FAKE NEWS DETECTION USING PASSIVE AGGRESSIVE CLASSIFIER

PROJECT REPORT

(BTCS 703-18)

Submitted in partial fulfillment of the

requirements for the award of the degree

of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

Sharpening Minds, Brightening Tomorrow.
Gulzar Group of Institutes

UNDER THE GUIDANCE OF **Dr./Er. RUPINDERJIT KAUR**

Submitted by-Rohit Rattan (1906135)

Sachin Kumar (1906136)

Sumaira nabi (1906153)

KAPURTHALA

ਆਈ. ਕੇ. ਗੁਜਰਾਲ ਪੰਜਾਬ ਟੈਕਨੀਕਲ ਯੂਨੀਵਰਸਿਟੀ, ਜਲੰਧਰ

I.K. GUJRAL PUNJAB TECHNICAL UNIVERSITY, JALANDHAR

CANDIDATE'S DECLARATION AND CERTIFICATE

We hereby certify that the work, which is being presented in this report entitled, **Fake News Detection using Passive Aggressive Classifier**, in partial fulfillment of the requirements for the degree of **Bachelor of Technology**, submitted in the **Department of Computer Science and Engineering**, Gulzar Group of Institutes, Khanna, Punjab; by **Rohit Rattan (1906135), Sachin Kumar (1906136) & Sumaira Nabi (1906153)** is the authentic record of our own work carried out under the supervision of **Dr/Er. Rupinderjit Kaur, Department of Computer Science and Engineering**, Gulzar Group of Institutes, Khanna, Punjab.

We further declare that the matter embodied in this report has not been submitted by us for the award of any other degree.

Candidate(s) Signature

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Signature of HOD **Signature of Supervisor**

Er. Jai Prakash **Dr./Er. RUPINDERJIT KAUR**

Date:

ACKNOWLEDGMENT

It is our pleasure to acknowledge the contributions of all who have helped us and supported us during this Project report.

First, we thank God for helping us in one way or another and providing strength and endurance to us. We wish to express my sincere gratitude and indebtedness to our supervisor, Supervisor Name, department name, Gulzar Group of Institutes, Khanna, Punjab; for her/his intuitive and meticulous guidance and perpetual inspiration in completion of this report. In spite of his/her busy schedule, he/she rendered help whenever needed, giving useful suggestions and holding informal discussions. Her invaluable guidance and support throughout this work cannot be written down in few words. We also thank her for providing facilities for my work in the department name.

We are also humbly obliged by the support of our group members and friends for their love and caring attitude. The sentimental support they rendered to us is invaluable and everlasting. They have helped us through thick and thin and enabled us to complete the work with joy and vigor. We thank the group members for entrusting in each other and following directions, without them this report would never have been possible.

We are also thankful to our parents, elders and all family members for their blessing, motivation and inspiration throughout our work and bearing with us even during stress and bad temper. They have always provided us a high moral support and contributed in all possible ways in completion of this Capstone report.

ABSTRACT

Recent public incident have lead to an increase in the acceptance and spread of fake news. As illustrate by the widespread result of the large arrival of fake news, peoples are out of step with if not outright poor detectors of fake news. With this, struggle will have been made to automate the way of fake news detection. The most popular of such experiment include “blacklists” of sources and authors that are irresponsible. While these tools are useful, in sequence to make a more complete end to end solution, and need to version for more heavy cases where dependable sources and authors release fake news. As such, the objective of this project was to design a LIWC tool for detecting the language patterns that specify fake and real news through the use of passive-aggressive techniques. The results of this project denote the capacity for passive-aggressive to be useful in this task. To built a model that capture many automatic warning of real and fake news as well as an application that aids in the visualization of the classification decision.

TABLE OF CONTENTS:

Chapter No. Title Page No.

LIST OF FIGURES

LIST OF TABLES

CHAPTER – 1 INTRODUCTION:

The fake news has been rapidly increasing in numbers. It is not a new problem but recently ithas been on a great rise. According to Wikipedia Fake news is false or misleading information presented as news. Detecting the fake news has been a challenging and a complex task. It is observed that humans have a tendency to believe the misleading information which makes the spreading of fake news even easier. According to reports it is found that human ability to detect deception without special assistance is only 54%.

