useful in a wide variety of applications, as the learning models have the tendency to reduce error rate by using techniques such as bagging and boosting. These techniques facilitate the training of different machine learning algorithms in an effective and efficient manner. We also conducted extensive experiments on real world publicly available datasets.

2. **LITERATURE REVIEW**

In this we shows a simple approach for fake news detection using different classifier. This approach was implemented as a software system and tested against a data set of Facebook news posts. They were collected from three large Facebook pages each from the right and from the left, as well as three large mainstream political news pages (Politico, CNN, ABC News). They achieved classification accuracy of approximately 74%. Classification accuracy for fake news is slightly worse. This may be caused by the skewness of the dataset: only 4.9% of it is fake news. A framework based on different machine learning approach that deals with various problems including accuracy shortage, time lag and high processing time to handle thousands of tweets in 1 sec. Firstly, they have collected 400,000 tweets from HSpam14 dataset. Then they further characterize the 150,000 spam tweets and 250,000 non- spam tweets. They also derived some lightweight features along with the Top-30 words that are providing highest information gain from Bag-of-Words model. 4. They were able to achieve an accuracy of 91.65% and surpassed the existing solution by approximately18%. First proposed a novel ML fake news detection method which, by combining news content and social context features, outperforms existing methods in the literature, increasing its accuracy up to 78.8%. Second, they implemented their method within a Facebook Messenger Chabot and validate it with a real-world application, obtaining a fake news detection accuracy of 81.7%. Their goal was to classify a news item as reliable or fake; they first described the datasets they used for their test, then presented the content-based approach they implemented and the method they proposed to combine it with a social-based approach available in the literature. The resulting dataset is composed of 15,500 posts, coming from 32 pages (14 conspiracy pages, 18 scientific pages), with more than 2, 300, 00 likes by 900,000+ users. 8,923 (57.6%) posts are hoaxes and 6,577 (42.4%) are non-hoaxes. Develops a method for automating fake news detection on Twitter by learning to predict accuracy assessments in two credibility-focused Twitter datasets: CREDBANK, a crowd sourced dataset of accuracy assessments for events in Twitter, and PHEME, a dataset of potential rumour in Twitter and journalistic assessments of their accuracies. They apply this method to Twitter content sourced from BuzzFeed’s fake news dataset. A feature analysis identifies features that are most predictive for crowd sourced and journalistic accuracy assessments, results of which are consistent with prior work. They rely on identifying highly retweeted threads of conversation and use the features of these threads to classify stories, limiting this work’s applicability only to the set of popular tweets. Since the majority of tweets are rarely retweeted, this method therefore is only usable on a minority of Twitter conversation threads. In this paper aims to present an insight of characterization of news story in the modern diaspora combined with the differential content types of news story and its impact on readers. Subsequently, we dive into existing fake news detection approaches that are heavily based on text-based analysis, and also describe popular fake news datasets. We conclude the paper by identifying 4 key open research challenges that can guide future research. It is a theoretical Approach which gives Illustrations of fake news detection by analysing the psychological factors.

**3. METHODOLOGY**

This paper explains the system which is developed in three parts. The first part is static which works on machine learning classifier. We studied and trained the model with 4 different classifiers and chose the best classifier for final execution. The second part is dynamic which takes the keyword/text from user and searches online for the truth probability of the news. The third part provides the authenticity of the URL input by user. In this paper, we have used Python and its Sci-kit libraries [14]. Python has a huge set of libraries and extensions, which can be easily used in Machine Learning. Sci-Kit Learn library is the best source for machine learning algorithms where nearly all types of machine learning algorithms are readily available for Python, thus easy and quick evaluation of ML algorithms is possible. We have used Django for the web-based deployment of the model, provides client side implementation using HTML, CSS and JavaScript. We have also used Beautiful Soup (bs4), requests for online scrapping.

**4.System Architecture**

Static Search-

The architecture of Static part of fake news detection system is quite simple and is done keeping in mind the basic machine learning process flow. The system design is shown below and self- explanatory. The main processes in the design are-

DATA SET

Pre-processing the data

(tokenizing, stemming, lemmatization, stop word removal)

