A Major Project Report

On

FAKE NEWS DETECTION USING NLP

Project submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

BY

PANDIRI VIJAYALAXMI	**(18C91A0568)**
PEDDI ADITHYA RAM	**(18C91A0573)**
RAHUL SINGH	**(18C91A0578)**

Under the Esteemed guidance of
Dr. BIRRU DEVENDER M.Tech,Ph.D
Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE
(COLLEGE OF ENGINEERING)

(Approved by AICTE New Delhi, Permanently Affiliated to JNTU Hyderabad, Accredited by NAAC with 'A' Grade)
Bogaram (V), Keesara (M), Medchal District -501 301.

2021 - 2022

HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE

(COLLEGE OF ENGINEERING)

(Approved by AICTE New Delhi, Permanently Affiliated to JNTU Hyderabad, Accredited by NAAC with 'A' Grade)
Bogaram (V), Keesara (M), Medchal Dist-501301.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the major project entitled "FAKE NEWS DETECTION USING NLP" is being submitted by PANDIRI VIJAYALAXMI (18C91A0568), PEDDI ADITHYA RAM (18C91A0573), RAHUL SINGH (18C91A0578) in Partial fulfillment of the academic requirements for the award of the degree of Bachelor of Technology in "COMPUTER SCIENCE AND ENGINEERING" from HOLY MARY INSTITUTE OF TECHNOLOGY & SCIENCE, JNTU Hyderabad during the year 2021- 2022.

INTERNAL GUIDE

Dr. BIRRU DEVENDER M.Tech,Ph.D
Professor
Dept. of Computer Science & Engineering

HEAD OF THE DEPARTMENT

Dr B. NARSIMHA M.Tech,Ph.D
Professor & HoD
Dept. of Computer Science & Engineering

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible, who's constant guidance and encouragement crowns all effort with success.

We take this opportunity to express my profound gratitude and deep regards to our Guide **DR. BIRRU DEVENDER, Professor**, Dept. of Computer Science & Engineering, Holy Mary Institute of Technology & Science for his / her exemplary guidance, monitoring and constant encouragement throughout the project work.

Our special thanks to **Dr. B. Narsimha, Head of the Department**, Dept. of Computer Science & Engineering, Holy Mary Institute of Technology & Science who has given immense support throughout the course of the project.

We also thank **Dr**. **P. Bhaskara Reddy,** the **Honorable Director** of my college Holy Mary Institute of Technology & Science for providing me the opportunity to carry out this work.

At the outset, we express my deep sense of gratitude to the beloved **Chairman A. Siddartha Reddy** of **Holy Mary Institute of Technology & Science**, for giving me the opportunity to complete my course of work

We are obliged to **staff members** of Holy Mary Institute of Technology & Science for the valuable information provided by them in their respective fields. We are grateful for their cooperation during the period of my assignment.

Last but not the least we thank our **Parents**, and **Friends** for their constant encouragement without which this assignment would not be possible.

PANDIRI VIJAYALAXMI	**(18C91A0568)**
PEDDI ADITHYA RAM	**(18C91A0573)**
RAHUL SINGH	**(18C91A0578)**

DECLARATION

This is to certify that the work reported in the present project titled **"FAKE NEWS DETECTION USING NLP"** is a record of work done by us in the Department of Computer Science & Engineering, Holy Mary Institute of Technology and Science.

To the best of our knowledge no part of the thesis is copied from books / journals/ internet and wherever the portion is taken, the same has been duly referred to in the text. The reports are based on the project work done entirely by us not copied from any other source.

PANDIRI VIJAYALAXMI	**(18C91A0568)**
PEDDI ADITHYA RAM	**(18C91A0573)**
RAHUL SINGH	**(18C91A0578)**

INDEX

ABSTRACT

Name of the Chapter Page No.

LIST OF FIGURES

LIST OF ABBREVIATIONS

CV	Count Vectorizer
DBMS	Database management system
UML	Unified Modeling Language
NLP	Natural Language Processing
W2V	Word 2 Vecctor

LIST OF IMAGES

ABSTRACT

Fake News has become one of the major problem in the existing society. Fake News has high potential to change opinions, facts and can be the most dangerous weapon in influencing society. The proposed project uses NLP techniques for detecting the 'fake news', that is, misleading news stories which come from the non-reputable sources. By building a model based on a K-Means clustering algorithm, the fake news can be detected. The data science community has responded by taking actions against the problem. It is impossible to determine a news as real or fake accurately. So the proposed project uses the datasets that are trained using count vectorizer method for the detection of fake news and its accuracy will be tested using machine learning algorithms.

1. INTRODUCTION

1.1 MACHINE LEARNING AND NLP:

1.1.1 MACHINE LEARNING

Machine learning (ML) is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions, relying on patterns and inference instead. It is seen as a subset of artificial intelligence. Machine learning algorithms build a mathematical model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to perform the task. Machine learning is closely related to computational statistics, which focuses on making predictions using computers. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E." This is Alan Turing's definition of machine learning.

Deep learning is a class of machine learning algorithms that utilizes a hierarchical level of artificial neural networks to carry out the process of machine learning. The artificial neural networks are built like the human brain, with neuron nodes connected together like a web. While traditional programs build analysis with data in a linear way, the hierarchical function of deep learning systems enables machines to process data with a nonlinear approach

The word "deep" in "deep learning" refers to the number of layers through which the data is transformed. More precisely, deep learning systems have a substantial credit assignment path (CAP) depth. The CAP is the chain of transformations from input to output. CAPs describe potentially causal connections between input and output.

Deep learning architectures such as deep neural networks, deep belief networks, recurrent neural networks and convolutional neural networks have been applied to fields including computer vision, speech recognition, natural language processing, audio recognition, social network filtering, machine translation, bioinformatics, drug design, medical image analysis

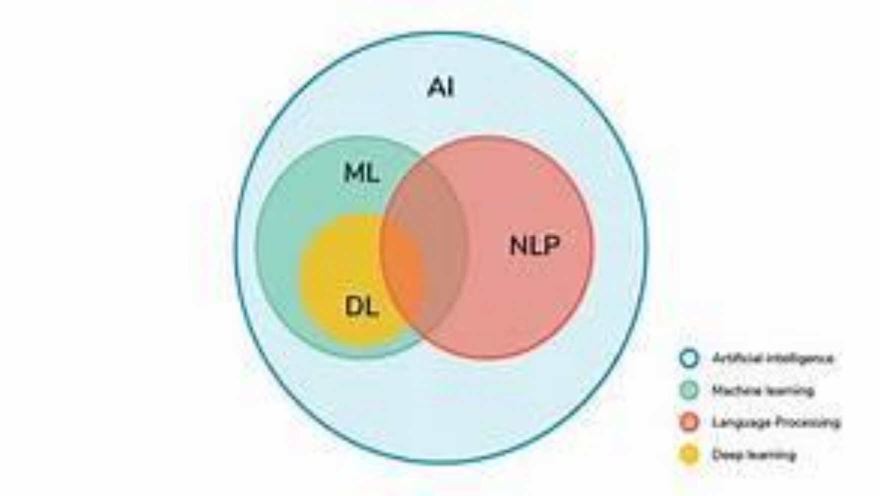


Fig. 1: Graphical representation of relationship between various fields in artificial intelligence

1.1.2 NATURAL LANGUAGE PROCESSING

NLP is an area of computer science and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to fruitfully process large amounts of natural language data.

Natural language processing (NLP) is a subfield of linguistics, computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language.

1.1.2.1 STAGES IN NLP

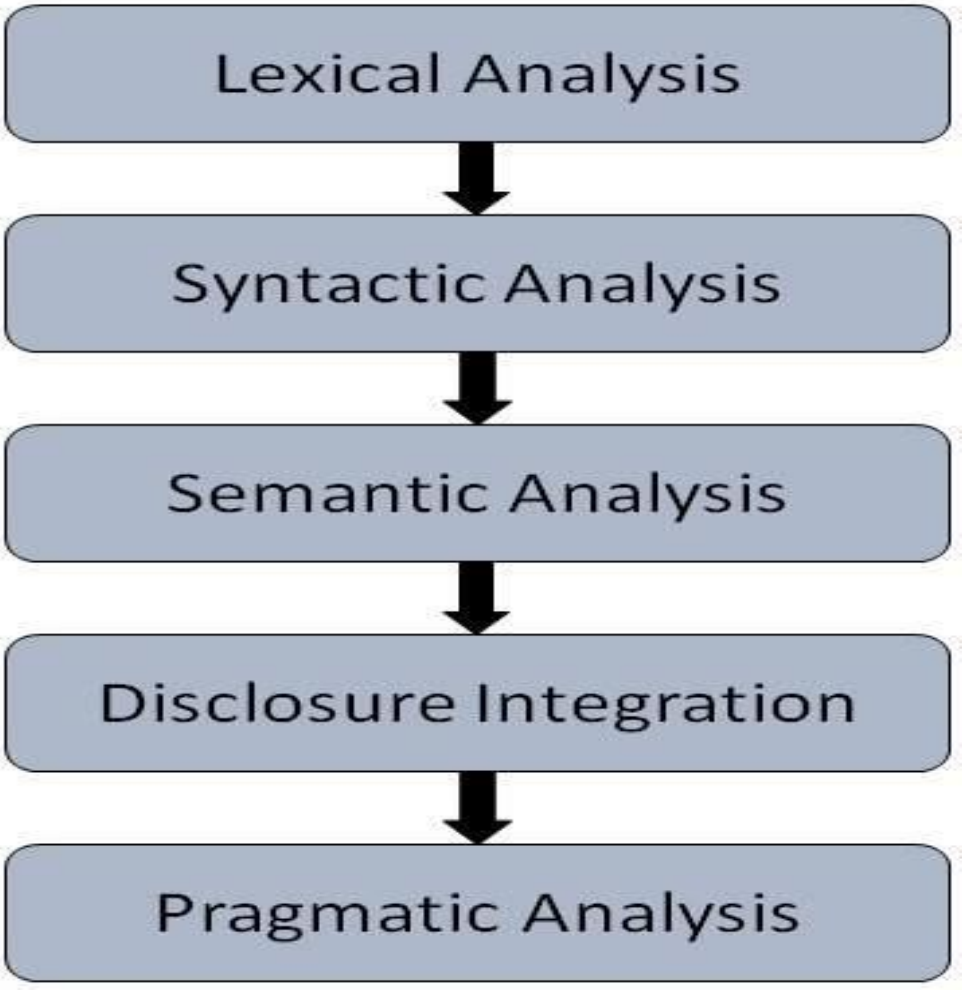


Fig 2: Stages in NLP

1.1.2.1.1 LEXICAL ANALYSIS

Lexical Analysis involves identifying and the structure of words. Lexicon of a language means the collection of words and phrases in a language. Lexical analysis is dividing the whole chunk of txt into paragraphs, sentences, and words.

1.1.2.1.2 SYNTACTIC ANALYSIS (PARSING)

Syntactic Analysis involves analysis of words in the sentence for grammar and arranging words in a manner that shows the relationship among the words. The sentence such as “The school goes to boy” is rejected by English syntactic analyzer.

1.1.2.1.3 SEMANTIC ANALYSIS

Semantic Analysis draws the exact meaning or the dictionary meaning from the text. The text is checked for meaningfulness. It is done by mapping syntactic structures and objects in the task domain. The semantic analyzer disregards sentence such as “hot ice-cream”

1.1.2.1.4 DISCOURSE INTEGRATION

The meaning of any sentence depends upon the meaning of the sentence just before it. In addition, it also brings about the meaning of immediately succeeding sentence. So in Discourse Integration gives the meaning based on all the sentences given before it. Eg. Consider the sentence “Water is flowing on the bank of the river” But bank has two meanings One Financial Institute and Two River of the bank here System has to consider the second meaning.

1.1.2.1.5 PRAGMATIC ANALYSIS

During this, what was said is re-interpreted on what it actually meant. It involves deriving those aspects of language which require real world knowledge.

1.2 MOTIVATION OF WORK

The rise of fake news during the 2016 U.S. Presidential Election highlighted not only the dangers of the effects of fake news but also the challenges presented when attempting to separate fake news from real news. Fake news may be a relatively new term but it is not necessarily a new phenomenon. Fake news has technically been around at least since the appearance and popularity of one-sided, partisan newspapers in the 19th century. However, advances in technology and the spread of news through different types of media have increased the spread of fake news today. As such, the effects of fake news have increased exponentially in the recent past and something must be done to prevent this from continuing in the future.

I have identified the three most prevalent motivations for writing fake news and chosen only one as the target for this project as a means to narrow the search in a meaningful way. The first motivation for writing fake news, which dates back to the 19th century one-sided party newspapers, is to influence public opinion. The second, which requires more recent advances in technology, is the use of fake headlines as clickbait to raise money. As such, this paper will focus primarily on fake news as defined by politifact.com, “fabricated content that intentionally masquerades as news coverage of actual events.” This definition excludes satire, which is intended to be humorous 8 and not deceptive to readers. Most satirical articles come from sources. Satire can already be classified, by machine learning techniques Therefore, our goal is to move beyond these achievements and use machine learning to classify, at least as well as humans, more difficult discrepancies between real and fake news.

The dangerous effects of fake news, as previously defined, are made clear by events in which a man attacked a pizzeria due to a widespread fake news article. This story along with analysis provide evidence that humans are not very good at detecting fake news, possibly not better than chance. As such, the question remains whether or not machines can do a better job.

There are two methods by which machines could attempt to solve the fake news problem better than humans. The first is that machines are better at detecting and keeping track of statistics than humans, for example it is easier for a machine to detect that the majority of verbs used are “suggests” and “implies” versus, “states” and “proves.” Additionally, machines may be more efficient in surveying a knowledge base to find all relevant articles and answering based on those many different sources. Either of these methods could prove useful in detecting fake news, but we decided to focus on how a machine can solve the fake news problem using supervised learning that extracts feature of the language and content only within the source in question, without utilizing any fact checker or knowledge base. For many fake news detection techniques, a “fake” article published by a trustworthy author through a trustworthy source would not be caught. This approach would combat those “false negative” classifications of fake news. In essence, the task would be equivalent to what a human faces when reading a hard copy of a newspaper article, without internet access or outside knowledge of the subject (versus reading something online where he can simply look up relevant sources). The machine, like the human in the coffee shop, will have only access to the words in the article and must use strategies that do not rely on blacklists of authors and sources. The current project involves utilizing machine learning and natural language processing techniques to create a model that can expose documents that are, with 9 high probability, fake news articles. Many of the current automated approaches to this problem are centered around a “blacklist” of authors and sources that are known producers of fake news. But, what about when the author is unknown or when fake news is published through a generally reliable source? In these cases it is necessary to rely simply on the content of the news article to make a decision on whether or not it is fake. By collecting examples of both real and fake news and training a model, it should be possible to classify fake news articles with a certain degree of accuracy.

1.3PROBLEM STATEMENT

News consumption is a double-edged sword. It enables the wide spread of “fake news”, i.e., low quality news with intentionally false information. The extensive spread of fake news has the potential for extremely negative impacts on individuals and society. Therefore, fake news detection has recently become an emerging research that is attracting tremendous attention. First, fake news is intentionally written to mislead readers to believe false information, which makes it difficult and nontrivial to detect based on news content. To develop a FAKE NEWS DETECTION system using natural language processing and its accuracy will be tested using machine learning algorithms. The algorithm must be able to detect fake news in a given scenario world.

Fig 3: Fake News Detection

1.4 OBJECTIVE

The objective of this project is to develop the method for detecting and classifying the news stories using Natural Language Processing. We gathered our data from kaggle, processed the text.

1.5 PROJECT GOAL

The goal of this project is to find the effectiveness and limitations of language-based techniques for detection of fake news through the use of machine learning algorithms. The outcome of this project should determine how much can be achieved in this task by analyzing patterns contained in the text and blind to outside information about the world.