Fake news is dangerous as it can deceive people easily and create a state of confusion among a community. This can further affect the society badly .The spread of fake news creates rumors circulating around and the victims could be badly impacted. Recent reports showed that due to the rise of fake news that was being created online it had impacted the US Presidential Elections.Fake news might be created by people or groups who are acting in their own interests or those of third parties.The creation of misinformation is usually motivated by personal, political, or economic agendas.

Since a lot of time is spent by users on social media and people prefer online means of information it has become difficult to know about the authenticity of the news. People acquire most of the information by these means as it is free and can be accessed from anywhere irrespective of place and time. Since this data can be put out by anyone there is lack of accountability in it which makes it less trustable unlike the traditional methods of gaining information like newspaper or some trusted source. In this paper, we deal with such fake news detection issue. We have used the techniques of NLP and ML to build the model .We have alsocompared text vectorization methods and obtained the one which gives a better output.

"Fake news" has been used in a multitude of ways in the last half a year and multiple definitions have been given. For instance, the New York times defines it as "a made-up story with an intention to deceive". This definition focuses on two dimensions: the intentionality (very difficult to prove) and the fact that the story is made up. This implies that honest mistakes (no matter how major they are, as long as they are accidental) are not considered to be fake news. First, an organization dedicated to improving skills and standards in the reporting and sharing of online information, has recently published a great article that explains the fake news environment and proposes 7 types of fake content:

1.False Connection: Headlines, visuals or captions don’t support the content.

2.False Context: Genuine content is shared with false contextual information.

3.Manipulated content: Genuine information or imagery is manipulated.

4.Satire or Parody: No intention to cause harm but potential to fool.

5.Misleading Content: Misleading use of information to frame an issue/individual.

6.Imposter Content: Impersonation of genuine sources.

7.Fabricated content: New content that is 100% false.

(1) Impact on society- Spams usually exist in personal emails or specific review websites and merely have an area impact on a little number of audiences, while the impact fake news in online social networks are often tremendous thanks to the huge user numbers globally, which is further boosted by the extensive information sharing and propagation among these users

(2) Audiences initiative- Rather than receiving spam emails passively, users in online social networks may search for, receive and share news information actively with no sense about its correctness;

(3) Identification difficulty- Identification difficulty: via differentiation with infinite structured messages (in emails or review websites), spams are usually easier to be famous, whereas identifying fake news with inaccurate information is enormously testing, since it requires both tedious evidence-collecting and careful fact-checking thanks to the shortage of other Comparative news articles available.

Machine Learning:

Machine Learning is a machine of pc algorithms which can examine from instance via self-development without being explicitly coded via way of means of a programmer. Machine getting to know is part of synthetic Intelligence which mixes records with statistical equipment to expect an output which may be used to make actionable insights. The step forward comes with the concept that a system can singularly examine from the records (i.e., instance) to provide correct results. Machine getting to know is carefully associated with records mining and Bayesian predictive modeling. The system gets records as enter and makes use of a set of rules to formulate answers. An ordinary system getting to know responsibilities are to offer a recommendation. For the ones who've a Netflix account, all tips of films or collection are primarily based totally at the person's ancient records. Tech corporations are the usage of unsupervised getting to know to enhance the person enjoy with personalizing recommendation. Machine getting to know is likewise used for a lot of responsibilities like fraud detection, predictive maintenance, portfolio optimization, automatize venture and so on. Machine Learning vs. Traditional Programming Traditional programming differs considerably from system getting to know. In conventional programming, a programmer codes all of the regulations in session with an professional with inside the enterprise for which software program is being developed. Each rule is primarily based totally on a logical foundation; the system will execute an output following the logical statement. When the machine grows complex, extra regulations want to be written. It can speedy emerge as unsustainable to maintain. Traditional programming differs considerably from system getting to know. In conventional programming, a programmer codes all of the regulations in session with a professional with inside the enterprise for which software program is being developed. Each rule is primarily based totally on a logical foundation; the system will execute an output following the logical statement. When the machine grows complex, extra regulations want
to be written. It can speedy emerge as unsustainable to maintain.