Feature Extraction

Training the classifier

Truthful opinion

Fake opinion

**5. IMPLEMENTATION**

**5.2 Datasets**

The datasets we used in this study are open source and freely available online. The data includes both fake and truthful news articles from multiple domains. The truthful news articles published contain true description of real world events, while the fake news websites contain claims that are not aligned with facts. The conformity of claims from the politics domain for many of those articles can be manually checked with fact checking websites such as politifact.com and snopes.com. We have used three different datasets in this study, a brief description of which is provided as follows. The dataset is available at Kaggle (hereafter referred to as DS2) which contains a total of 20,386 articles used for training and 5,126 articles used for testing. The dataset is built from multiple sources on the Internet. The articles are not limited to a single domain such as politics as they include both fake and true articles from various other domains. The second dataset is also available at Kaggle (hereafter referred to as DS3); it includes a total of 3,352 articles, both fake and true. The true articles are extracted from trusted online sources such as CNN, Reuters, the New York Times, and various others, while the fake news articles are extracted from untrusted news websites. The domains it covered include sports, entertainment, and politics. A combined dataset is the collection of articles from the three datasets (hereafter referred to as DS4). As the articles vary in nature in each dataset, the fourth dataset is created to evaluate the performance of algorithms on datasets which cover a wide array of domains in a single dataset.

**5.2 DATA COLLECTION AND ANALYSI**S

We can get online news from different sources like social media websites, search engine, homepage of news agency websites or the fact-checking websites. On the Internet, there are a few publicly available datasets for Fake news classification like BS Detector etc. Online news can be collected from different sources, such as news agency homepages, search engines, and social media websites. However, manually determining the veracity of news is a challenging task, usually requiring annotators with domain expertise who performs careful analysis of claims and additional evidence, context, and reports from authoritative sources. However, there are no agreed upon benchmark datasets for the fake news detection problem. Data gathered must be pre-processed- that is, cleaned, transformed and integrated before it can undergo training process [16]. The dataset that we used in this project from the Kaggle

Kaggle is the world largest data science community with powerful tools and resources to help you achieve your data science goals.

Here is the dataset description

**train.csv**: A full training dataset with the following attributes:

* **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**: a label that marks the article as potentially unreliable
  + 1: unreliable
  + 0: reliable

**test.csv**: A testing training dataset with all the same attributes at train.csv without the label.

**submit.csv**: A sample submission that you ca

**6. DEFINITIONS AND DETAILS**

**A. Pre-processing Data**

Social media data is highly unstructured – majority of them are informal communication with typos, slangs and bad-grammar etc. Quest for increased performance and reliability has made it imperative to develop techniques for utilization of resources to make informed decisions . To achieve better insights, it is necessary to clean the data before it can be used for predictive modelling. For this purpose, basic pre-processing was done on the News training data. This step was comprised of-

**Data Cleaning**:

While reading data, we get data in the structured or unstructured format. A structured format has a well-defined pattern whereas unstructured data has no proper structure. In between the 2 structures, we have a semi-structured format which is a comparably better structured than unstructured format. Cleaning up the text data is necessary to highlight attributes that we’re going to want our machine learning system to pick up on. Cleaning (or pre-processing) the data typically consists of a number of steps:

**a) Remove punctuation**

Punctuation can provide grammatical context to a sentence which supports our understanding. But for our vectorizer which counts the

number of words and not the context, it does not add value, so we remove all special characters E.g.: How are you ? ->How are you

**b) Tokenization**

Tokenizing separates text into units such as sentences or words. It gives structure to previously unstructured text.

**c) Remove stop words**

Stop words are common words that will likely appear in any text. They don’t tell us much about our data so we remove them. Eg: silver or

lead is fine for me-> silver, lead, fine.

**d) Stemming**

Stemming helps reduce a word to its stem form. It often makes sense to treat related words in the same way. It removes suffices, by a simple rule-based approach. It reduces the corpus of words but often the actual words get neglected. eg: Entitling, Entitled -> Entitle. Note: Some search engines treat words with the same stem as synonyms [18].

**B. Feature Generation**

We can use text data to generate a number of features like word count, frequency of large words, frequency of unique words, n-grams etc.

By creating a representation of words that capture their meanings, semantic relationships, and numerous types of context they are used in, we can enable computer to understand text and perform Clustering, Classification etc [19].

**Vectorizing Data:**

Vectorizing is the process of encoding text as integers i.e. numeric form to create feature vectors so that machine learning algorithms can understand our data.

**1. Vectorizing Data: Bag-Of-Words**

Bag of Words (BOW) or Count Vectorizer describes the presence of words within the text data. It gives a result of 1 if present in the sentence and 0 if not present. It, therefore, creates a bag of words with a document-matrix count in each text document.

**2. Vectorizing Data: N-Grams**

N-grams are simply all combinations of adjacent words or letters of length n that we can find in our source text. N grams with n=1 are

called unigrams. Similarly, bigrams (n=2), trigrams (n=3) and so on can also be used. Unigrams usually don’t contain much information as compared to bigrams and trigrams. The basic principle behind n-grams is that they capture the letter or word is likely to follow the given word. The longer the n-gram (higher n), the more context you have to work with [20].