1.6 EXISTING SYSTEM

Rubin, and Chen outlines several approaches that seem promising towards the aim of perfectly classify the misleading articles. They note that simple content-related n-grams and shallow parts-of-speech (POS) tagging have proven insufficient for the classification task, often failing to account for important context information. Rather, these methods have been shown useful only in tandem with more complex methods of analysis. Deep Syntax analysis using Probabilistic Context Free Grammars (PCFG) have been shown to be particularly valuable in combination with n-gram.

1.7 PROPOSED SYSTEM

In our project, The proposed system when subjected to a scenario of a set of news articles, the new articles are categorized as true or fake by the existing data available. This prediction is done by using the relationship between the words used in the article with one another. The proposed system contains a Word2Vec model for finding the relationship between the words and with the obtained information of the existing relations, the new articles are categorized into fake and real news.

In our project, The proposed system when subjected to a scenario of a set of news articles, the new articles are categorized as true or fake by the existing data available. This prediction is done by using the relationship between the words used in the article with one another. The proposed system contains a Word2Vec model for finding the relationship between the words and with the obtained information of the existing relations, the new articles are categorized into fake and real news.

1.7.1 Algorithm for Proposed system

- Step 1: Start
- Step 2: Input is collected from various sources and prepare a dataset.
- Step 3: Preprocessing of data is done and dataset is divided into 2 parts training and testing data.
- Step 4: Count vectorization technique is used to convert the train data into numerical.
- Step 5: K MEANS clustering algorithm is used to build the predictive model using the train data.
- Step 6: Confusion matrix is obtained.
- Step 7: Accuracy is calculated

1.8 ADVANTAGES OF PRPOSED SYSTEM

- Its low cost.
- Easy access, and rapid dissemination of information lead people to seek out and consume news from social media.
- Easy to identify that the given news is real or fake.

2. LITERATURE REVIEW

In the world of rapidly increasing technology, information sharing has become an easy task. There is no doubt that internet has made our lives easier and access to lots of information. This is an evolution in human history, but at the same time it unfocusses the line between true media and maliciously forged media. Today anyone can publish content – credible or not – that can be consumed by the world wide web. Sadly, fake news accumulates a great deal of attention over the internet, especially on social media. People get deceived and don't think twice before circulating such mis-informative pieces to the world. This kind of news vanishes but not without doing the harm it intended to cause. The social media sites like Facebook, Twitter, Whatsapp play a major role in supplying these false news. Many scientists believe that counterfeited news issue may be addressed by means of machine learning and artificial intelligence.

Various models are used to provide an accuracy range of 60-75%. Which comprises of Naive Bayes classifier Linguistic features based, Bounded decision tree model, SVM etc. The parameters that are taken in consideration do not yield high accuracy. The motive of this project is to increase the accuracy of detecting fake news more than the present results that are available. By fabricating this new model which will judge the counterfeit news articles on the basis of certain criteria like spelling mistake, jumbled sentences, punctuation errors, words used.

There are two categories of important researches in automatic classification of real and fake news up to now:

In the first category, approaches are at conceptual level, distinction among fake news is done for three types: serious lies (which means news is about wrong and unreal events or information like famous rumors), tricks (e.g. providing wrong information) and comics (e.g. funny news which is an imitation of real news but contain bizarre contents).

In the second category, linguistic approaches and reality considerations techniques are used at a practical level to compare the real and fake contents. Linguistic approaches try to detect text features like writing styles and contents that can help in distinguishing fake news. The main idea behind this technique is that linguistic behaviors like using marks, choosing various types of words or adding labels for parts of a lecture are rather unintentional, so they are beyond the author’s attention. Therefore, an appropriate intuition and evaluation of using linguistic techniques can reveal hoping results in detecting fake news.

Rubin studied the distinction between the contents of real and comic news via multilingual features, based on a part of comparative news (The Onion, and The Beaverton) and real news (The Toronto Star and The New York Times) in four areas of civil, science, trade and ordinary news. She obtained the best performance of detecting fake news with a set of features including unrelated, marking and grammar.

Balmas believe that the cooperation of information technology specialists in reducing fake news is very important. In order to deal with fake news, using data mining as one of the techniques has attracted many researchers. In data mining based approaches, data integration is used in detecting fake news. In the current business world, data are an ever-increasing valuable asset and it is necessary to protect sensitive information from unauthorized people. However, the prevalence of content publishers who are willing to use fake news leads to ignoring such endeavors. Organizations have invested a lot of resources to find effective solutions for dealing with clickbait effects.

2.1 PREVIOUS CONTRIBUTIONS

Shloka gilda presented concept approximately how NLP is relevant to stumble on fake information. They have used time period frequency-inverse record frequency (TFIDF) of bi- grams and probabilistic context free grammar (PCFG) detection. They have examined their dataset over more than one class algorithms to find out the great model. They locate that TFIDF of bi-grams fed right into a stochastic gradient descent model identifies non-credible resources with an accuracy of 77%.

Mykhailo granik proposed simple technique for fake news detection the usage of naïve Bayes classifier. They used buzzfeed news for getting to know and trying out the naïve Bayes classifier. The dataset is taken from facebook news publish and completed accuracy upto 74% on test set.

Cody buntain advanced a method for automating fake news detection on twitter. They applied this method to twitter content sourced from buzzfeed's fake news Dataset. Furthermore, leveraging non-professional, crowdsourced people instead of Journalists presents a beneficial and much less costly way to classify proper and fake Memories on twitter rapidly.

Marco L. Della offered a paper which allows us to recognize how social networks and gadget studying (ML) strategies may be used for faux news detection. They have used novel ML fake news detection method and carried out this approach inside a Facebook Messenger chatbot and established it with a actual-world application, acquiring a fake information detection accuracy of 81%.

Shivam B. Parikh 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.

Himank Gupta et. al. [10] gave a framework based on different machine learning approach that deals with various problems including accuracy shortage, time lag (BotMaker) 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-.

2.2 RELATED WORK

Spam Detection

The problem of detecting not-genuine sources of information through content based analysis is considered solvable at least in the domain of spam detection, spam detection utilizes statistical machine learning techniques to classify text (i.e. tweets or emails) as spam or legitimate. These techniques involve pre-processing of the text, feature extraction (i.e. bag of words), and feature selection based on which features lead to the best performance on a test dataset. Once these features are obtained, they can be classified using Nave Bayes, Support Vector Machines, TF-IDF, or K-nearest neighbors classifiers. All of these classifiers are characteristic of supervised machine learning, meaning that they require some labeled data in order to learn the function.

$$f(message, \theta) = \left\{ \begin{array}{cc} \mathrm{C_{spam}} & \text{if classified as spam} \\ \mathrm{C_{leg}} & \text{otherwise} \end{array} \right\}$$

where, m is the message to be classified and is a vector of parameters and Cspam and Cleg are respectively spam and legitimate messages. The task of detecting fake news is similar and almost analogous to the task of spam detection in that both aim to separate examples of legitimate text from examples of illegitimate, ill-intended texts

Stance Detection

The goal of this contest was to encourage the development of tools that may help human fact checkers identify deliberate misinformation in news stories through the use of machine learning, natural language processing and artificial intelligence. The organizers decided that the first step in this overarching goal was understanding what other news organizations are saying about the topic in question. As such, they decided that stage one of their contest would be a stance detection competition. More specifically, the organizers built a dataset of headlines and bodies of text and challenged competitors to build classifiers that could correctly label the stance of a body text, relative to a given headline, into one of four categories: “agree”, “disagree”, “discusses” or “unrelated.” The top three teams all reached over 80% accuracy on the test set for this task. convolutional neural network.

2.3 SUMMARY

In this project, we will focus on text-based news and try to build a model that will help us to identify if a piece of given news is fake or real**.**

3. SOFTWARE REQUIREMENTS AND SPECIFICATIONS

3.1 SOFTWARE REQUIREMENTS

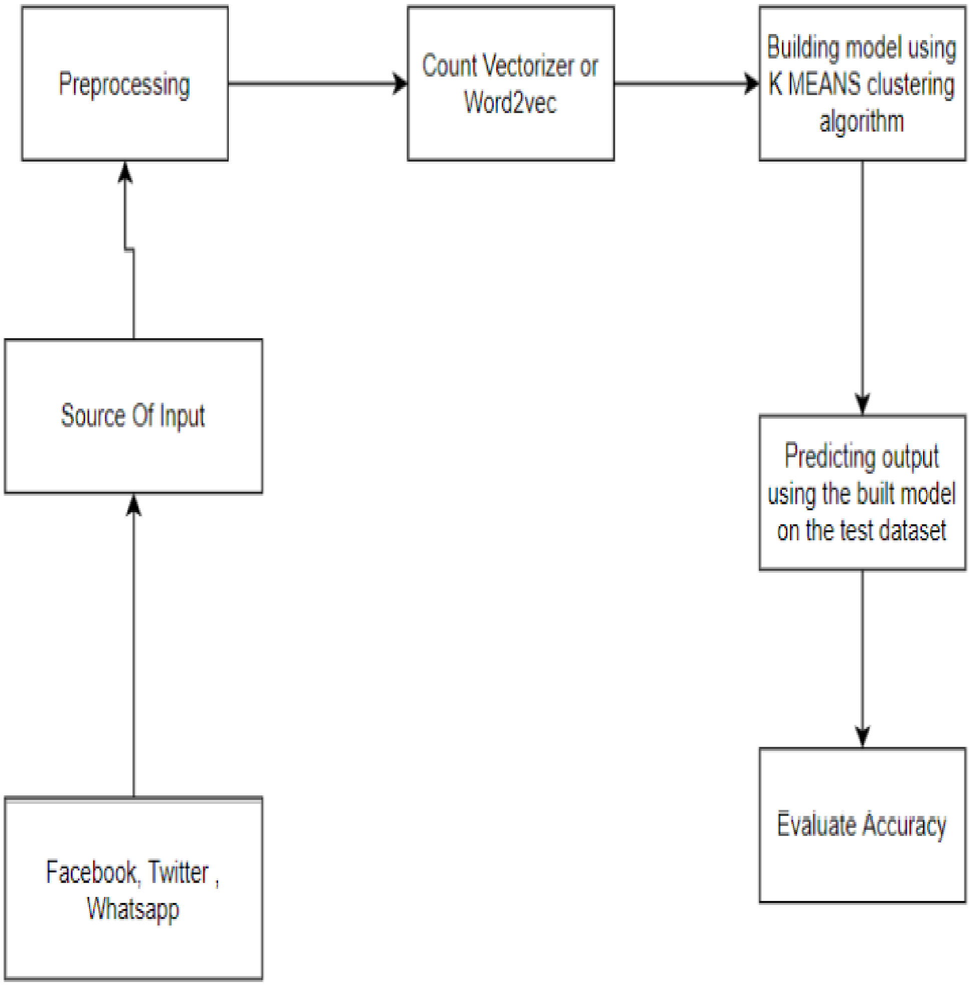
- Python 3.5 in Google Colab is used for data pre-processing, model training and prediction.
- Operating System : Windows 7 and above or Linux based OS or MAC OS.
- Database : Dataset

3.2 HARDWARE REQUIREMENTS

RAM	:	4 GB
Hard Disk	:	500 GB
CPU	:	2 GHz or faster
System Type	:	32-bit or 64-bit
Input device	:	Keyboard, Mouse.
Output device	:	Monitor.

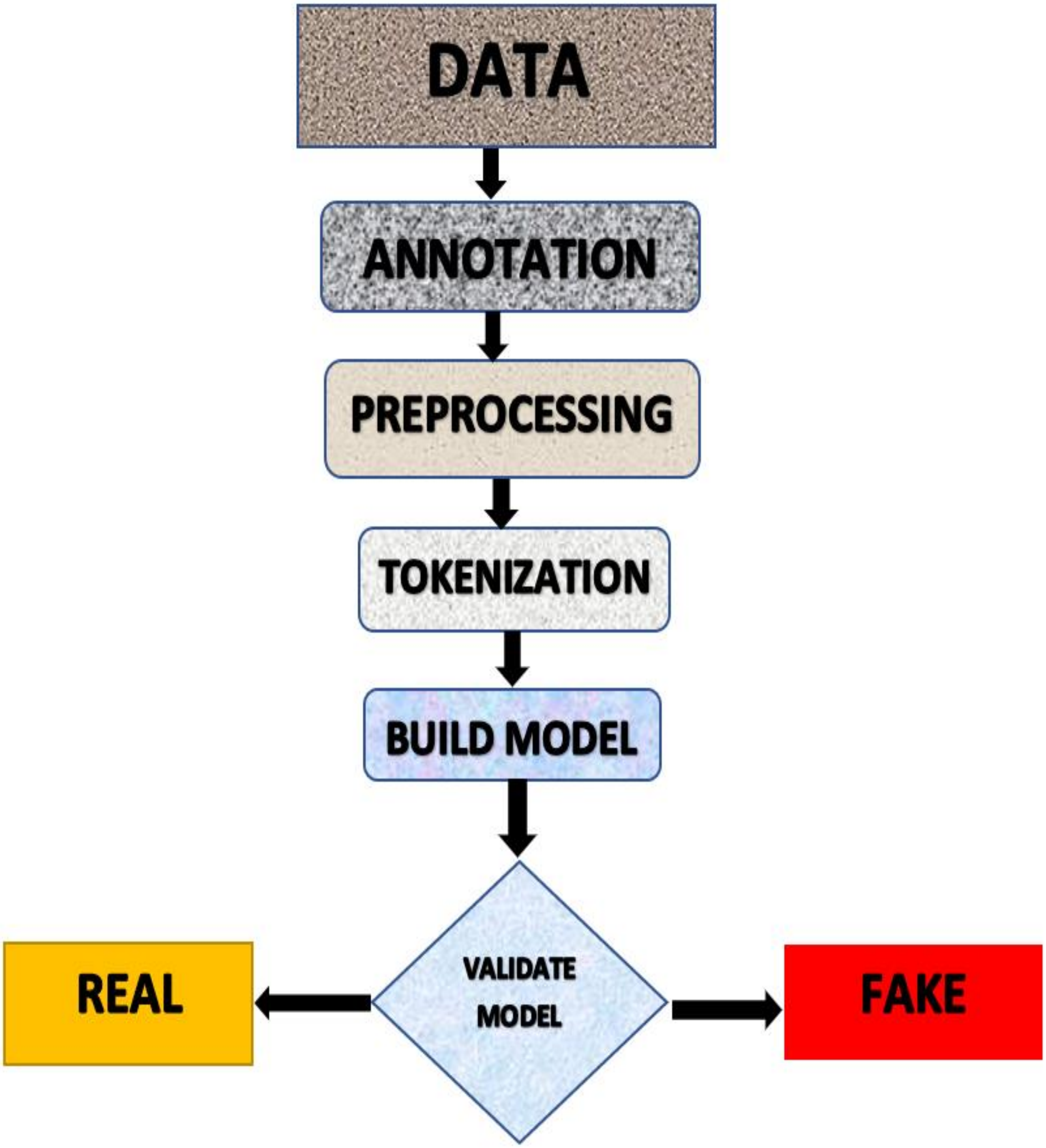
4. SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE


Preprocessing
Count Vectorizer or Word2vec
Building model using K MEANS clustering algorithm
Source Of Input
Predicting output using the built model on the test dataset
Facebook, Twitter , Whatsapp
Evaluate Accuracy

Input is collected from various sources such as newspapers, social media and stored in datasets. System will take input from datasets. The datasets undergo preprocessing and the unnecessary information is removed from it and the data types of the columns are changed if required. Jupyter notebook and python libraries are used in the above step. Count vectorizer technique is used in the initial step. For fake news detection, we have to train the system using dataset. Before entering to the detection of fake news, entire dataset is divide into two datasets. 80% is used for training and 20% is used for testing. During training, K-Means algorithm is used to train the model using the train dataset. In testing, the test dataset is given as input and the output is predicted. After the testing time, The predicted output and the actual output are compared using confusion matrix obtained. The confusion matrix gives the information regarding the number of correct and wrong predictions in the case of real and fake news. The accuracy is calculated by the equation No Of Correct Predictions/Total Test Dataset Input Size.