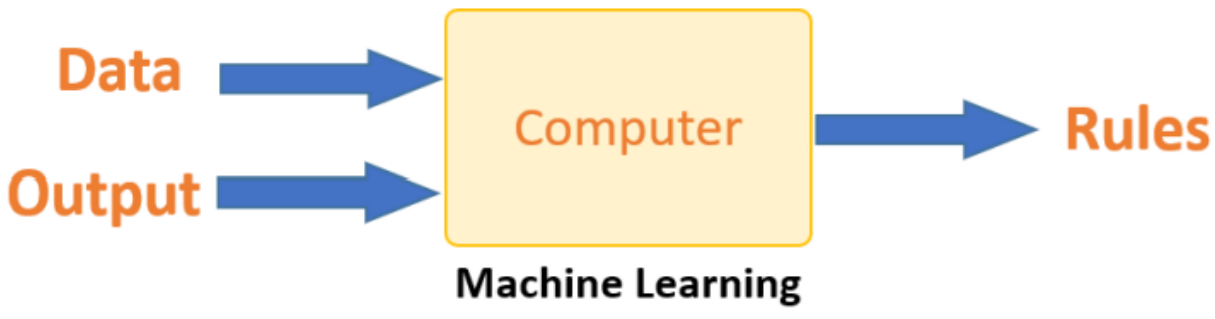

Fig 1.1 Machince Learning

Work of Machine:

LearningMachine getting to know is the mind where in all of the getting to know takes place. The manner the system learns is much like the human being. Humans study from experience. The extra we know, the extra without difficulty is predict in this work. By analogy, while we are facing an unknown situation, the chance of achievement is decrease than the acknowledged situation. Machines are educated the same. To give a correct prediction, the system sees an example. When we deliver the system a comparable example, it is able to parent out the outcome. However, like a human, if it feed a formerly unseen example, the system has problems to predict. The centre goal of system getting to know is the getting to know and inference. First of all, the system learns via the invention of patterns. This discovery is made way to the facts. One critical a part of the facts scientist is to select cautiously which facts to offer to the system. The listing of attributes used to resolve a trouble is known as a characteristic vector. You can think about a characteristic vector as a subset of facts this is used to address a trouble. The system makes use of a few fancy algorithms to simplify the truth and remodel this discovery right into a model. Therefore, the getting to know degree is used to explain the facts and summarize it right into a model.