**3. Vectorizing Data: TF-IDF**

It computes “relative frequency” that a word appears in a document compared to its frequency across all documents TF-IDF weight represents the relative importance of a term in the document and entire corpus [17]. TF stands for Term Frequency: It calculates how frequently a term appears in a document. Since, every document size varies, a term may appear more in a long sized document that a short one. Thus, the length of the document often divides Term frequency.

Note: Used for search engine scoring, text summarization, document clustering.

TF (t, d)

IDF stands for Inverse Document Frequency: A word is not of much use if it is present in all the documents. Certain terms like “a”, “an”, “the”, “on”, “of” etc. appear many times in a document but are of little importance. IDF weighs down the importance of these terms and increase the importance of rare ones. The more the value of IDF, the more unique is the word [17].

TF-IDF is applied on the body text, so the relative count of each word in the sentences is stored in the document matrix.

T f I d f (t, d) = T F (t, d) \* ID F(t)

Note: Vectorizers outputs sparse matrices. Sparse Matrix is a matrix in which most entries are 0 [21].

**7. Algorithms used for Classification**

**A. Logistic Regression**

It is a classification not a regression algorithm. It is used to estimate discrete values (Binary values like 0/1, yes/no, true/false) based on set of independent variables (s). In simple words, it predicts the probability of occurrence of an event by fitting data to a logit function. Hence, it is also known as logit regression. Since, it predicts the probability, its output values lie between 0 and 1 (as expected). Mathematically, the log odds of the outcome are modelled as a linear combination of the predictor variables [23]. Odds = p/(1-p) = probability of event occurrence / probability of not event occurrence

ln(odds) = ln(p/(1-p))

logit(p)=ln(p/(1-p)) = b0+b1X1+b2X2+b3X3...+ b k X k

1. **Decision Tree**

A decision tree is a flowchart-like tree structure where an internal node represents feature (or attribute), the branch represents a decision rule, and each leaf node represents the outcome. The topmost node in a decision tree is known as the root node. It learns to partition on the basis of the attribute value. It partitions the tree in recursively manner call recursive partitioning

nodes in a decision tree with its leaves to minimize the number of tests in the convinced path. Usually, the subtree raising is of a modest impact on the models of the decision tree. Typically, there is no exact way to predict an option's

utility, although it can be advisable to turn it off when the induction procedure takes longer because of the subtree's raising being relatively computationally complicate

1. **Random Forest**

Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees (DT) from a randomly selected subset of the training set and then It collects the votes from different decision trees to decide the final prediction. Random forests are a significant innovation of the bagging in which it forms a large group of de-correlated trees, and after that, take an average for them.

Random Forest enhanced on bagging through decreasing correlation between trees with no increase in the variance. In many situations, the random forest performance is like boosting in which they are simpler to be trained and tuned.

As a result, random forests are widespread algorithms that

are applied to various packages

1. **Gradient Boosting Classifier**

Gradient boosting classifiers are a group of machine learning algorithms that combine many weak learning models together to create a strong predictive model. Decision trees are usually used when doing gradient boosting. The objective of Gradient Boosting classifiers is to minimize the loss, or the difference between the actual class value of the training example and the predicted class value. It isn't required to understand the process for reducing the classifier's loss, but it operates similarly to [gradient descent](https://en.wikipedia.org/wiki/Gradient_descent) in a neural network.

**MANUAL TESTING**

Manual testing is the process of manually testing software for defects. It requires a tester to play the role of an end user whereby they use most of the application's features to ensure correct behaviour.

In the project we also uses manual testing that helps us to check the news is fake or real .

**8.IMPLEMENTATION STEPS**

A. Static Search Implementation-

In static part, we have trained and used 3 out of 4 algorithms for classification. They are decision tree, Random Forest and Logistic Regression.

Step 1.: Here, we have built all the classifiers for predicting the fake news detection. The extracted features are fed into different

classifiers. We have used decision tree, Logistic Regression, gradient boosting classifier, and Random forest classifiers. Each of the extracted features

was used in all of the classifiers.

Step 2.: Once fitting the model, we compared the f1 score and checked the confusion matrix.

Step 3: After fitting all the classifiers, 2 best performing models were selected as candidate models for fake news classification.

Step 4: We have performed parameter tuning by implementing Grid Search CV methods on these candidate models and chosen best

performing parameters for these classifiers.