4.2 DATA FLOW DAIGRAM



4.3 DESIGN GOALS

To make the project runs smoothly it's required that we make plan and design some accepts like flowcharts and system architecture which are defined below

Data Collection

Data collection is one of the important and basic thing in our project. The right dataset must be provided to get robust results. We will be taking and analyzing data from Kaggle. After that seeing the accuracy we will use the data in our model.

Data Preprocessing

Human can understand any type of data but machine can't our model will also learn from scratch so it's better to make the data more machine readable. Raw data is usually inconsistent or incompleteff,l. Data preprocessing involves checking missing values, splitting the dataset and training the machine etc.

Training Model

Similar to feeding somethings, machine/model should also learn by feeding and learning on data. The data set extracted from Kaggle will be used to train the model. The training model uses a raw set of data as the undefined dataset which is collected from the previous fiscal year and from the same dataset a refine view is presented which is seen as the desired output. For the refining of the dataset various algorithms are implemented to show the desired output.

4.5 CONCEPTS

4.5.1 PREPROCESSING

In any Machine Learning process, Data Preprocessing is that step in which the data gets transformed, or Encoded, to bring it to such a state that now the machine can easily parse it. In other words, the features of the data can now be easily interpreted by the algorithm.

In this fake news detection, pre processing is the major thing that should be done . Firstly , as the data dataset is collected from various sources unnecessary information should be removed ,converted to lower case , remove punctuation , symbols , stop words.

4.5.2 STEPS IN TEXT PRE-PROCESSING

TEXT NORMALIZATION

Text normalization is a process of transforming text into a single canonical form. Normalizing text before storing or processing it allows for separation of required data from the rest so that the system can send consistent data as an input to the other steps of the algorithm.

STOP WORD REMOVAL

Stop Word: A Stop Word is a commonly used word in any natural language such as “a, an , the, for, is, was, which, are, were, from, do, with, and, so, very, that, this, no, yourselves etc....”.

These Stop Words will have a very high frequency and so these should be eliminated while calculating the term frequency so that the other important things are given priority. Stop word removal is such a Pre-processing step which removes these stop words and thereby helping in the further steps and also reducing some processing time because the size of the document decreases tremendously.

Consider a Sentence

“This is a sample sentence, showing off the stop word removal”.

Output after Stop word removal is:

[“sample”, “sentence”, “showing”, “stop”, “word”, “removal”]

Note: Though Stop words refer to the most commonly used words in a particular language, there is no single universal list of stop words, different tools uses different stop words.

STEMMING

Stemming is a pre-processing step in Text Mining applications as well as a very common requirement of Natural Language processing functions. In fact it is very important in most of the Information Retrieval systems. The main purpose of stemming is to reduce different grammatical forms / word forms of a word like its noun, adjective, verb, adverb etc. to its root form. The goal of stemming is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form.

Eg: A stemmer for English should identify the strings "cats", "catlike", "catty" as based on the root "cat".

RULES OF SUFFIX STRIPPING STEMMERS:

1.If the word ends in 'ed', remove the 'ed'.

2.If the word ends in 'ing', remove the 'ing'.

3.If the word ends in 'ly', remove the 'ly'.

RULES OF SUFFIX SUBSTITUTION STEMMERS

1.If the word ends in ‘ies’ substitute ‘ies’ with ‘y’.

Generally this stemmer is used because of some word like families etc

4.5.3 COUNT VECTORIZER

Count Vectorizer tokenizes (tokenization means breaking down a sentence or paragraph or any text into words) the text along with performing very basic preprocessing like removing the punctuation marks, converting all the words to lowercase, etc. The vocabulary of known words is formed which is also used for encoding unseen text later. An encoded vector is returned with a length of the entire vocabulary and an integer count for the number of times each word appeared in the document.

4.5.4 WORD2VEC MODEL

WORD2VEC is a class of models that represents a word in a large text corpus as a vector in n dimensional space bringing similar words closer to each other. Embeddings learned through word2vec have proven to be successful on a variety of downstream natural language processing tasks. The context of a word can be represented through a set of skip-gram pairs of (target_word,context_word)where context_word appears in the neighboring context of target_word.

WORD2VEC ALGORITHM

1.Take a sentence as input.

2.Consider a window size.

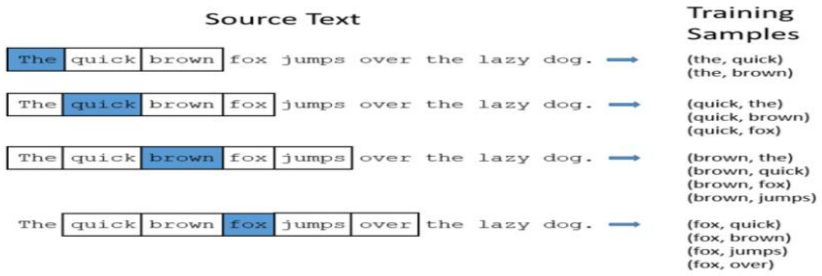
3.For every word in the sentence

    1.Consider current word as context.

    2.Other words in the window to the left and right of the word as targets and form (context,target) pair.

    3.From the pre defined vocabulary in the tensorflow library , the position of the context and target are found and then those values are applied in this formula

    4.The output is sent to the sigmoid function to result in the range [-1,1]



$$W_{ij} = \frac{\Sigma_p (x_i - \overline{x_i})(x_j - \overline{x_j})}{\sqrt{\Sigma_p (x_i - \overline{x_i})^2}\sqrt{\Sigma_p (x_j - \overline{x_j})^2}}$$

4.5.5 K-MEANS ALGORITHM

K-means clustering is one of the simplest and popular unsupervised machine learning algorithms. Typically, unsupervised algorithms make inferences from datasets using only input vectors without referring to known, or labelled, outcomes. A cluster refers to a collection of data points aggregated together because of certain similarities, You’ll define a target number k, which refers to the number of centroids you need in the dataset. A centroid is the imaginary or real location representing the center of the cluster. In other words, the K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible. The ‘means’ in the K-means refers to averaging of the data; that is, finding the centroid. To process the learning data, the K-means algorithm in machine learning starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster, and then performs iterative (repetitive) calculations to optimize the positions of the centroids

It halts creating and optimizing clusters when

The centroids have stabilized — there is no change in their values because the clustering has been successful.

4.5.6 EVALUATION MEASURES

Whenever we build Machine Learning models, we need some form of metric to measure the goodness of the model. Bear in mind that the “goodness” of the model could have multiple interpretations, but generally when we speak of it in a Machine Learning context we are talking of the measure of a model's performance on new instances that weren’t a part of the training data.

Determining whether the model being used for a specific task is successful depends on 2 key factors

1. Whether the evaluation metric we have selected is the correct one for our problem

2. If we are following the correct evaluation process

In this article, I will focus only on the first factor — Selecting the correct evaluation metric

DIFFERENT TYPES OF EVALUATION METRICS

The evaluation metric we decide to use depends on the type of NLP task that we are doing. To further add, the stage the project is at also affects the evaluation metric we are using. For instance, during the model building and deployment phase, we’d more often than not use a different evaluation metric to when the model is in production. In the first 2 scenarios, ML metrics would suffice but in production, we care about business impact, therefore we’d rather use business metrics to measure the goodness of our model.

With that being said, we could categorize evaluation metrics into 2 buckets.

- Intrinsic Evaluation — Focuses on intermediary objectives (i.e. the performance of an NLP component on a defined subtask)
- Extrinsic Evaluation — Focuses on the performance of the final objective (i.e. the performance of the component on the complete application)

4.5.7 DEFINING THE METRICS

Some common intrinsic metrics to evaluate NLP systems are as follows:

- **ACCURACY**

Whenever the accuracy metric is used, we aim to learn the closeness of a measured value to a known value. It’s therefore typically used in instances where the output variable is categorical or discrete — Namely a classification task.

- **PRECISION**

In instances where we are concerned with how exact the model's predictions are we would use Precision. The precision metric would inform us of the number of labels that are actually labeled as positive in correspondence to the instances that the classifier labeled as positive.

- **RECALL**

Recall measures how well the model can recall the positive class (i.e. the number of positive labels that the model identified as positive.

5. IMPLEMENTATON

5.1 ENVIRONMENTAL SETUP

Tools:

The following tools which are used to develop the ML application

1. Jupyter:

Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. Uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

2. Umbrello:

Umbrello is a Unified Modelling Language (UML) modeling tool and code generator. It can create industry-standard UML format,and can also generate code from UML diagrams in a variety of programming languages

- we need to install and setup the IDE
- after installing we need to set the path in environmental variables
- the process for installing is as belo

Sofware installation procedure

Step 1) Installing Anaconda

1. Install **Anaconda** from https://repo.anaconda.com/archive/Anaconda32021.05-Windowshttps://repo.anaconda.com/archive/Anaconda3-2021.05-Windows-x86_64.exex86_64.exe.
2. After opening link u can see this download

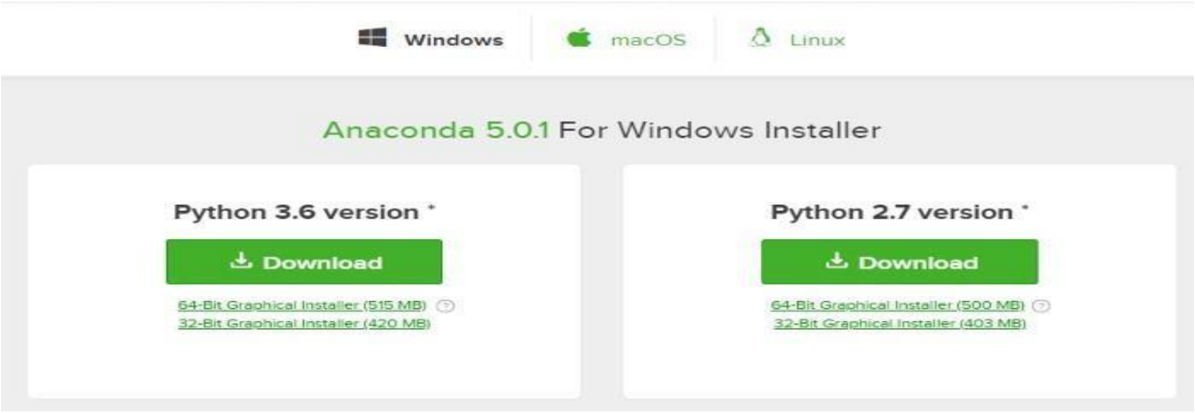
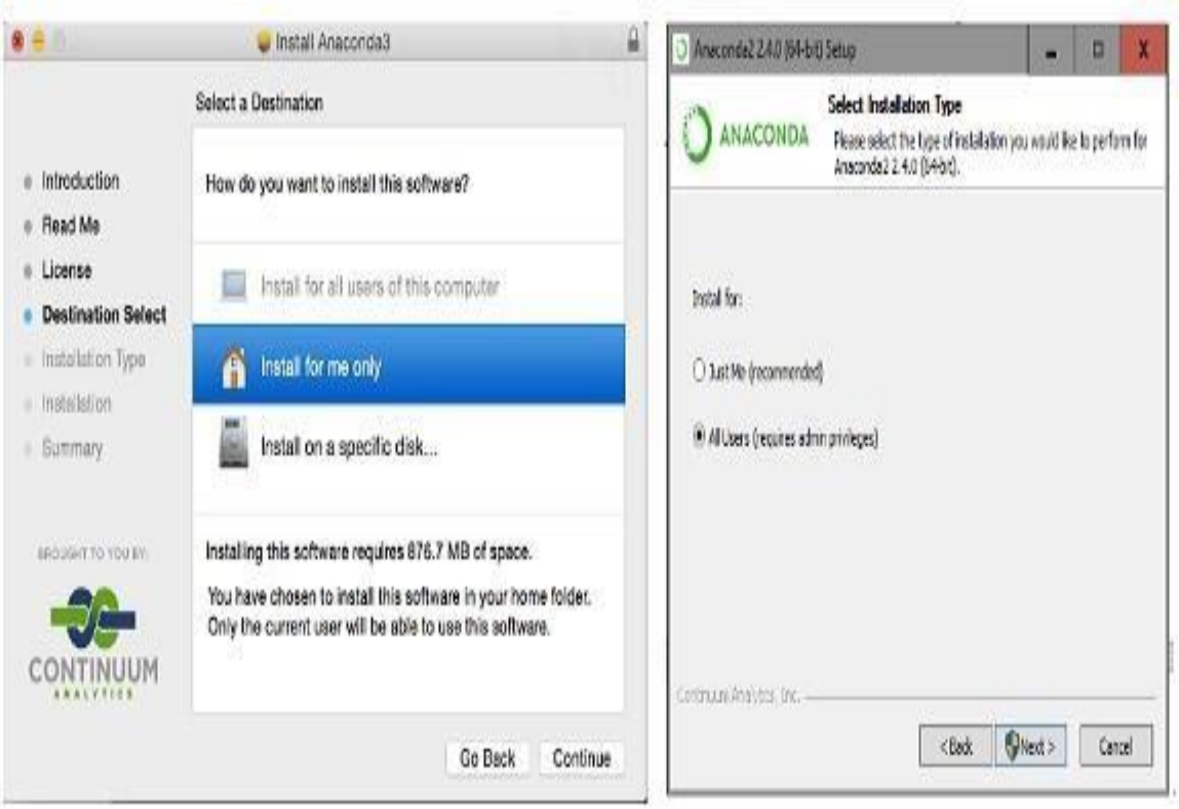

Windows
macOS
Linux
Anaconda 5.0.1 For Windows Installer
Python 3.6 version *
Download
64-Bit Graphical Installer (515 MB)
32-Bit Graphical Installer (420 MB)
Python 2.7 version *
Download
64-Bit Graphical Installer (500 MB)
32-Bit Graphical Installer (403 MB)

Fig 5.1.1

Downloading Anaconda Navigator

3. click on the download option in above image.

4. after downloading start installation.



Select a Destination

How do you want to install this software?

Install for all users of this computer

Install for me only

Install on a specific disk...

Installing this software requires 876.7 MB of space.

You have chosen to install this software in your home folder. Only the current user will be able to use this software.

Select Installation Type

Please select the type of installation you would like to perform for Anaconda2 2.4.0 (64-bit).

Install for:

○ Just Me (recommended)

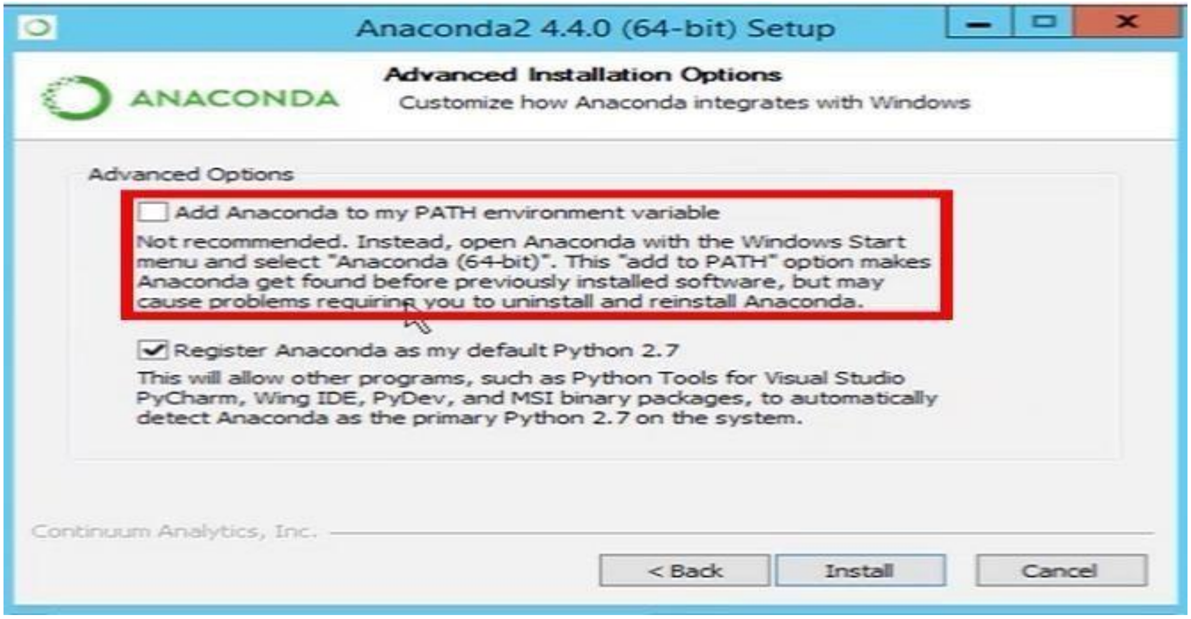
● All Users (requires admin privileges)

Fig 5.1.2

Installation Anaconda Navigator

5. The default options when prompted during the installation of Anaconda as shown above.

6. Ensure that the path to the folder where Anaconda is installed is added to your computer/system.



Advanced Installation Options

Customize how Anaconda integrates with Windows

Advanced Options

□ Add Anaconda to my PATH environment variable

Not recommended. Instead, open Anaconda with the Windows Start menu and select "Anaconda (64-bit)". This "add to PATH" option makes Anaconda get found before previously installed software, but may cause problems requiring you to uninstall and reinstall Anaconda.