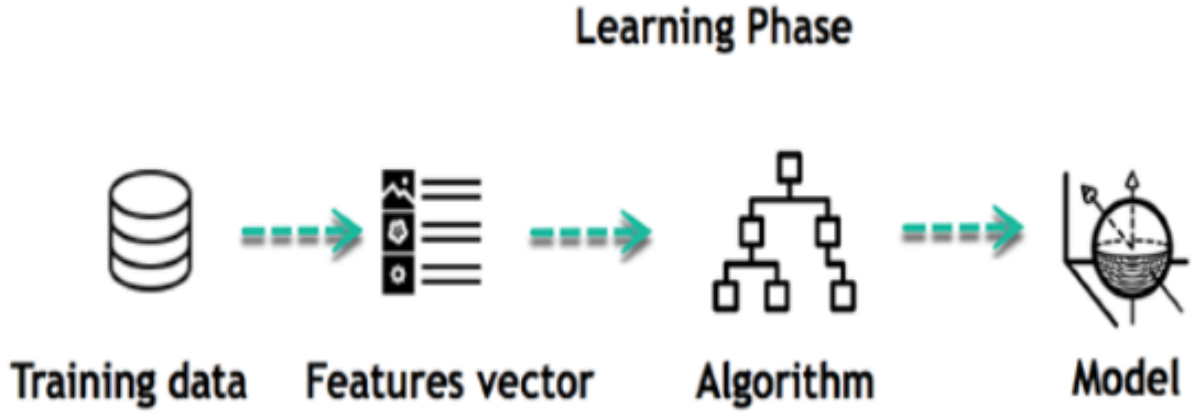

Fig 1.2

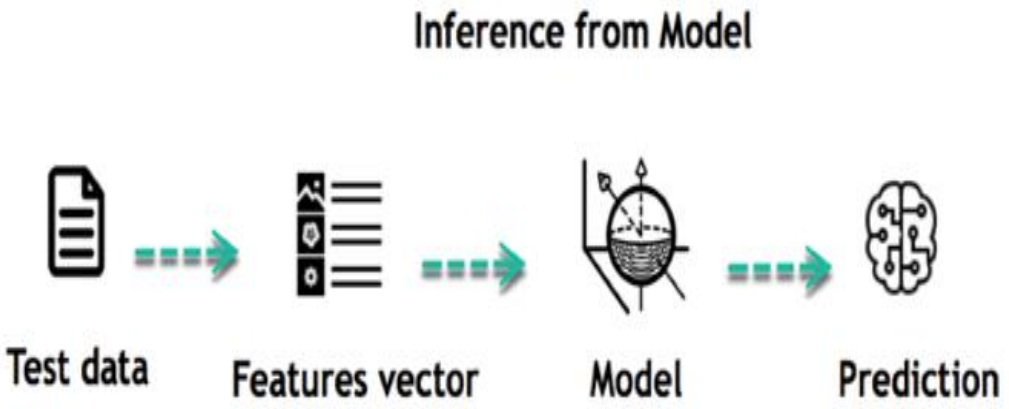
Inference from Model
Test data
Features vector
Model
Prediction

Fig 1.3

CHAPTER – 2 Problem Statement:

In this day and age, it is extremely difficult to decide whether the news we come across is real or not. There are very few options to check the authenticity and all of them are sophisticated and not accessible to the average person. There is an acute need for a web-based fact-checking platform that harnesses the power of Machine Learning to provide us with that opportunity.

Motivation:

Social media facilitates the creation and sharing of information that uses computer-mediated technologies. This media changed the way groups of people interact and communicate. It allows low cost, simple access and fast dissemination of information to them. The majority of people search and consume news from social media rather than traditional news organizations these days. On one side, where social media have become a powerful source of information and bringing people together, on the other side it also 1 put a negative impact on society. Look at some examples herewith; Facebook Inc's popular messaging service, WhatsApp became a political battle-platform in Brazil's election. False rumours, manipulated photos, de-contextualized videos, and audio jokes were used for campaigning. These kinds of stuff went viral on the digital platform without monitoring their origin or reach. A nationwide block on major social media and messaging sites including Facebook and Instagram was done in Sri Lanka after multiple terrorist attacks in the year 2019. The government claimed that "false news reports" were circulating online. This is evident in the challenges the world's most powerful tech companies face in reducing the spread of misinformation. Such examples show that Social Media enables the widespread use of "fake news" as well. The news Fake News Detector 9 disseminated on social media platforms may be of low quality carrying misleading information intentionally. This sacrifices the credibility of the information. Millions of news articles are being circulated every day on the Internet – how one can trust which is real and which is fake? Thus incredible or fake news is one of the biggest challenges in our digitally connected world. Fake news detection on social media has recently become an emerging research domain. The domain focuses on dealing with the sensitive issue of preventing the spread of fake news on social media. Fake news identification on social media faces several challenges. Firstly, it is difficult to collect fake news data. Furthermore, it is difficult to label fake news manually. Since they are intentionally written to mislead readers, it is difficult to detect them simply based on news content. Furthermore,

Facebook, Whatsapp, and Twitter are closed messaging apps. The misinformation disseminated by trusted news outlets or their friends and family is therefore difficult to be considered as fake. It is not easy to verify the credibility of newly emerging and time-bound news as they are not sufficient to train the application dataset. Significant approaches to differentiate credible users, extract useful news features and develop authentic information dissemination systems are some useful domains of research and need further investigations. If we can't control the spread of fake news, the trust in the system will collapse. There will be widespread Fake News Detector 10 distrust among people. There will be nothing left that can be objectively used. It means the destruction of political and social coherence. We wanted to build some sort of web-based system that can fight this nightmare scenario. And we made some significant progress towards that goal.

OBJECTIVE:

- The spreading of fake news has given rise to many problems in society. It is due to its ability to cause a lot of social and national damage with destructive impacts. Sometimes it gets very difficult to know if the news is genuine or fake. Therefore it is very important to detect if the news is fake or not.
- "Fake News" is a term used to represent fabricated news or propaganda comprising misinformation communicated through traditional media channels like print, and television as well as non-traditional media channels like social media. Techniques of NLP and Machine learning can be used to create models which can help to detect fake news. In this we have presented many models using the techniques of NLP and ML. The datasets in comma- separated values format, pertaining to political domain were used in the project. The different attributes like the title and text of the news headline/article were used to perform the fake newsdetection.
- The results showed that the proposed solution performs well in terms of providing an output with good accuracy, precision and recall. Further, a larger dataset for better output and also other factors such as the author , publisher of the news can be used to determine the credibility of the news. Also, further research can also be done on images, videos, images containing text which can help in improving the models in future.