Step 5: Finally selected model was used for fake news detection with the probability of truth.

Step 6: Our finally selected and best performing classifier was Logistic Regression which was then saved on disk. It will be used to classify the fake news.

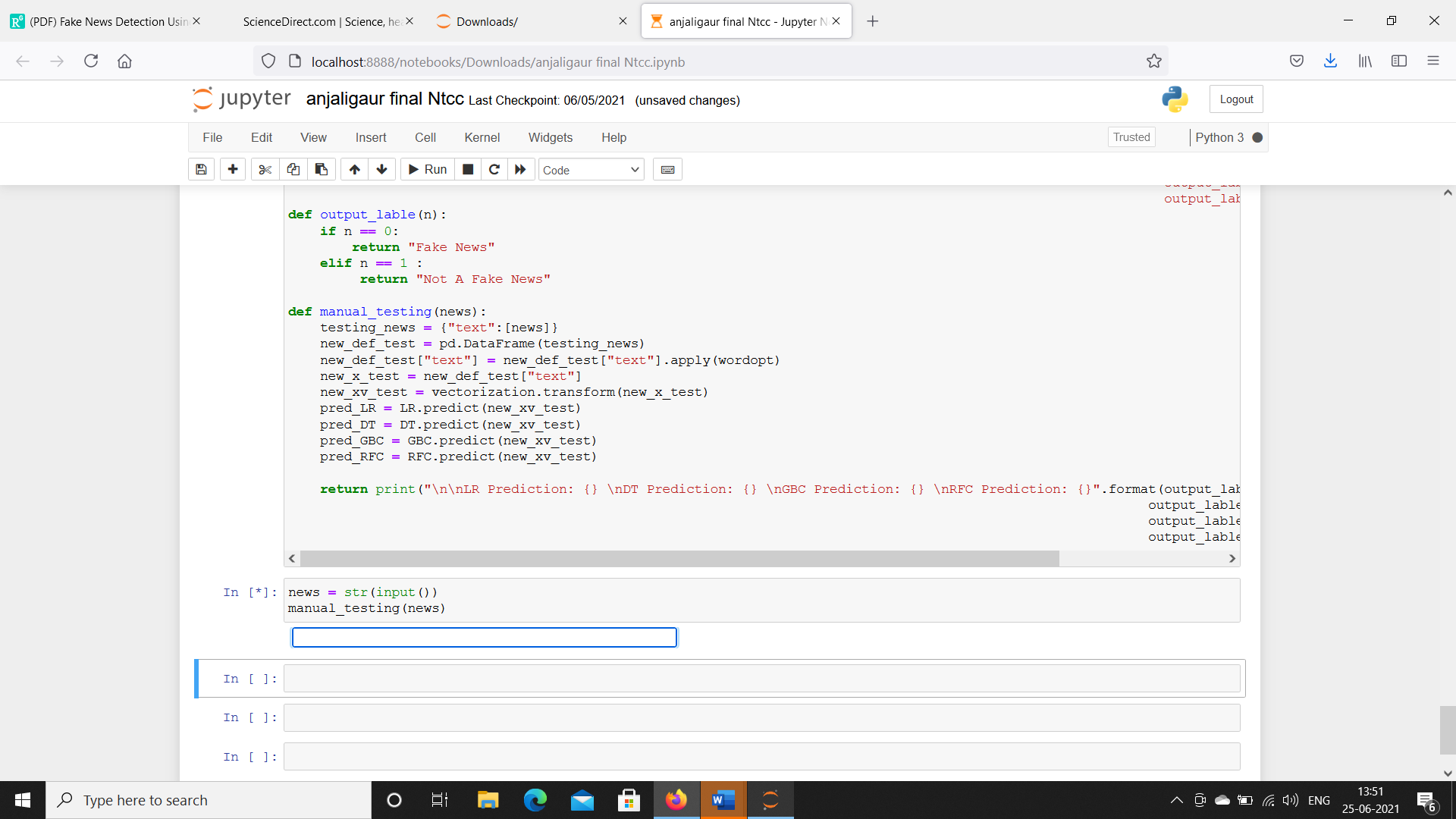
**9.FUTURE SCOPE**

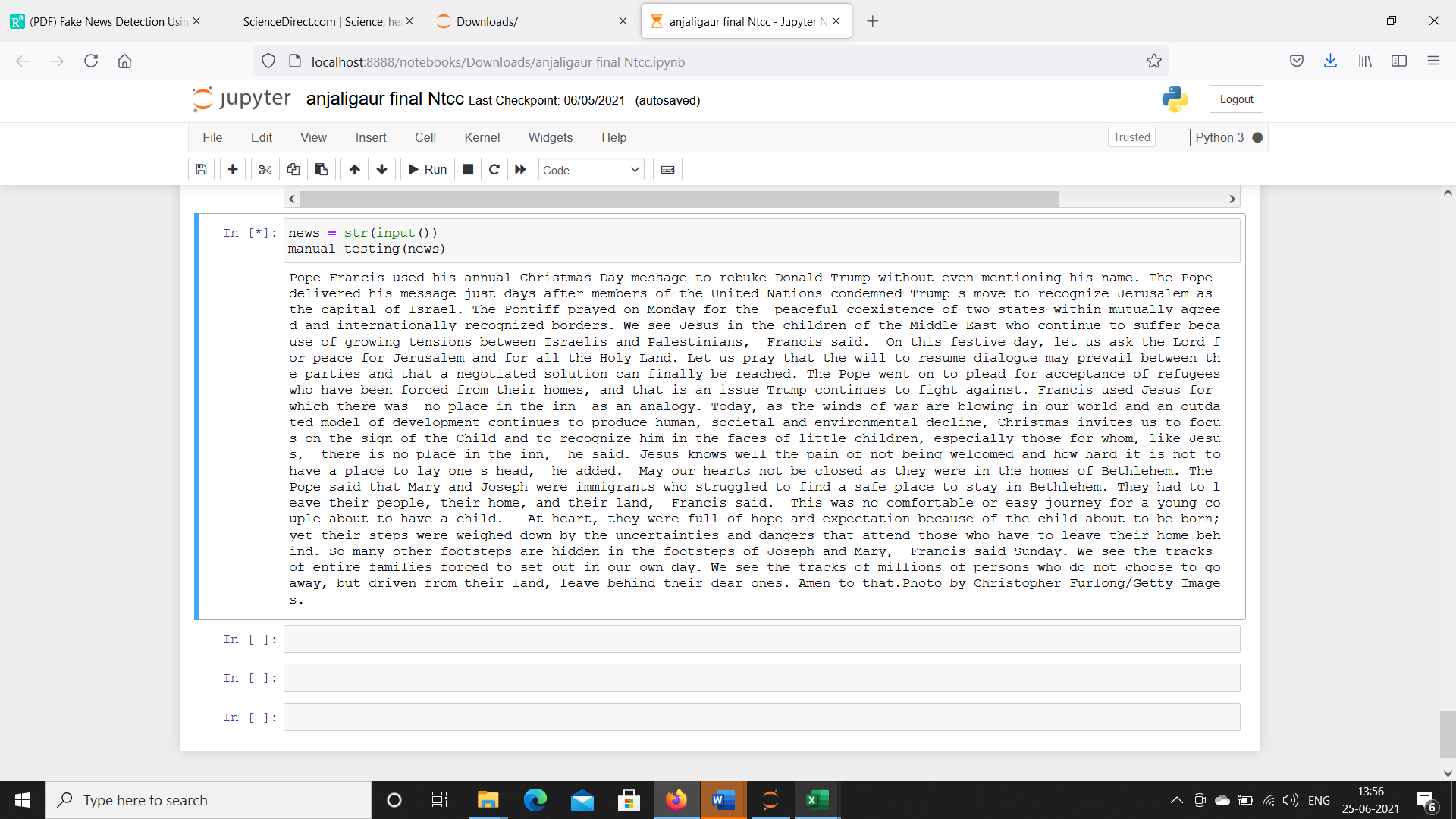
Once a source is labelled as a procedure of fake news . We can predict with high confidence that any future articles from that source will be fake news.