☑ Register Anaconda as my default Python 2.7

This will allow other programs, such as Python Tools for Visual Studio PyCharm, Wing IDE, PyDev, and MSI binary packages, to automatically detect Anaconda as the primary Python 2.7 on the system.

Fig 5.1.3

Anaconda installation Options

7. Open “Anaconda Prompt” by finding it in the Windows (Start) Menu.

8. Type the command in red to verified Anaconda was installed.

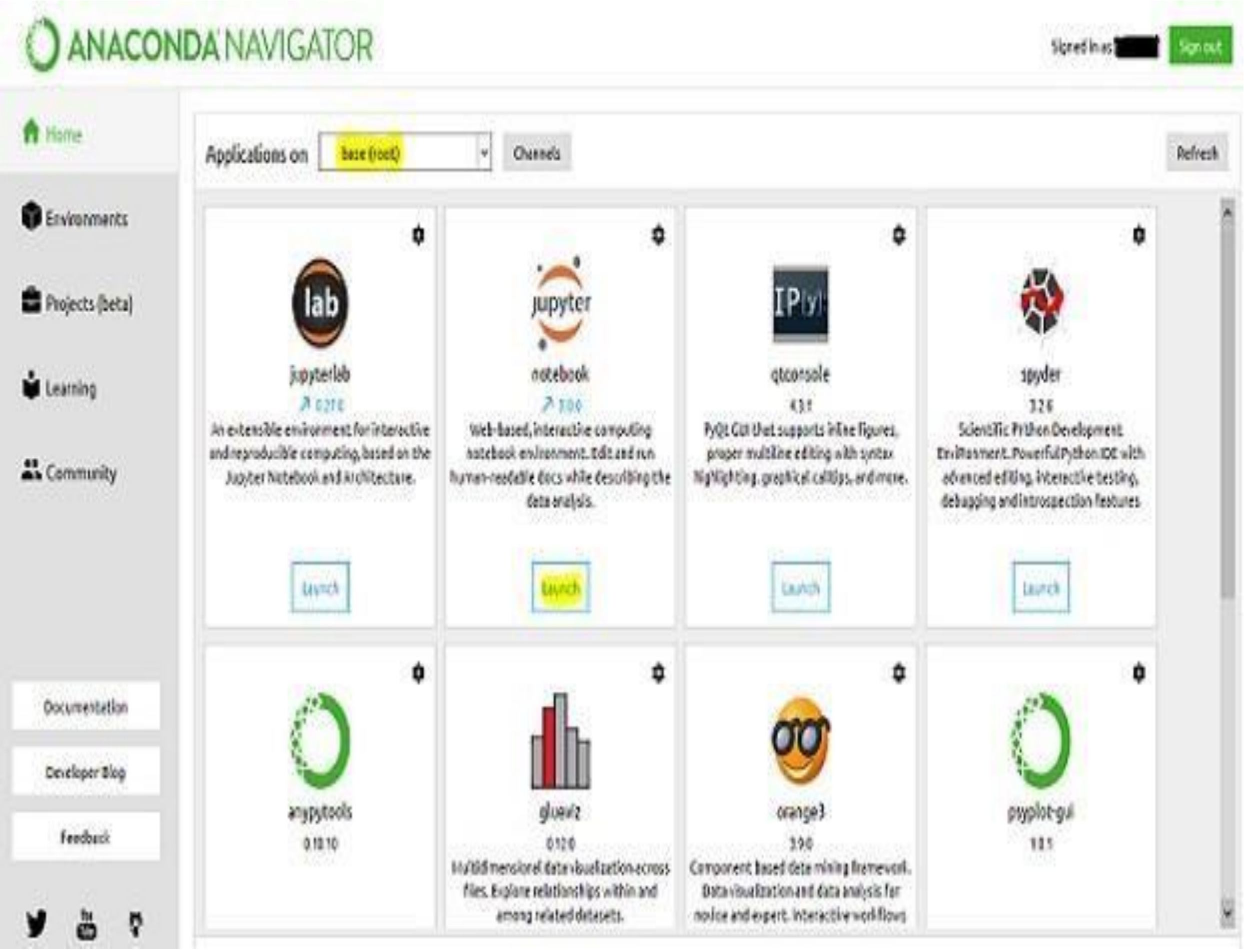
```
python --
version  P>
ython 3.7.3
```

9. Type the command in red to update Anaconda.

```
> conda update --all --yes
```

Start Jupyter Notebook

1. open anaconda navigator and the screen which is similar to below appears.



An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

Scientific Python Development Environment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

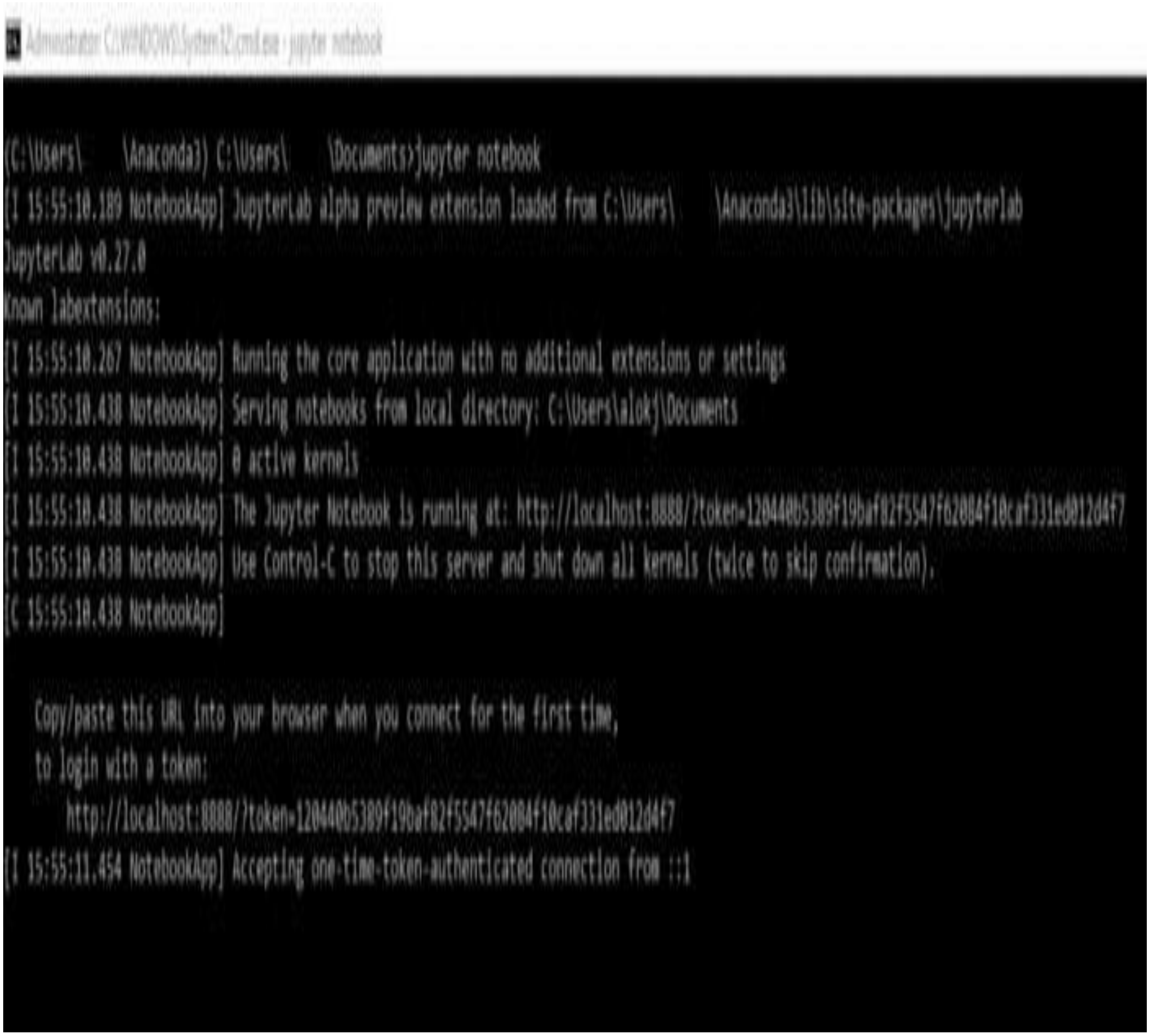
Multidimensional data visualization across files. Explore relationships within and among related datasets.

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows

Fig 5.1.4

Anaconda Navigator

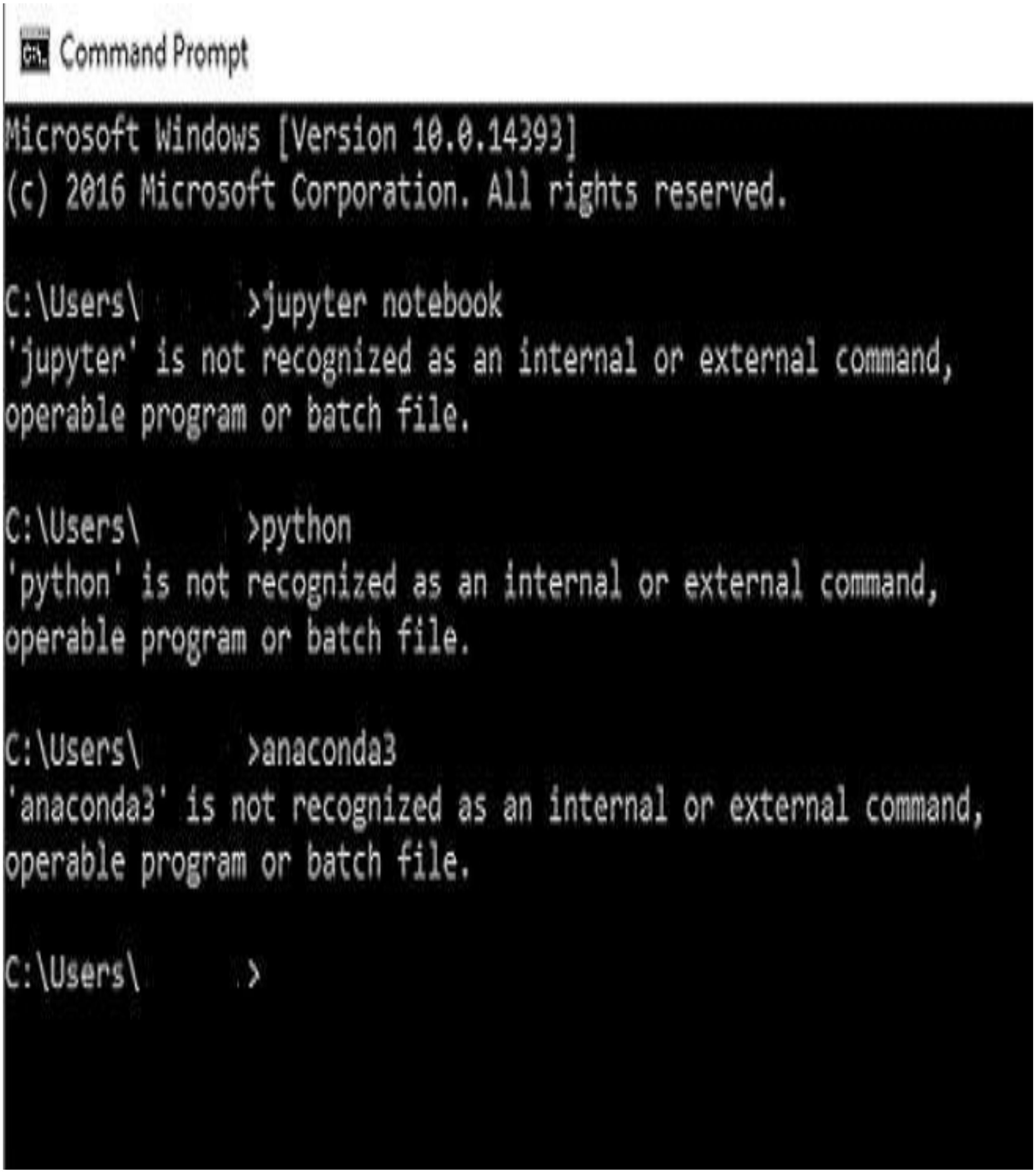
2. open anaconda prompt to open jupyter note book


Administrator: C:\WINDOWS\System32\cmd.exe - jupyter notebook

```
(C:\Users\     \Anaconda3) C:\Users\     \Documents>jupyter notebook
[I 15:55:10.189 NotebookApp] JupyterLab alpha preview extension loaded from C:\Users\     \Anaconda3\lib\site-packages\jupyterlab
JupyterLab v0.27.0
Known labextensions:
[I 15:55:10.267 NotebookApp] Running the core application with no additional extensions or settings
[I 15:55:10.438 NotebookApp] Serving notebooks from local directory: C:\Users\alokj\Documents
[I 15:55:10.438 NotebookApp] 0 active kernels
[I 15:55:10.438 NotebookApp] The Jupyter Notebook is running at: http://localhost:8888/?token=120440b5389f19baf82f5547f62084f10caf331ed012d4f7
[I 15:55:10.438 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 15:55:10.438 NotebookApp]

    Copy/paste this URL into your browser when you connect for the first time,
    to login with a token:
        http://localhost:8888/?token=120440b5389f19baf82f5547f62084f10caf331ed012d4f7
[I 15:55:11.454 NotebookApp] Accepting one-time-token-authenticated connection from ::1
```