CHAPTER – 3 LITERATURE REVIEW:

M. Granik et.al proposed a simple approach for the detection of fake news by using Naive BayesClassifier. They tested it against a dataset of Facebook news posts. They also made use of the BuzzFeed news dataset. They achieved classification accuracy of approximately 74% on the test set.

Niall J.Conroy et.al designed a basic fake news detector that provides high accuracy for classification tasks. They used the linguistic cues approaches and network analysis approach init. Both approaches adopt machine learning techniques for training classifiers to suit the analysis.They achieved an accuracy of 72% which could be improved. This could be done if the size of the input feature vector is reduced and also by performing cross-corpus analysis of the classification models.

R. Barua et.al identified if a news article is real or misleading by using an ensemble technique using recurrent neural networks (LSTM and GRU). An android application was also developed for determining the sanctity of a news article. They tested this model on a large dataset which was prepared in their work. The limitation of this method was that it required the article to be ofa particular size. It would give wrong predictions if the article was not enough to generate a summary.

B. Bhutani et.al used sentiment as an important feature to improve the accuracy of detecting fakenews. They have used 3 different datasets. They used Count vectorizer, Tf-Idf vectorizer along with cosine similarity and Bi-grams ,Tri-grams methods. The methods used to train the model are Naive Bayes and Random forest. They used different performance metrics to evaluate the model. They got an accuracy of 81.6%.

M. Vohra et.al proposed, a rumor detection system which determine the authenticity of an information and classify it as rumor or not a rumor. Data was collected by Twitter API. To generate topics from the preprocessed data, topic modelling was performed via Latent Dirichlet Allocation(LDA).They did web scraping on 4 trusted news website. After scraping these sites for articles the links of these articles are save and displayed in the GUI. These keywords were searched on their selected four news websites and news articles were

extracted from the results. If no article was found in all the four sites the new assigned that topic as rumor otherwise if article was found its was assigned as not a rumor.

CHAPTER – 4 Feasibility Study:

Passive-aggressive classifier, logistic regression, LSTM can be used in fake news detection. Bi-directional LSTM was used to detect fake news. It had reasonably good accuracy but if the news was a bit more sophisticated, it would be difficult to achieve good accuracy. Because this model picks up the sensational/clickbaity words as part of fake news. For example, if a news title says, 'Donald Trump is the greatest president ever, the model will pick it up as fake news with reasonable accuracy. If the title is more nuanced and written in a sophisticated way, it'd be difficult to do so. We believe that our LSTM model is not enough by itself to detect fake news. That's why we included passive aggressive classifier with it and when we compared passive news with reputable news sources, but the scope of the work is so vast that we couldn't do it with the resources available to us. Our model can act as a first step in detecting fake news. But more work is needed to call the model reliable enough.

CHAPTER – 5 Tools and Techniques:

The dataset we'll use for this python project- we'll call it news.csv. This dataset has a shape of 7796×4. The first column identifies the news, the second and third are the title and text, and the fourth column has labels denoting whether the news is REAL or FAKE.