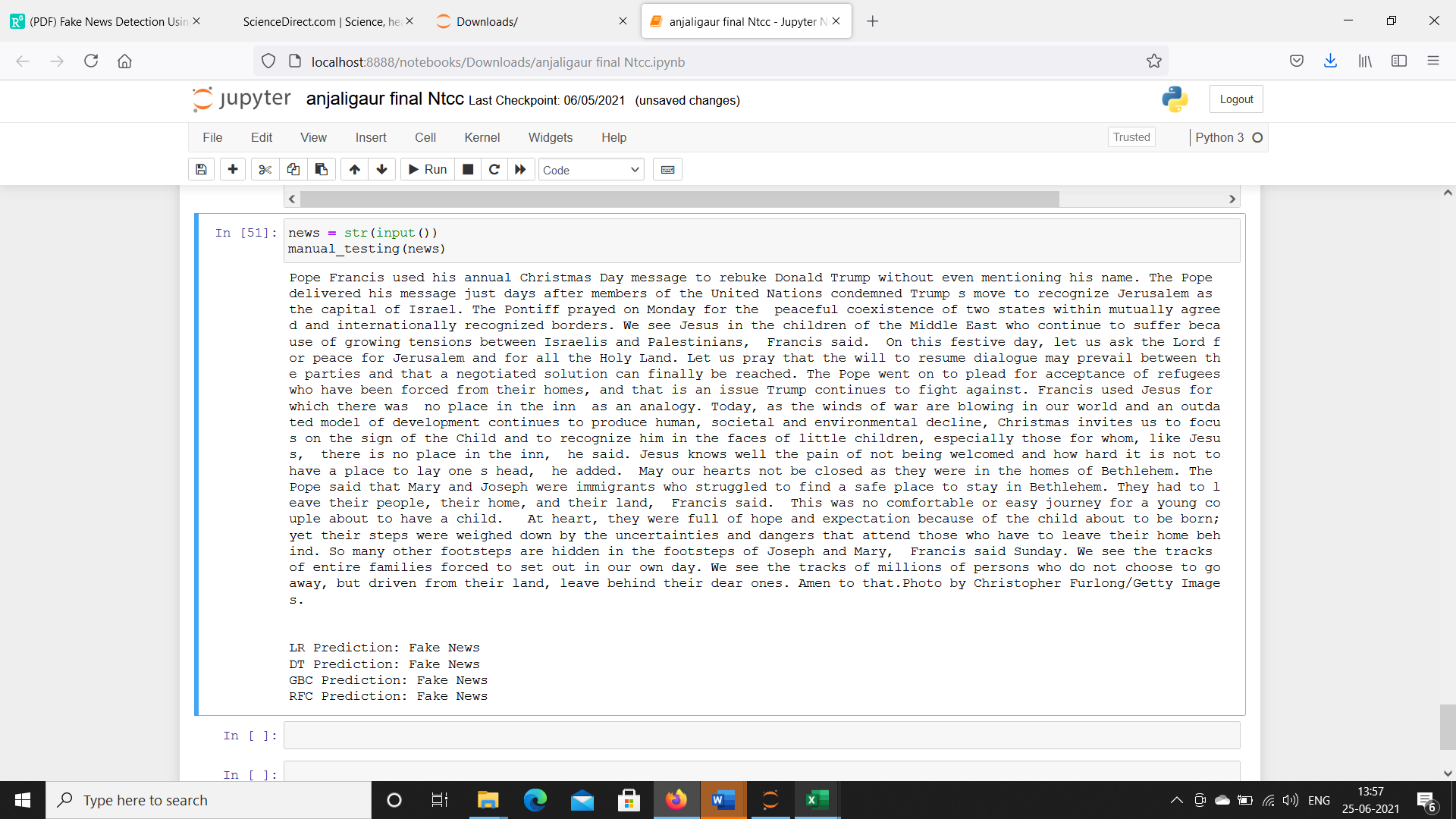
**10.RESULT**

Implementation was done using the above algorithms with Vector features- Count Vectors and T f -I d f vectors at Word level and N gram-level. Accuracy was noted for all models. We used K-fold cross validation technique to improve the effectiveness of the models.