Fig 5.1.5

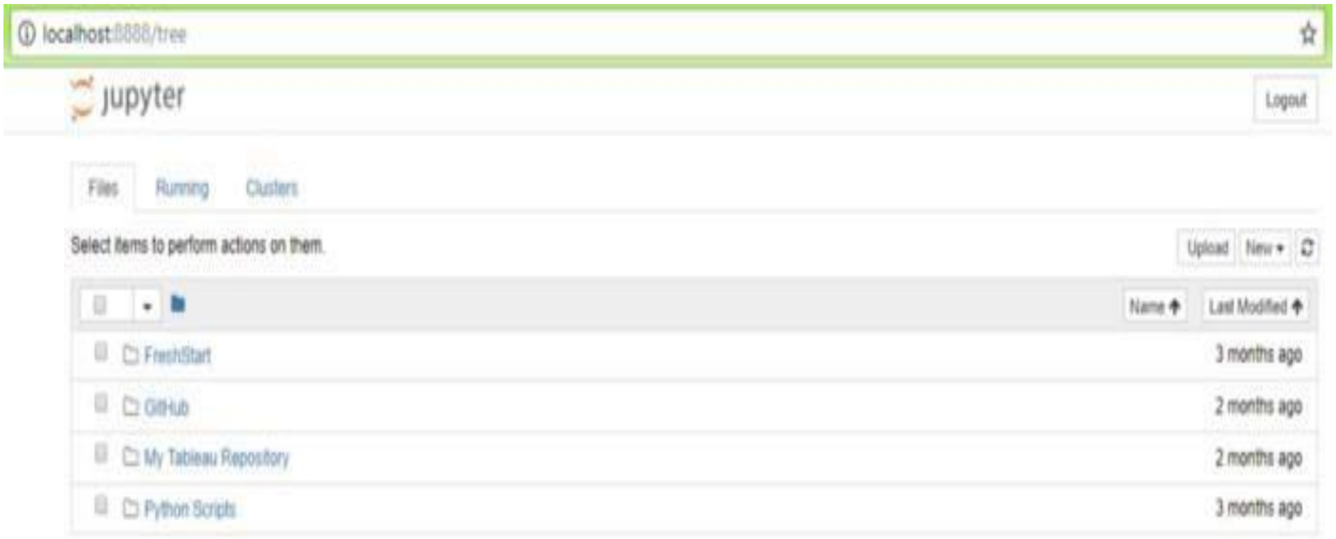
Jupyter Notebook Installation



```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\[redacted]>jupyter notebook
'jupyter' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\[redacted]>python
'python' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\[redacted]>anaconda3
'anaconda3' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\[redacted]>
```

Fig 5.1.6

Checking Jupyter installed or not

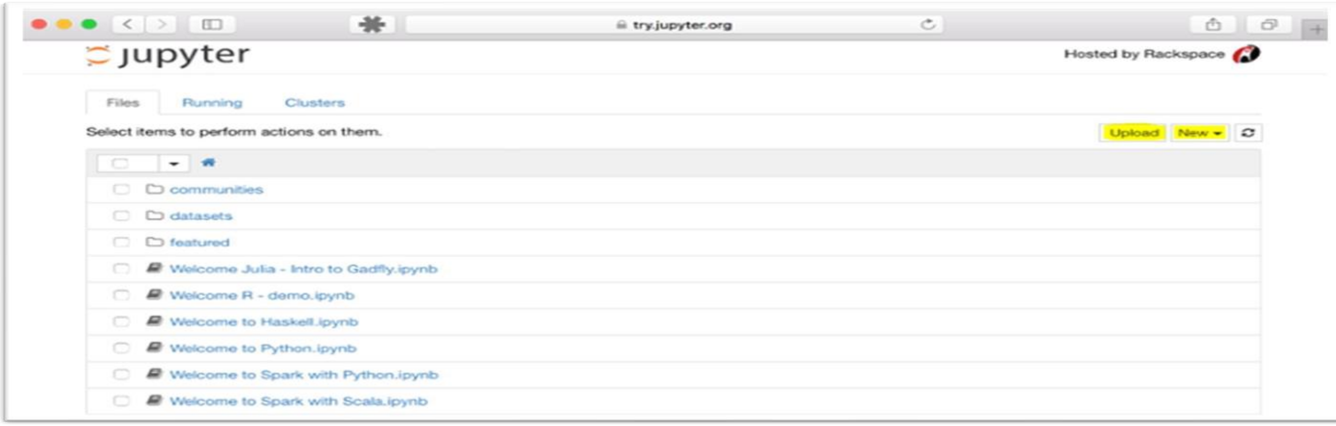


Select items to perform actions on them.

Name	Last Modified
FreshStart	3 months ago
GitHub	2 months ago
My Tableau Repository	2 months ago
Python Scripts	3 months ago

Fig 5.1.7

Jupyter Notebook



Select items to perform actions on them.

- communities
- datasets
- featured
- Welcome Julia - Intro to Gadfly.ipynb
- Welcome R - demo.ipynb
- Welcome to Haskell.ipynb
- Welcome to Python.ipynb
- Welcome to Spark with Python.ipynb
- Welcome to Spark with Scala.ipynb

Fig 5.1.8

Opening new kernel in Jupyter

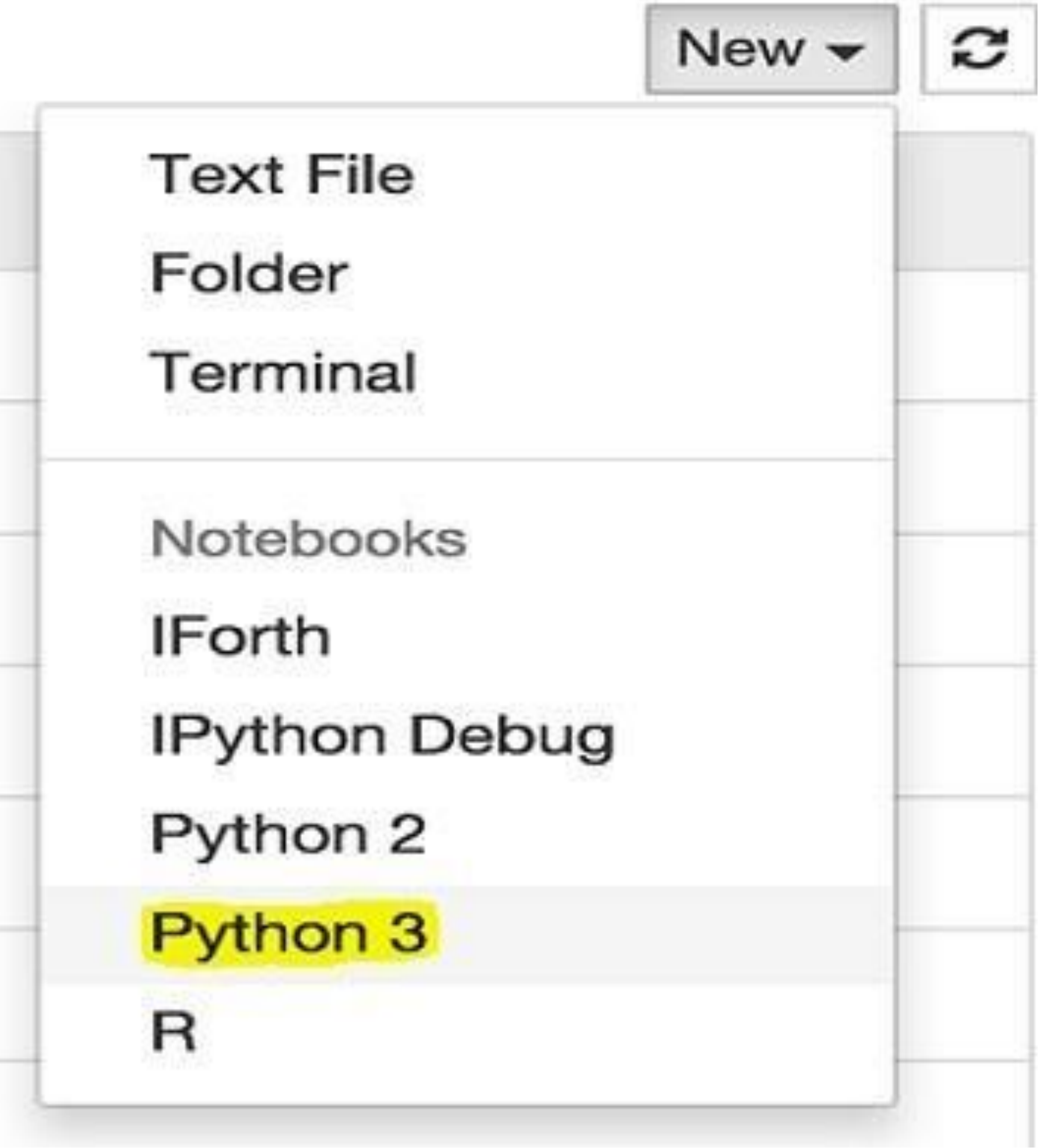


Fig 5.1.9

Selecting Language

5.2 MODULE DESCRIPTION

The modules which we used are

PANDAS - Pandas is a Python library for data analysis and it is most widely used in machine learning tasks.

NUMPY – It stands for ‘Numerical Python’ module, it can be utilised to perform number of mathematical operations on arrays such as trigonometric,statistical and algebric routines.

MATPLOTLIB – It is a python library which is used for data visualization.

SEABORN - It is a python library which is used for making statistical graphs

5.3 SOFTWARE DESCRIPION

Anaconda Cloud is a package management service by Anaconda where users can find, access, store and share public and private notebooks, environments, and conda and PyPI packages.[27] Cloud hosts useful Python packages, notebooks and environments for a wide variety of applications. Users do not need to log in or to have a Cloud account, to search for public packages, download and install them.

Users can build new packages using the Anaconda Client command line interface (CLI), then manually or automatically upload the packages to Cloud.

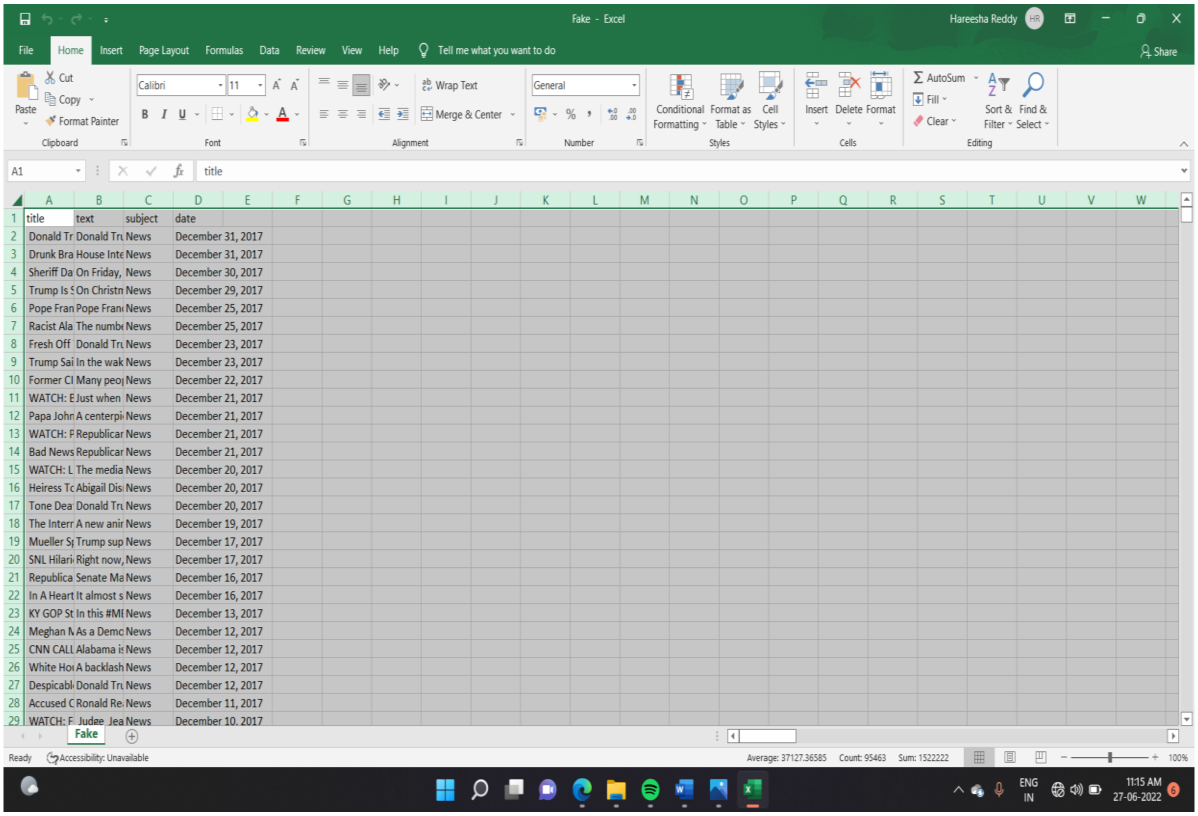
5.4 SAMPLE CODE

Sample input

The dataset contains 4 columns

1. Title
2. Text
3. Subject
4. Date

Fake.CSV

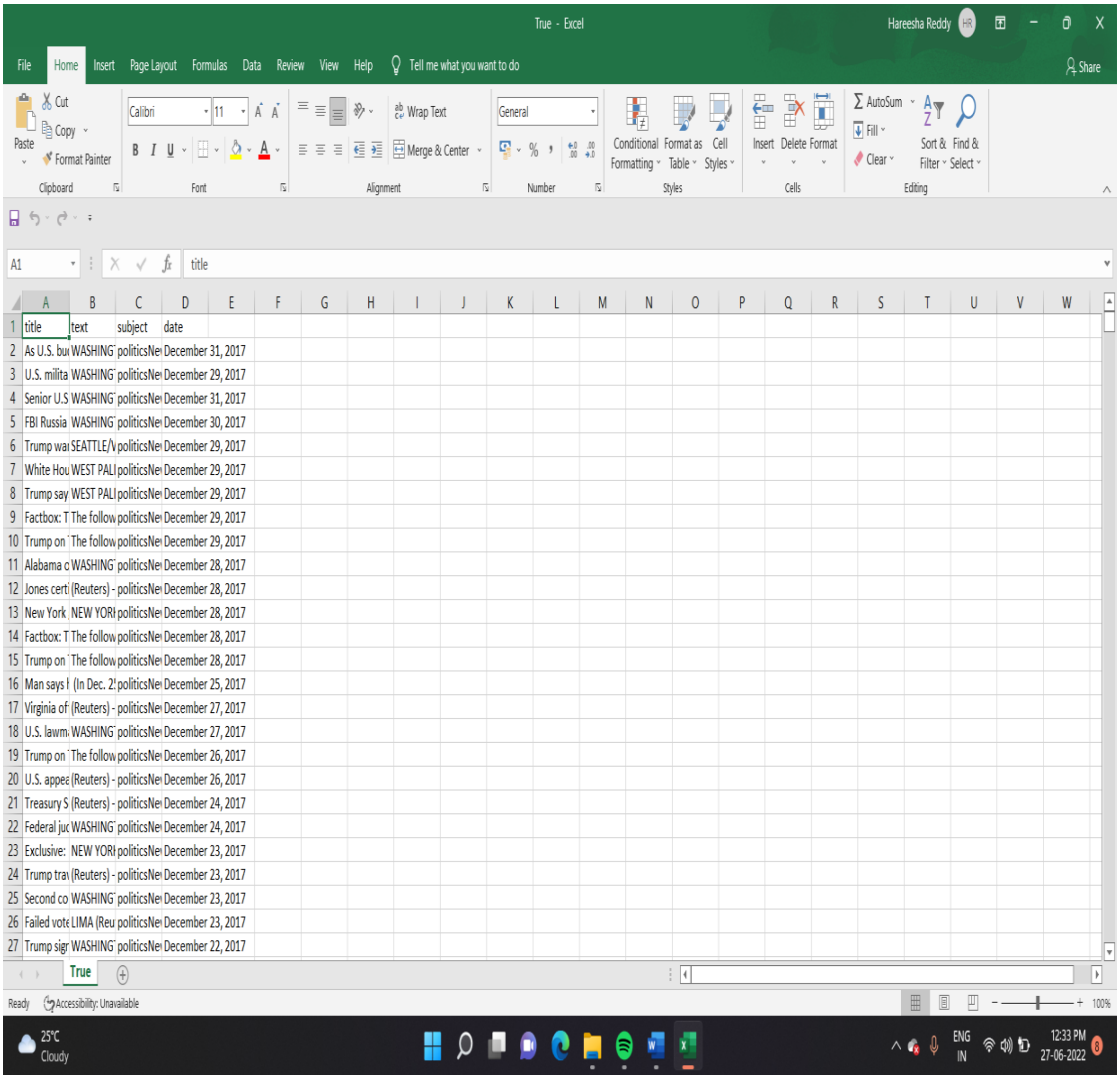


	A	B	C	D	E
1	title	text	subject	date	
2	Donald Tr	Donald Tru	News	December 31, 2017	
3	Drunk Bra	House Inte	News	December 31, 2017	
4	Sheriff Da	On Friday,	News	December 30, 2017	
5	Trump Is S	On Christm	News	December 29, 2017	
6	Pope Fran	Pope Fran	News	December 25, 2017	
7	Racist Ala	The numbe	News	December 25, 2017	
8	Fresh Off	Donald Tru	News	December 23, 2017	
9	Trump Sai	In the wak	News	December 23, 2017	
10	Former Cl	Many peo	News	December 22, 2017	
11	WATCH: E	Just when	News	December 21, 2017	
12	Papa John	A centerpi	News	December 21, 2017	
13	WATCH: P	Republicar	News	December 21, 2017	
14	Bad News	Republicar	News	December 21, 2017	
15	WATCH: L	The media	News	December 20, 2017	
16	Heiress Tc	Abigail Dis	News	December 20, 2017	
17	Tone Dea	Donald Tru	News	December 20, 2017	
18	The Interr	A new anir	News	December 19, 2017	
19	Mueller S	Trump sup	News	December 17, 2017	
20	SNL Hilari	Right now,	News	December 17, 2017	
21	Republica	Senate Ma	News	December 16, 2017	
22	In A Heart	It almost s	News	December 16, 2017	
23	KY GOP St	In this #MI	News	December 13, 2017	
24	Meghan M	As a Demc	News	December 12, 2017	
25	CNN CALL	Alabama is	News	December 12, 2017	
26	White Ho	A backlash	News	December 12, 2017	
27	Despicabl	Donald Tru	News	December 12, 2017	
28	Accused C	Ronald Re	News	December 11, 2017	
29	WATCH: F	Judge Jea	News	December 10, 2017	