	A	B	C	D
1		title	text	label
2	8476	You Can Smell Hillaryâ€™s Fear	Daniel Greenfield, a	FAKE
3	10294	Watch The Exact Moment Paul Ryan Committed Political Su	Google Pinterest Digg	FAKE
4	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John	REAL
5	10142	Bernie supporters on Twitter erupt in anger against the DNC	â€" Kaydee King	FAKE
6	875	The Battle of New York: Why This Primary Matters	It's primary day in New	REAL
7	6903	Tehran, USA		FAKE
8	7341	Girl Horrified At What She Watches Boyfriend Do After He L	Share This Baylee Luciani	FAKE
9	95	â€˜Britainâ€™s Schindlerâ€™ Dies at 106	A Czech stockbroker who sa	REAL
10	4869	Fact check: Trump and Clinton at the 'commander-in-chief'	Hillary Clinton and Donald	REAL
11	2909	Iran reportedly makes new push for uranium concessions in	Iranian negotiators	REAL
12	1357	With all three Clintons in Iowa, a glimpse at the fire that ha	CEDAR RAPIDS, Iowa â€"	REAL
13	988	Donald Trumpâ€™s Shockingly Weak Delegate Game Somel	Donald Trumpâ€™s	REAL
14	7041	Strong Solar Storm, Tech Risks Today | S0 News Oct.26.201	Click Here To Learn More	FAKE
15	7623	10 Ways America Is Preparing for World War 3	October 31, 2016 at 4:52	FAKE
16	1571	Trump takes on Cruz, but lightly	Killing Obama administratior	REAL
17	4739	How women lead differently	As more women move into	REAL
18	7737	Shocking! Michele Obama & Hillary Caught Glamorizing Dat	Shocking! Michele Obama	FAKE
19	8716	Hillary Clinton in HUGE Trouble After America Noticed SICK	0	FAKE
20	3304	What's in that Iran bill that Obama doesn't like?	Washington (CNN) For	REAL
21	3078	The 1 chart that explains everything you need to know abou	While paging through Pew's	REAL
22	2517	The slippery slope to Trumpâ€™s proposed ban on Muslims	With little fanfare this fall,	REAL
23	10348	Episode #160 â€" SUNDAY WIRE: â€˜Hail to the Deplorable	November 13, 2016 By	FAKE
24	778	Hillary Clinton Makes A Bipartisan Appeal on Staten Island	Hillary Clinton told a Staten	REAL
25	3300	New Senate majority leaderâ€™s main goal for GOP: Donâ€	Mitch McConnell has an	REAL
26	6155	â€˜Infernoâ€™ and the Overpopulation Myth	Mises.org November 1,	FAKE
27	636	Anti-Trump forces seek last-ditch delegate revolt	Washington (CNN) The	REAL
28	755	Sanders Trounces Clinton in W. Va. -- But Will It Make a Dif	Meanwhile, Democrat	REAL
29	626	Donald Trump Is Changing His Campaign Slogan to Prove He	After a week of nonstop	REAL
30	691	Pure chaos: Donald Trumpâ€™s campaign management off	If you want a glimpse into a	REAL
31	5743	Syrian War Report â€" November 1, 2016: Syrian Military D	Syrian War Report â€"	FAKE

Table 1

This advanced python project of detecting fake news deals with fake and real news. Using sklearn, we build a TfidfVectorizer on our dataset. Then, we initialize a PassiveAggressive

Classifier and fit the model. In the end, the accuracy score and theconfusion matrix tell us how well our model fares.

Passive Aggressive algorithms are online learning algorithms. Such an algorithm remains passive for a correct classification outcome, and turns aggressive in the event of a miscalculation, updating and adjusting. Unlike most other algorithms, it does not converge. Its purpose is to make updates that correct the loss, causing very little change in the norm of the weight vector.

TF (Term Frequency): The number of times a word appears in a document is its Term Frequency. A higher value means a term appears more often than others, and so, the document is a good match when the term is part of the search terms.

IDF (Inverse Document Frequency): Words that occur many times a document, but also occur many times in many others, may be irrelevant. IDF is a measure of how significant a term is in the entire corpus.

The TfidfVectorizer converts a collection of raw documents into a matrix of TF-IDFfeatures..

Python:

Python is a high-level, general-purpose and a very popular programming language. Python programming language (latest Python 3) is being used in web development, Machine Learning applications, along with all cutting edge technology in Software Industry. Python Programming Language is very well suited for Beginners, also for experienced programmers with other programmin languages like C++ and Java. .

This specially designed Python tutorial will help you learn Python Programming Language in most efficient way, with the topics from basics to advanced (like Web-scraping, Django, Deep-Learning, etc.) with examples.

Below are some facts about Python Programming Language:

1. Python is currently the most widely used multi-purpose, high-level programming language.
2. Python allows programming in Object-Oriented and Procedural paradigms.
3. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.
4. Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.
5. The biggest strength of Python is huge collection of standard library which can be used for the following:

    - Machine learning
    - GUI Applications (lik, Tkinter, PyQt etc.)
    - Web frameworks (used by YouTube, Instagram, Dropbox)
    - Image processing (like Pillow)
    - Web scraping (like Scrapy, BeautifulSoup, Selenium)
    - Test frameworks
    - Multimedia
    - Scientific computing
    - Text processing and many more..

Libraries in Python:

Normally, a library is a collection of books or is a room or place where many books are stored to be used later. Similarly, in the programming world, a library is a collection of precompiled codes that can be used later on in a program for some specific well-defined operations. Other than pre-compiled codes, a library may contain documentation, configuration data, message templates, classes, and values, etc.