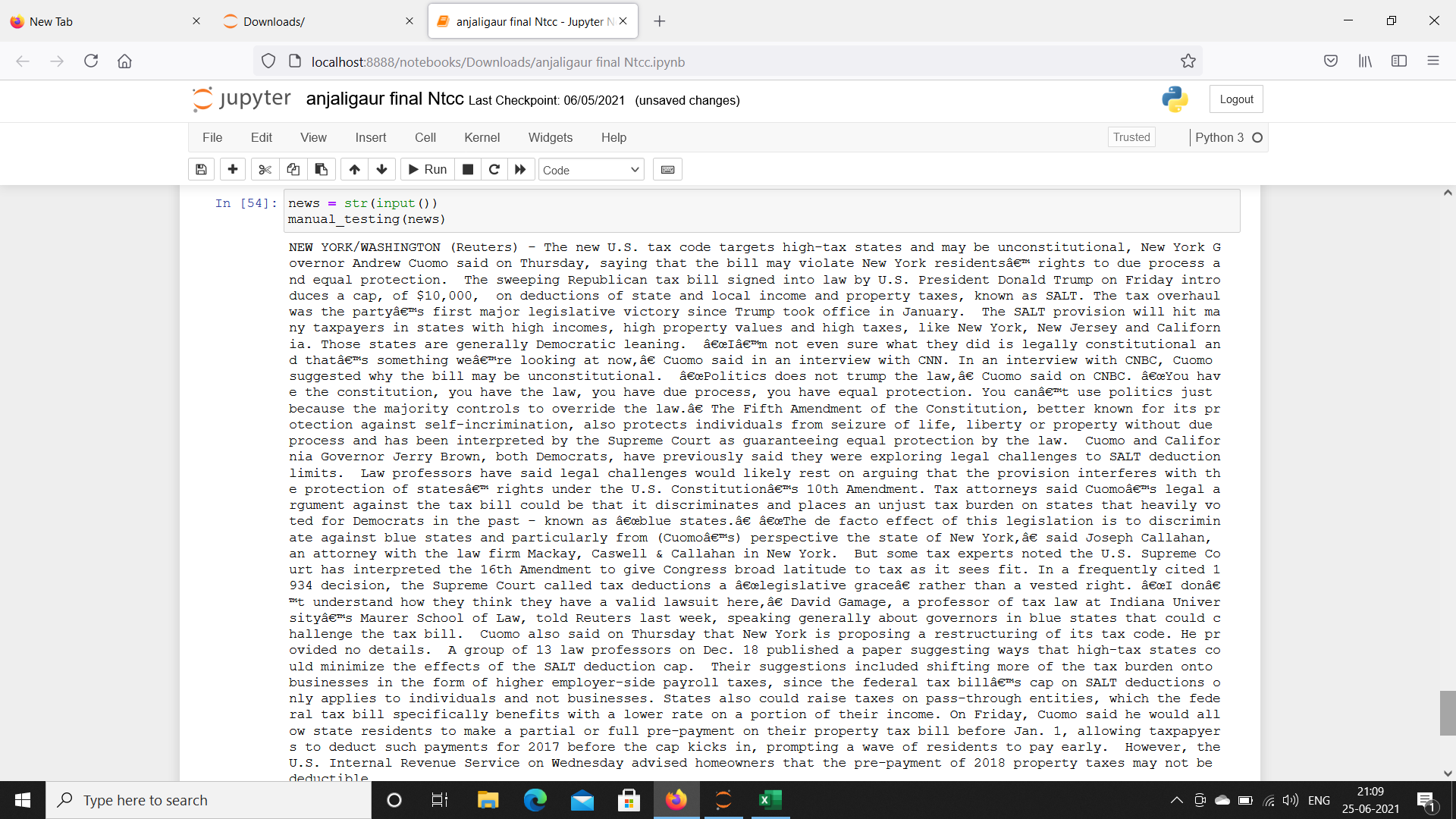
**10.1SNAPSHOTS OF SYSTEM WORKING**

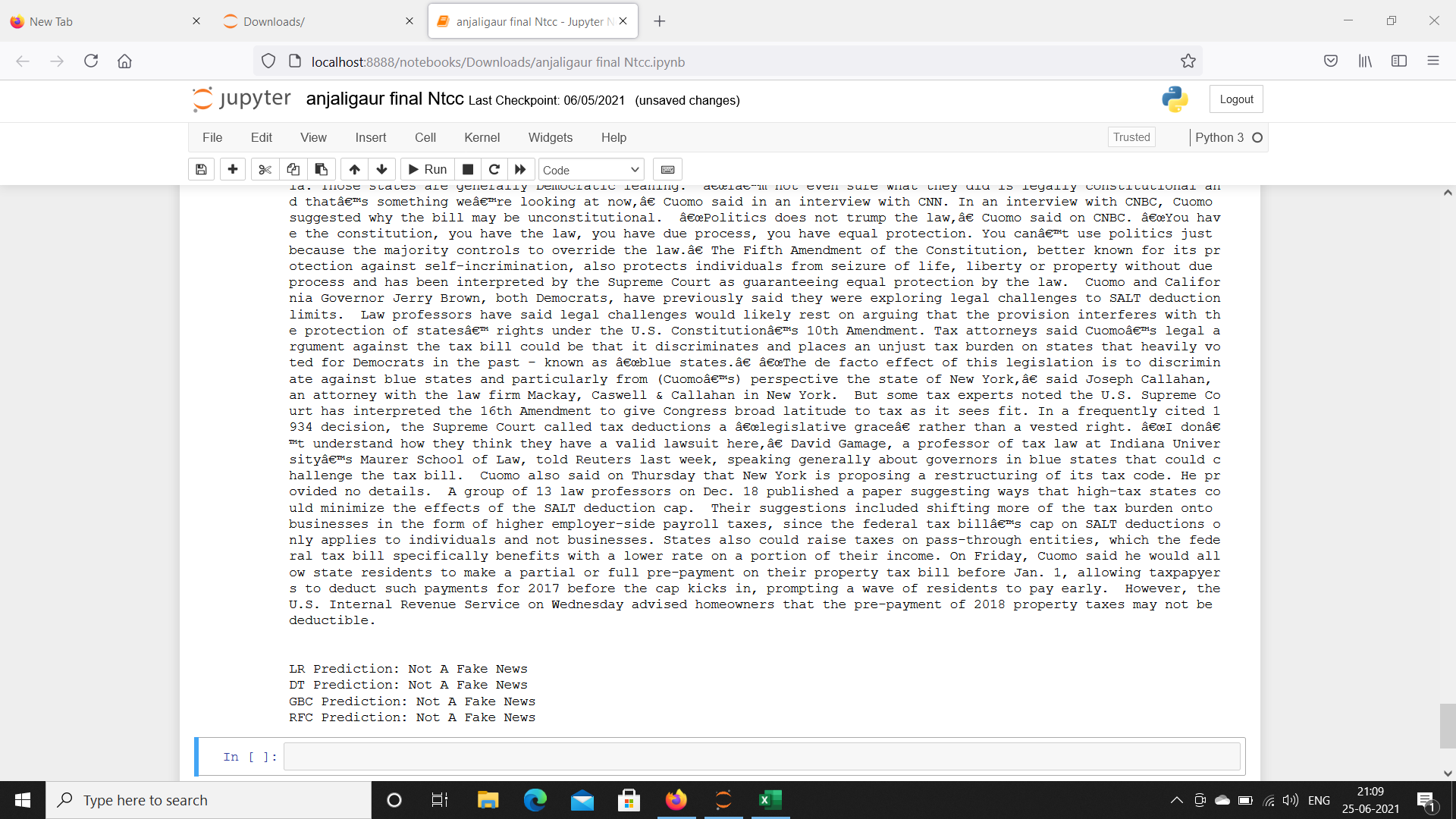
**Static System**





1. **Static output that it is a fake news**





1. **Static output is that it is not a fake news**

**11. CONCLUSION**

The task of classifying news manually requires in-depth knowledge of the domain and expertise to identify anomalies in the text. In this research, we discussed the problem of classifying fake news articles using machine learning models and ensemble techniques. The data we used in our work is collected from the World Wide Web and contains news articles from various domains to cover most of the news rather than specifically classifying political news. The primary aim of the research is to identify patterns in text that differentiate fake articles from true news. We extracted different textual features from the articles using an LIWC tool and used the feature set as an input to the models. The learning models were trained and parameter-tuned to obtain optimal accuracy. Some models have achieved comparatively higher accuracy than others. We used multiple performance metrics to compare the results for each algorithm. The ensemble learners have shown an overall better score on all performance metrics as compared to the individual learners.Fake news detection has many open issues that require attention of researchers. For instance, in order to reduce the spread of fake news, identifying key elements involved in the spread of news is an important step. Graph theory and machine learning techniques can be employed to identify the key sources involved in spread of fake news. Likewise, real time fake news identification in videos can be another possible future direction.