Fig 5.4.1

Fake.CSV

True.CSV



	A	B	C	D
1	title	text	subject	date
2	As U.S. bu	WASHING	politicsNe	December 31, 2017
3	U.S. milita	WASHING	politicsNe	December 29, 2017
4	Senior U.S	WASHING	politicsNe	December 31, 2017
5	FBI Russia	WASHING	politicsNe	December 30, 2017
6	Trump wa	SEATTLE/V	politicsNe	December 29, 2017
7	White Hou	WEST PALI	politicsNe	December 29, 2017
8	Trump say	WEST PALI	politicsNe	December 29, 2017
9	Factbox: T	The follow	politicsNe	December 29, 2017
10	Trump on	The follow	politicsNe	December 29, 2017
11	Alabama c	WASHING	politicsNe	December 28, 2017
12	Jones certi	(Reuters) -	politicsNe	December 28, 2017
13	New York	NEW YORI	politicsNe	December 28, 2017
14	Factbox: T	The follow	politicsNe	December 28, 2017
15	Trump on	The follow	politicsNe	December 28, 2017
16	Man says I	(In Dec. 2	politicsNe	December 25, 2017
17	Virginia of	(Reuters) -	politicsNe	December 27, 2017
18	U.S. lawm	WASHING	politicsNe	December 27, 2017
19	Trump on	The follow	politicsNe	December 26, 2017
20	U.S. appea	(Reuters) -	politicsNe	December 26, 2017
21	Treasury S	(Reuters) -	politicsNe	December 24, 2017
22	Federal juc	WASHING	politicsNe	December 24, 2017
23	Exclusive:	NEW YORI	politicsNe	December 23, 2017
24	Trump tra	(Reuters) -	politicsNe	December 23, 2017
25	Second co	WASHING	politicsNe	December 23, 2017
26	Failed vote	LIMA (Reu	politicsNe	December 23, 2017
27	Trump sigr	WASHING	politicsNe	December 22, 2017

True

Fig 5.4.1

True.CSV

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set_style('darkgrid')
import nltk
from sklearn.preprocessing import LabelBinarizer
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from wordcloud import STOPWORDS,WordCloud
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize,sent_tokenize
from bs4 import BeautifulSoup
import re,string,unicodedata
from keras.preprocessing import text,sequence
from nltk.tokenize.toktok import ToktokTokenizer
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
from sklearn.model_selection import train_test_split
from string import punctuation
from nltk import pos_tag
from nltk.corpus import wordnet
import keras
from keras.models import Sequential
from keras.callbacks import ReduceLROnPlateau
import tensorflow as tf
```

```
fake = pd.read_csv(“Fake.csv”)

true = pd.read_csv(“True.csv”)

print(fake.isnull().sum())
print('************')

print(true.isnull().sum())

true=true.fillna(' ')

fake=fake.fillna(' ')

cleansed_data = []

for data in true.text:

if "@realDonaldTrump : - " in data:

cleansed_data.append(data.split("@realDonaldTrump : - ")[1])

elif "(Reuters) -" in data:

cleansed_data.append(data.split("(Reuters) - ")[1])

else:

cleansed_data.append(data)

true["text"] = cleansed_data

true.head(10)

fake['Label'] = 0

true['Label'] = 1

final_data = pd.concat([fake, true])

final_data = final_data.sample(frac=1).reset_index(drop=True)

final_data = final_data.drop(['title', 'text', 'subject', 'date'], axis = 1)

real_words = "

fake_words = "
```

```
for val in final_data[final_data['Label']==1].Sentences:

# split the value

tokens = val.split()

# Converts each token into lowercase

for i in range(len(tokens)):

tokens[i] = tokens[i].lower()

real_words += " ".join(tokens)+" "

for val in final_data[final_data['Label']==0].Sentences:

# split the value

tokens = val.split()

# Converts each token into lowercase

for i in range(len(tokens)):

tokens[i] = tokens[i].lower()

fake_words += " ".join(tokens)+" "

from wordcloud import WordCloud, STOPWORDS

from nltk.corpus import stopwords

stopwords = set(STOPWORDS)

wordcloud = WordCloud(width = 800, height = 800,

background_color ='white',

stopwords = stopwords,

min_font_size = 10).generate(real_words)
```

```
# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
 plt.imshow(wordcloud)
 plt.axis("off")
plt.tight_layout(pad = 0)
 plt.show()
wordcloud = WordCloud(width = 800, height = 800,
 background_color ='white',
stopwords = stopwords,
min_font_size = 10).generate(fake_words)
 # plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
 plt.imshow(wordcloud)
plt.axis("off")
 plt.tight_layout(pad = 0)
 plt.show()
```

To remove urls

```
def remove_URL(s):
regex = re.compile(r'https?://\S+|www\.\S+|bit\.ly\S+')
return regex.sub(r",s)
```

1.To convert text to lower case - x.lower()

2.Remove unneseccary spaces at the end - strip_tags

3.To remove url – Above function

4.To remove punctuation – strip_punctuation

5.To remove multiple white spaces in the sentence between words strip_multiple_whitespaces

6.To remove numbers – strip_numeric

7.To remove stopwords – remove_stopword

```
CUSTOM_FILTERS = [lambda x: x.lower(), strip_tags, remov e_URL, strip_punctuation,
 strip_multiple_whitespaces, strip_n umeric, remove_stopwords, strip_short]

processed_data = []

processed_labels = []

for index, row in final_data.iterrows():

words_broken_up = preprocess_string(row['Sentences'], CU STOM_FILTERS)

if len(words_broken_up) > 0: processed_data.append(words_broken_up)
 processed_labels.append(row['Label'])

print(len(processed_data))

import math

#for training 80 percent of data is used

trainlen=math.ceil((4*len(processed_data))/5)

print(trainlen)

#for testing 20 percent of data is used

testlen=len(processed_data)-trainlen

 print(testlen)
```

```
train=processed_data[:trainlen]
 test=processed_data[trainlen:]
out=final_data.Sentences[trainlen:]
 print(len(test))
print(out[35912])
print(test[0])
 # Word2Vec model trained on processed data
model = Word2Vec(train, min_count=1)
def ReturnVector(x):
try:
return model[x]
except:
return np.zeros(100)
 def Sentence_Vector(sentence):
word_vectors = list(map(lambda x: ReturnVector(x), sentenc e))
return np.average(word_vectors, axis=0).tolist()
X = []
 for data_x in test:
# print(data_x)
X.append(Sentence_Vector(data_x))
 print(test[0])
 X_np = np.array(X)

X_np.shape

 kmeans = cluster.KMeans(n_clusters=2, verbose=0)

 clustered = kmeans.fit_predict(X_np)

testing_df = {'Sentences': test, 'Labels': processed_labels[35912:], 'Pre diction': clustered}

testing_df = pd.DataFrame(data=testing_df)

testing_df.head(10)

trueneg=truepos=falseneg=falsepos=0

for index, row in testing_df.iterrows():

if row['Labels'] == row['Prediction']==0:

trueneg+=1

if row['Labels'] == row['Prediction']==1:

 truepos+=1

if row['Labels'] ==1 and row['Prediction']==0:

falseneg+=1

 if row['Labels'] ==0 and row['Prediction']==1:

falsepos+=1

print("Correctly clustered news: " +
str(((truepos+trueneg)*100)/(trueneg+truepos+falseneg+falsepos)) + "%")

print(trueneg,falsepos,sep=" ")

 print(falseneg,truepos,sep="
```

6. SYSTEM TESTING

Testing is vital to the success of the system. System testing makes a logical assumption that if all parts of the system correct the results will be successfully achieved. Effective testing early in the process translates directly into long term cost saving, reduced number of errors.

System testing is done when all the modules of the system are in working order and has been tested independently for proper working. All the pieces are put into one system and test to determined, whether it needs user's requirements. The best program is worthless if doesn't needs.

System testing is designed to uncover weakness that were not found in earlier tests like program testing in which only syntactical and logical are removed. The purpose of System Testing is to consider all the likely variations to which it will be subjected and then push the system to its limits.

Whenever a software is build, there is always scope for improvement and those improvements brings changes in picture. Changes may be required to modify or update any existing solution or to create a new solution for a problem. Requirements keeps on changing on daily basis and so we need to keep on upgrading our systems based on the current requirements and needs to meet desired outputs. Changes should be analyzed before they are made to the existing system, recorded before they are implemented, reported to have details of before and after, and controlled in a manner that will improve quality and reduce error. This is where the need of System Configuration Management comes.

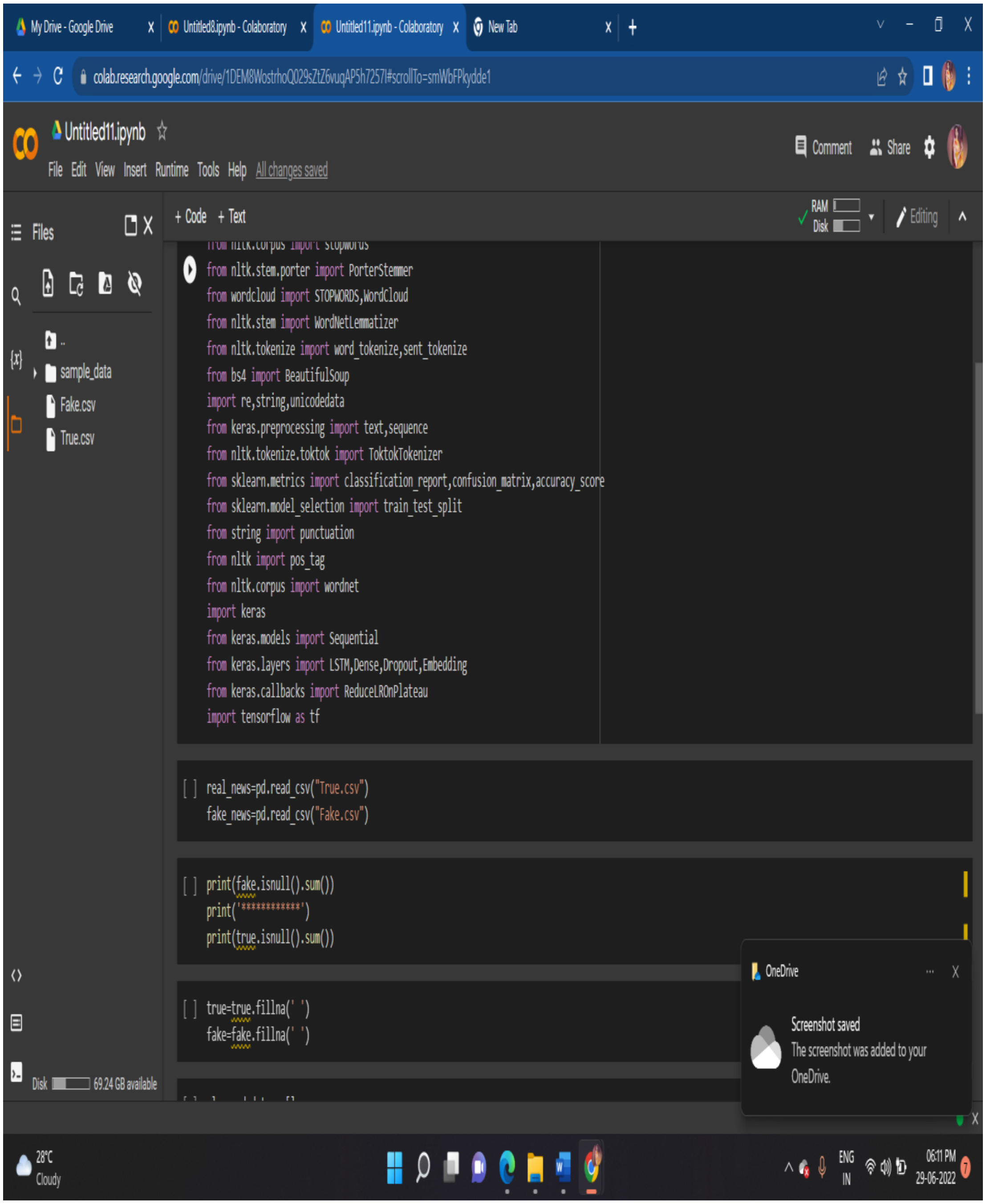
UNIT TESTING

Unit testing involves the planning of test cases that validate that the interior program logic is functioning properly, which program inputs produce valid outputs. All decision branches and internal code flow should be validated. it's the testing of individual software units of the appliance.

It is done after the completion of a private unit before integration. this is often a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a selected business process, application, and/or system configuration. Unit tests make sure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

7. RESULTS SCREENSHOTS

7.1 loading the data



My Drive - Google Drive
Untitled8.ipynb - Colaboratory
Untitled11.ipynb - Colaboratory
New Tab
colab.research.google.com/drive/1DEM8WostrhoQ029sZtZ6vuqAP5h7257I#scrollTo=smWbFPkydde1
Untitled11.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
Files
sample_data
Fake.csv
True.csv
Disk 69.24 GB available
28°C
Cloudy
ENG
IN
06:11 PM
29-06-2022

```
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from wordcloud import STOPWORDS,WordCloud
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import word_tokenize,sent_tokenize
from bs4 import BeautifulSoup
import re,string,unicodedata
from keras.preprocessing import text,sequence
from nltk.tokenize.toktok import ToktokTokenizer
from sklearn.metrics import classification_report,confusion_matrix,accuracy_score
from sklearn.model_selection import train_test_split
from string import punctuation
from nltk import pos_tag
from nltk.corpus import wordnet
import keras
from keras.models import Sequential
from keras.layers import LSTM,Dense,Dropout,Embedding
from keras.callbacks import ReduceLROnPlateau
import tensorflow as tf
```