A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don't need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, etc.

Working of Python Library:

As is stated above, a Python library is simply a collection of codes or modules of codes that we can use in a program for specific operations. We use libraries so that we don't need to write the code again in our program that is already available. But how it works. Actually, in the MS Windows environment, the library files have a DLL extension (Dynamic Load Libraries). When we link a library with our program and run that program, the linker automatically searches for that library. It extracts the functionalities of that library and interprets the program accordingly. That's how we use the methods of a library in our program. We will see further, how we bring in the libraries in our Python programs.

Python standard library:

The Python Standard Library contains the exact syntax, semantics, and tokens of Python. It contains built-in modules that provide access to basic system functionality like I/O and some other core modules. Most of the Python Libraries are written in the C programming language. The Python standard library consists of more than 200 core modules. All these work together to make Python a high-level programming language. Python Standard Library plays a very important role. Without it, the programmers can't have access to the functionalities of Python. But other than this, there are several other libraries in Python that make a programmer's life easier. Let's have a look at some of the commonly used libraries:

1. **TensorFlow:** This library was developed by Google in collaboration with the Brain Team. It is an open-source library used for high-level computations. It is also used in machine learning and deep learning algorithms. It contains a large number of tensor operations. Researchers also use this Python library to solve complex computations in Mathematics and Physics.

2. **Matplotlib:** This library is responsible for plotting numerical data. And that's why it is used in data analysis. It is also an open-source library and plots high-defined figures like pie charts, histograms, scatterplots, graphs, etc.
3. **Pandas:** Pandas are an important library for data scientists. It is an open-source machine learning library that provides flexible high-level data structures and a variety of analysis tools. It eases data analysis, data manipulation, and cleaning of data. Pandas support operations like Sorting, Re-indexing, Iteration, Concatenation, Conversion of data, Visualizations, Aggregations, etc.
4. **Numpy:** The name "Numpy" stands for "Numerical Python". It is the commonly used library. It is a popular machine learning library that supports large matrices and multi-dimensional data. It consists of in-built mathematical functions for easy computations. Even libraries like TensorFlow use Numpy internally to perform several operations on tensors. Array Interface is one of the key features of this library.
5. **SciPy:** The name "SciPy" stands for "Scientific Python". It is an open-source library used for high-level scientific computations. This library is built over an extension of Numpy. It works with Numpy to handle complex computations. While Numpy allows sorting and indexing of array data, the numerical data code is stored in SciPy. It is also widely used by application developers and engineers.
6. **Scrapy:** It is an open-source library that is used for extracting data from websites. It provides very fast web crawling and high-level screen scraping. It can also be used for data mining and automated testing of data.
7. **Scikit-learn:** It is a famous Python library to work with complex data. Scikit-learn is an open-source library that supports machine learning. It supports variously supervised and unsupervised algorithms like linear regression, classification, clustering, etc. This library works in association with Numpy and SciPy.
8. **PyGame:** This library provides an easy interface to the Standard Directmedia Library (SDL) platform-independent graphics, audio, and input libraries. It is used for developing video games using computer graphics and audio libraries along with Python programming language.

9. **PyTorch:** PyTorch is the largest machine learning library that optimizes tensor computations. It has rich APIs to perform tensor computations with strong GPU acceleration. It also helps to solve application issues related to neural networks.
10. **PyBrain:** The name "PyBrain" stands for Python Based Reinforcement Learning, Artificial Intelligence, and Neural Networks library. It is an open-source library built for beginners in the field of Machine Learning. It provides fast and easy-to-use algorithms for machine learning tasks. It is so flexible and easily understandable and that's why is really helpful for developers that are new in research fields.

There are many more libraries in Python. We can use a suitable library for our purposes. Hence, Python libraries play a very crucial role and are very helpful to the developers.

Use of Libraries in Python Program:

As we write large-size programs in Python, we want to maintain the code's modularity. For the easy maintenance of the code, we split the code into different parts and we can use that code later ever we need it. In Python, *modules* play that part. Instead of using the same code in different programs and making the code complex, we define mostly used functions in modules and we can just simply import them in a program wherever there is a requirement. We don't need to write that code but still, we can use its functionality by importing its module. Multiple interrelated modules are stored in a library. And whenever we need to use a module, we import it from its library. In Python, it's a very simple job to do due to its easy syntax. We just need to use **import**.