In the 21st century, the majority of the tasks are done online. Newspapers that were earlier preferred as hard - copies are now being substituted by applications like Facebook, Twitter, and news articles to be read online. WhatsApp’ s forwards are also a major source. The growing problem of fake news only makes things more complicated and tries to change or hamper the opinion and attitude of people towards use of digital technology. When a person is deceived by the real news two possible things happen- People start believing that their perceptions about a particular topic are true as assumed. Thus, in order to curb the phenomenon, we have developed our Fake news Detection system that takes input from the user and classify it to be true or fake. To implement this, various NLP and Machine Learning Techniques have to be used. The model is trained using an appropriate dataset and performance evaluation is also done using various performance measures. The best model, i.e. the model with highest accuracy is used to classify the news headlines or articles. As evident above for static search, our best model came out to be Logistic Regression with an accuracy of 65%. Hence we then used grid search parameter optimization to increase the performance of logistic regression which then gave us the accuracy of 75%. Hence we can say that if a user feed a particular news article or its headline in our model, there are 75% chances that it will be classified to its true nature. The user can check the news article or keywords online; he can also check the authenticity of the website. The accuracy for dynamic system is 93% and it increases with every iteration. We intend to build our own dataset which will be kept up to date according to the latest news. All the live news and latest data will be kept in a database using Web Crawler and online database.