```
real_news=pd.read_csv("True.csv")
fake_news=pd.read_csv("Fake.csv")
```

```
print(fake.isnull().sum())
print('************')
print(true.isnull().sum())
```

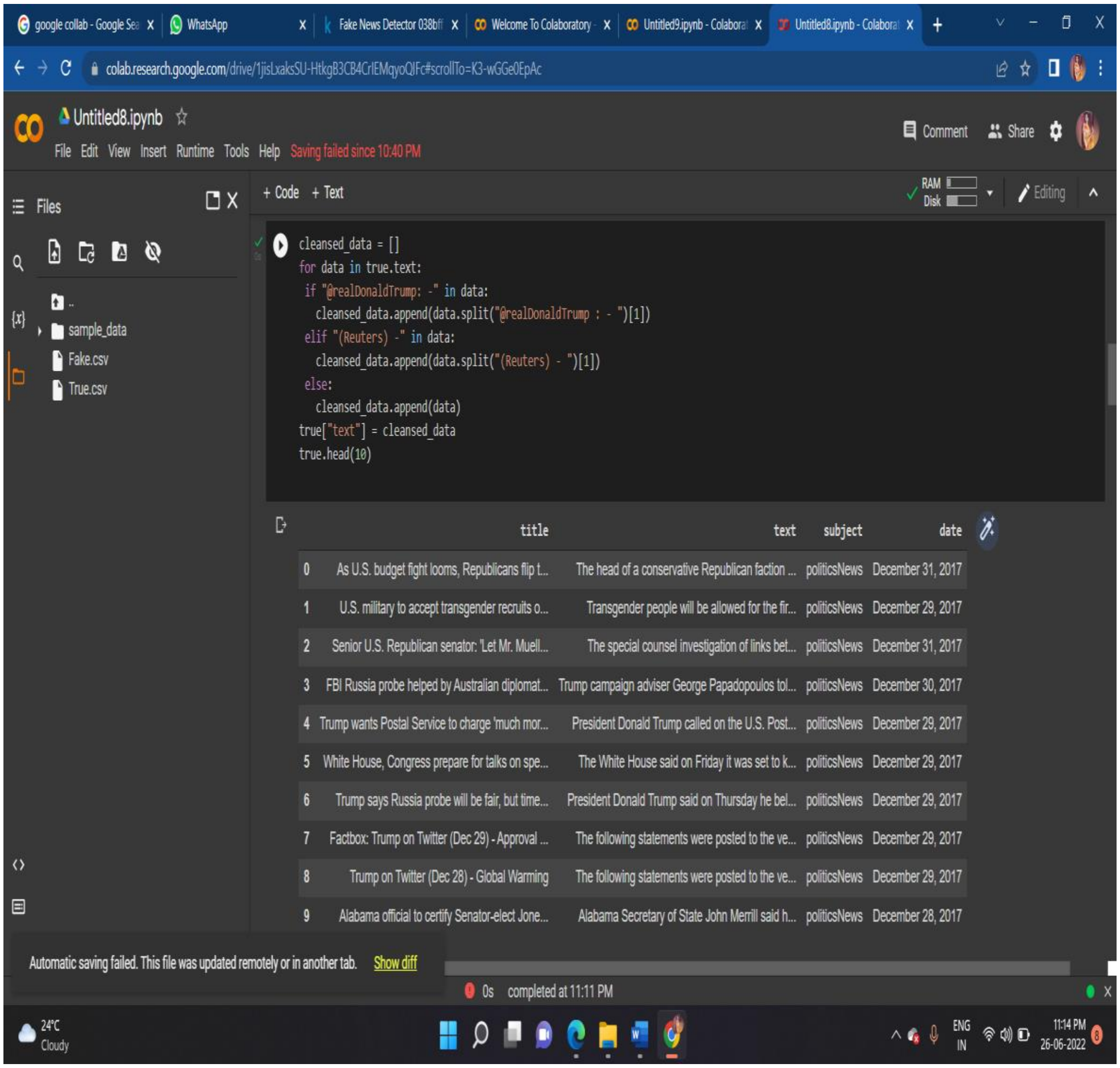
```
true=true.fillna(' ')
fake=fake.fillna(' ')
```

OneDrive

Screenshot saved

The screenshot was added to your OneDrive.

7.2 Remove unnecessary data



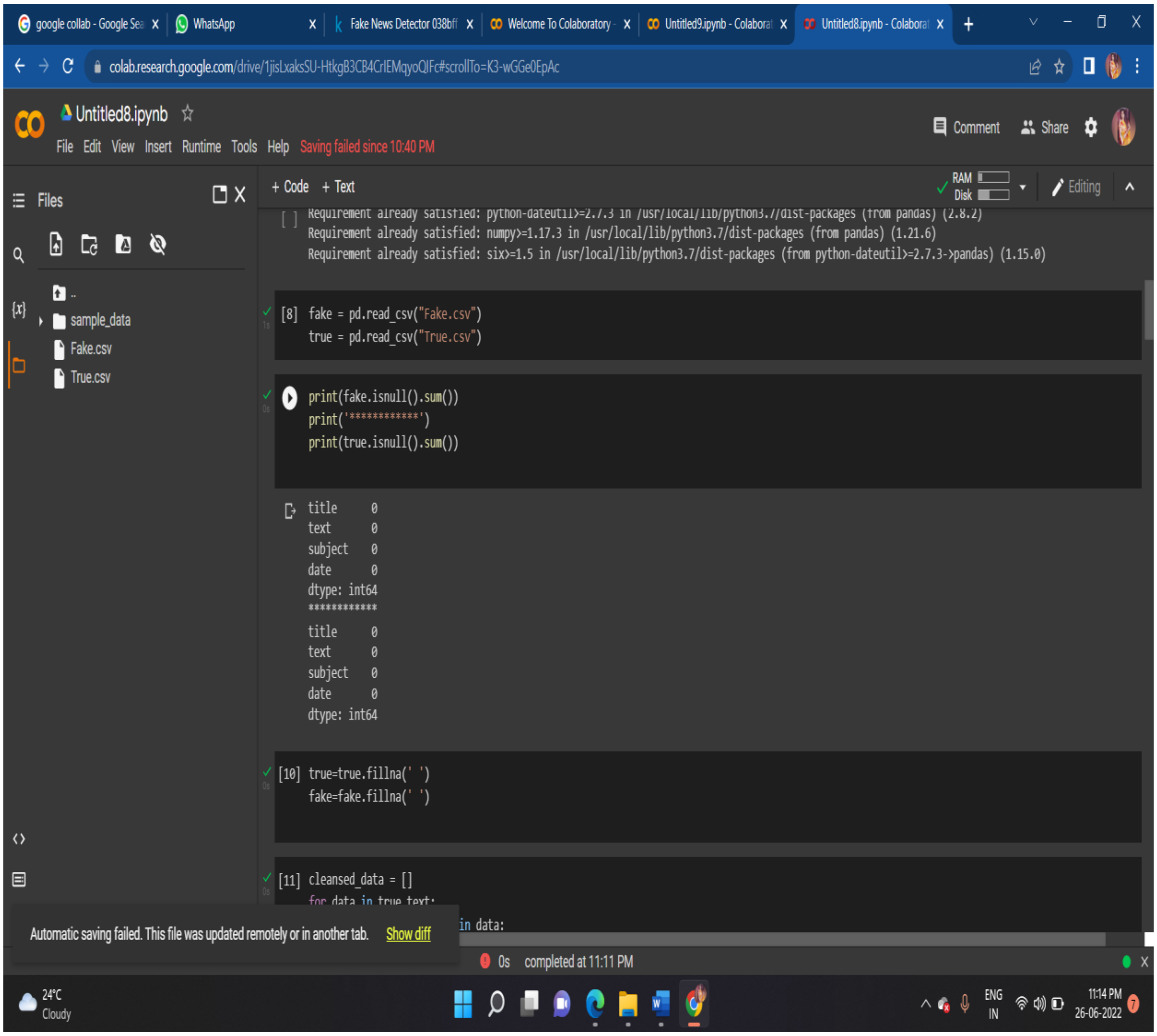
```
cleansed_data = []
for data in true.text:
 if "@realDonaldTrump: -" in data:
   cleansed_data.append(data.split("@realDonaldTrump : - ")[1])
 elif "(Reuters) -" in data:
   cleansed_data.append(data.split("(Reuters) - ")[1])
 else:
   cleansed_data.append(data)
true["text"] = cleansed_data
true.head(10)
```

	title	text	subject	date
0	As U.S. budget fight looms, Republicans flip t...	The head of a conservative Republican faction ...	politicsNews	December 31, 2017
1	U.S. military to accept transgender recruits o...	Transgender people will be allowed for the fir...	politicsNews	December 29, 2017
2	Senior U.S. Republican senator: 'Let Mr. Muell...	The special counsel investigation of links bet...	politicsNews	December 31, 2017
3	FBI Russia probe helped by Australian diplomat...	Trump campaign adviser George Papadopoulos tol...	politicsNews	December 30, 2017
4	Trump wants Postal Service to charge 'much mor...	President Donald Trump called on the U.S. Post...	politicsNews	December 29, 2017
5	White House, Congress prepare for talks on spe...	The White House said on Friday it was set to k...	politicsNews	December 29, 2017
6	Trump says Russia probe will be fair, but time...	President Donald Trump said on Thursday he bel...	politicsNews	December 29, 2017
7	Factbox: Trump on Twitter (Dec 29) - Approval ...	The following statements were posted to the ve...	politicsNews	December 29, 2017
8	Trump on Twitter (Dec 28) - Global Warming	The following statements were posted to the ve...	politicsNews	December 29, 2017
9	Alabama official to certify Senator-elect Jone...	Alabama Secretary of State John Merrill said h...	politicsNews	December 28, 2017

Automatic saving failed. This file was updated remotely or in another tab. Show diff

0s completed at 11:11 PM

7.3 find the null values


google collab - Google Sea
WhatsApp
Fake News Detector 038bf
Welcome To Colaboratory
Untitled9.ipynb - Colabora
Untitled8.ipynb - Colabora
colab.research.google.com/drive/1jisLxaksSU-HtkgB3CB4CrIEMqyoQIFc#scrollTo=K3-wGGe0EpAc
Untitled8.ipynb
File Edit View Insert Runtime Tools Help
Saving failed since 10:40 PM
Comment
Share
Files
sample_data
Fake.csv
True.csv
+ Code + Text
RAM
Disk
Editing

```
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-packages (from pandas) (2.8.2)
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.7/dist-packages (from pandas) (1.21.6)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.7.3->pandas) (1.15.0)
```

```
[8] fake = pd.read_csv("Fake.csv")
    true = pd.read_csv("True.csv")
```

```
print(fake.isnull().sum())
print('************')
print(true.isnull().sum())
```

```
title      0
text       0
subject    0
date       0
dtype: int64
************
title      0
text       0
subject    0
date       0
dtype: int64
```

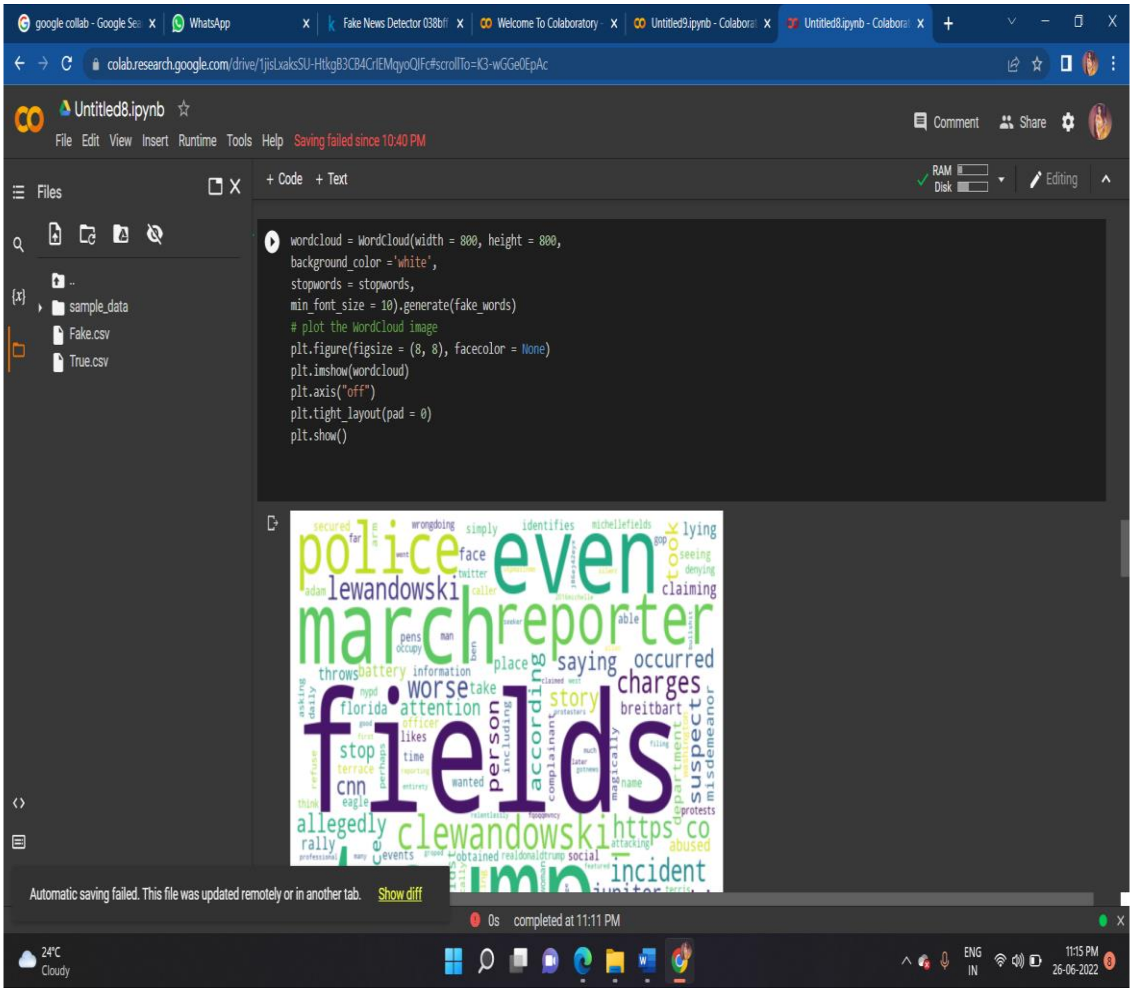
```
[10] true=true.fillna(' ')
     fake=fake.fillna(' ')
```

```
[11] cleansed_data = []
     for data in true.text:
                       in data:
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

0s completed at 11:11 PM

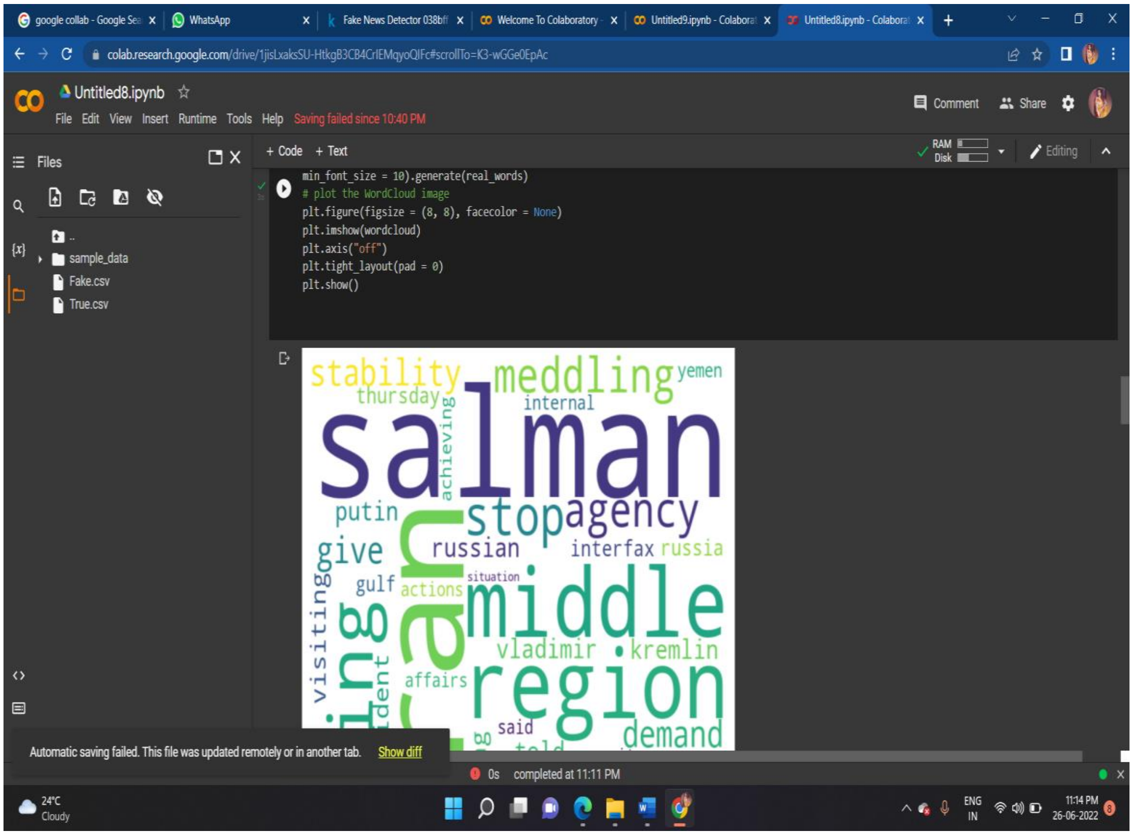
7.4 Visualize Real Words



```
wordcloud = WordCloud(width = 800, height = 800,
background_color ='white',
stopwords = stopwords,
min_font_size = 10).generate(fake_words)
# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
plt.imshow(wordcloud)
plt.axis("off")
plt.tight_layout(pad = 0)
plt.show()
```

Automatic saving failed. This file was updated remotely or in another tab. Show diff

7.5 Visualize Fake Words

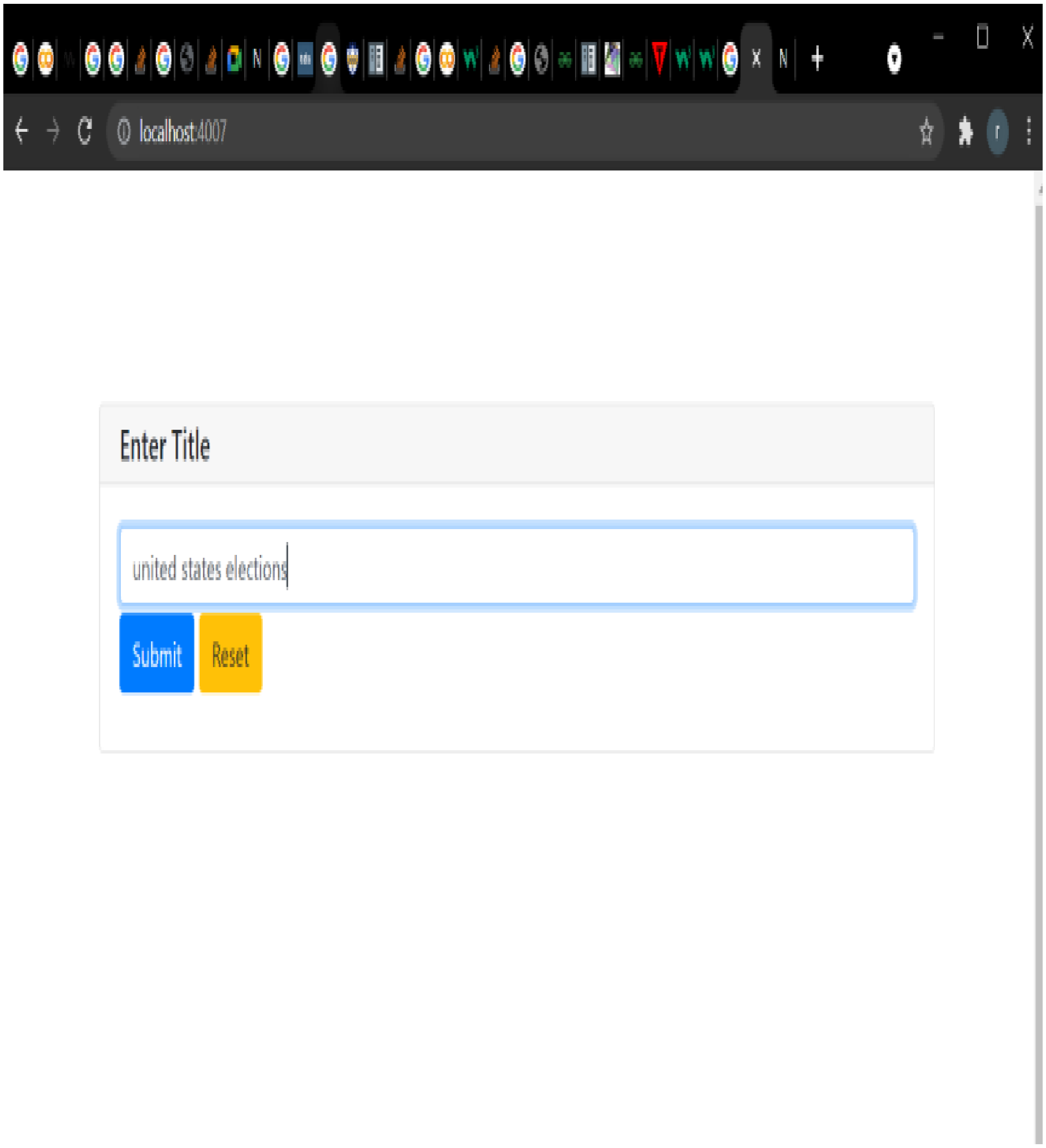


Saving failed since 10:40 PM

```
min_font_size = 10).generate(real_words)
# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
plt.imshow(wordcloud)
plt.axis("off")
plt.tight_layout(pad = 0)
plt.show()
```

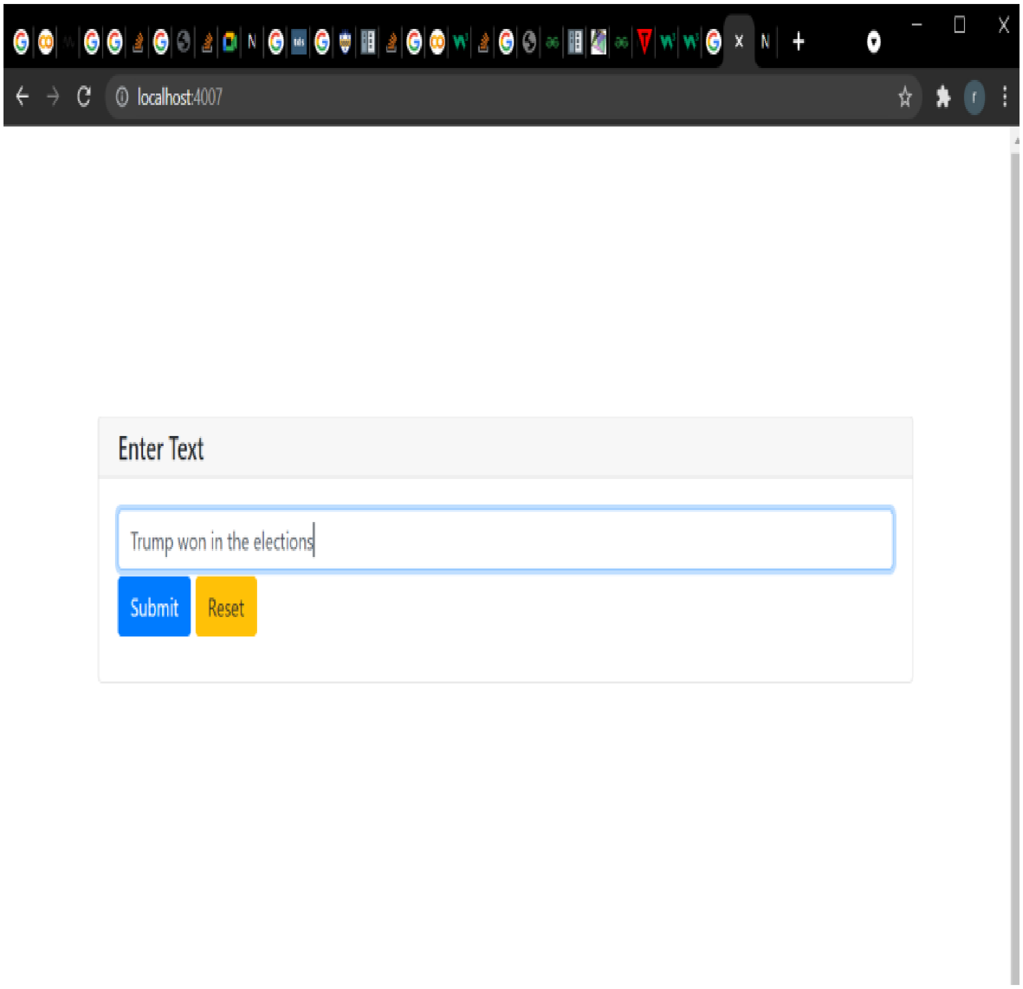
Automatic saving failed. This file was updated remotely or in another tab. Show diff

0s completed at 11:11 PM



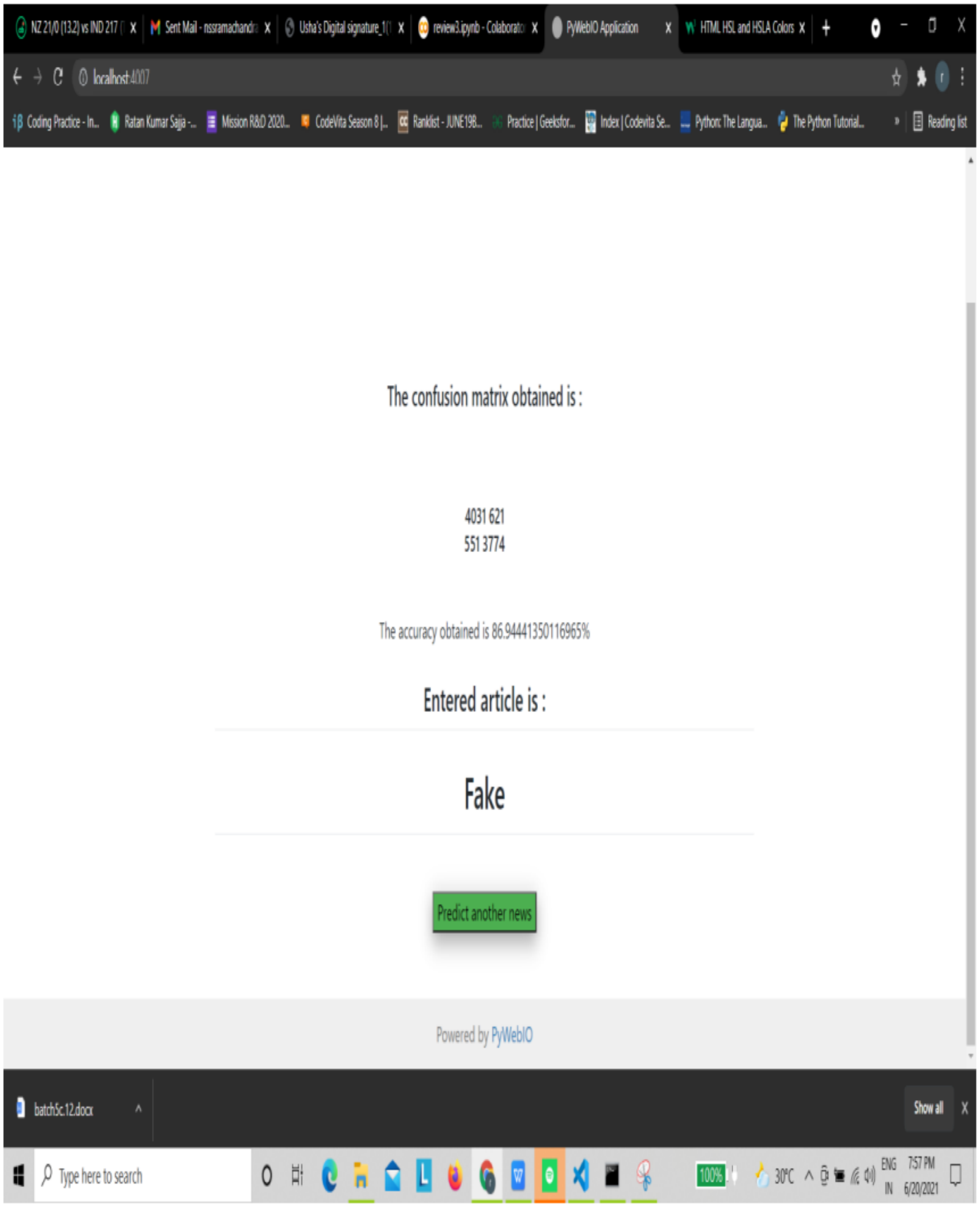
Enter Title

united states elections



Enter Text

Trump won in the elections



The confusion matrix obtained is :

```
4031 621
551 3774
```

The accuracy obtained is 86.94441350116965%

Entered article is :

Fake

Predict another news

Powered by PyWebIO

8. CONCLUSION

In this project, we are predicting whether an article is a real or fake article based on the relationship between the words. We have used the 2016 US president election datasets for creation of this system . We used Word2Vec model for building model and K -Means for the prediction and obtained an accuracy of 87%.

8.1 Future Work:

1. We want to use web scraping and get the data from various social media and websites by ourself and use them in our system.

2. We also want to improve the accuracy by query optimssation

9. BIBLIOGRAPHY

1. Shloka Gilda,“Evaluating Machine Learning Algorithms for Fake News Detection” ,2017 IEEE 15th Student Conference on Research and Development (SCOReD).
2. Mykhailo Granik, Volodymyr Mesyura, “Fake News Detection Using Naive Bayes Classifier”, 2017 IEEEFirst Ukraine Conference on Electrical and Computer Engineering (UKRCON).
3. Zhang, C., et al., Detecting fake news for reducing misinformation risks using analytics approaches. European Journal of Operational Research, 2019.
4. Robbins, K.R., W. Zhang, and J.K. Bertrand, The ant colony algorithm for feature selection in high-dimension gene expression data for disease classification. Journal of Mathematical Medicine and Biology, 2008.
5. Alirezaei, M., S.T.A. Niaki, and S.A.A. Niaki, A bi-objective hybrid optimization algorithm to reduce noise and data dimension in diabetes diagnosis using support vector machines. Expert Systems with Applications, 2019. 127: p. 47-57.
6. Zakeri, A. and A. Hokmabadi, Efficient feature selection method using real-valued grasshopper optimization algorithm. Expert Systems with Applications, 2019. 119: p. 61-72.
7. Yimin Chen, Niall J Conroy, and Victoria L Rubin. 2015. News in an online world: The need for an “automatic crap detector”. Proceedings of the Association for Information Science and Technology, 52(1):1–4.
8. Victoria L Rubin, Niall J Conroy, Yimin Chen, and Sarah Cornwell. 2016. Fake news or truth? Using satirical cues to detect potentially misleading news. In Proceedings of NAACL-HLT, pages 7–17.
9. Balmas, M., 2014. When fake news becomes real: Combined exposure to multiple news sources and political attitudes of inefficacy, alienation, and cynicism. Communication Research 41, 430–454.
10. Pogue, D., 2017. How to stamp out fake news. Scientific American 316, 24–2

11. Aldwairi, M. and A. Alwahedi, Detecting Fake News in Social Media Networks. Procedia Computer Science, 2018. 141: p. 215-222.

12. Mehdi H.A, Nasser G.A, Mohammad B, Text feature selection using ant colony optimization, Expert Systems with Applications, 2009

13. Jain, A.K., 2010. Data clustering: 50 years beyond K-means. Pattern recognition letters, 31(8), pp.651-666.

14. Reis, J.C., Correia, A., Murai, F., Veloso, A., Benevenuto, F. and Cambria, E., 2019. Supervised Learning for Fake News Detection.

15. Cortes, Corinna; Vapnik, Vladimir N. (1995). "Support-vector networks" (PDF). Machine Learning. 20 (3): 273–297. CiteSeerX