TfidfVectorizer:

In NLP, tf-idf is an important measure and is used by algorithms like cosine similarity to find documents that are similar to a given search query.

we will break tf-idf and sklearn's TfidfVectorizer calculates tf-idf values. I had a hard time matching the tf-idf values generated by TfidfVectorizer and with the ones I calculated. The reason is that there are many ways in which tf-idf values are calculated, and we need to be aware

of the method that TfidfVectorizer uses to calculate tf-idf. This will save a lot of time and effort for you. I spent a couple of days troubleshooting before I could realize the issue.

We will write a simple Python program that uses TfidfVectorizer to calculate tf-idf and manually validate this. Before we get into the coding part, let's go through a few terms that make up tf-idf.

What is Term Frequency (tf)

tf is the number of times a term appears in a particular document. So it's specific to a document. A few of the ways to calculate tf is given below:-

tf(t) = No. of times term 't' occurs in a document

OR

tf(t) = (No. of times term 't' occurs in a document) / (No. Of terms in a document)

OR

tf(t) = (No. of times term 't' occurs in a document) / (Frequency of most common term in a document)

sklearn uses the first one i:e No. Of times a term 't' appears in a document

Inverse Document Frequency (idf)

idf is a measure of how common or rare a term is across the entire corpus of documents. So the point to note is that it's common to all the documents. If the word is common and appears in many documents, the idf value (normalized) will approach 0 or else approach 1 if it's rare. A few of the ways we can calculate idf value for a term is given below

$idf(t) = 1 + \log_e [n / df(t)]$

OR

$idf(t) = \log_e [n / df(t)]$

where

n = Total number of documents available

t = term for which idf value has to be calculated

df(t) = Number of documents in which the term t appears

But as per sklearn's online documentation, it uses the below method to calculate idf of a term in a document.

$idf(t) = \log_e [(1+n) / (1 + df(t))] + 1$ (default i:e smooth_idf = True)

and

$idf(t) = \log_e [n / df(t)] + 1$ (when smooth_idf = False)

Term Frequency-Inverse Document Frequency (tf-idf)

tf-idf value of a term in a document is the product of its tf and idf. The higher is the value, the more relevant the term is in that document.

Passive aggressive classifier:

Passive-Aggressive algorithms are called so because

Passive- If the prediction is correct, keep the representation and do not make any interchanges. i.e., the data in the example is not enough to cause any changes in the representation. Aggressive- If the prediction is incorrect, make interchanges to the representation. i.e., some interchange to the representation may correct it.

Understanding the mathematics supporting this algorithm is not very simple and is supporting the scope of a single article. This section provides just an overview of the algorithm and a simple implementation of it. To learn more about the mathematics supporting this algorithm

The **passive aggressive classifier** algorithm falls under the category of online learning algorithms, can handle large datasets, and updates its model based on each new instance it encounters. The passive aggressive algorithm is an online learning algorithm, which means that it can update its weights as new data comes in. The passive aggressive classifier has a parameter, namely, the regularization parameter, C that allows for a tradeoff between the size of the margin and the number of misclassifications. In each iteration, the passive aggressive classifier looks at a new instance, assesses whether it has been correctly classified or not, and then updates its weights accordingly. If the instance is correctly classified, there is no change in weight. However, if it is misclassified, the passive aggressive algorithm adjusts its weights in order to better classify future instances based on this misclassified instance. The degree to which the Passive Aggressive algorithm adjusts its weights is dependent on the regularization parameter C and how confident it is in the classification of that particular instance.

As with other supervised learning algorithms, the passive aggressive classifier works by taking a set of training data and dividing it into two groups: a training set and a test set. The passive aggressive classifier then uses the training set to learn how to correctly classify objects into one of two categories. Once it has learned how to do this, it is then tested on the data in the test set, and its accuracy is measured.

The passive aggressive classifier can be trained using a variety of different loss functions, such as the **hinge loss (PA-I)** or the **squared hinge loss (PA-II)**. The hinge loss is a linear loss function that is used to minimize the distance between two decision boundaries. This makes it a good choice for situations where you want to classify objects into two categories as accurately as possible. For example, if you are using the passive aggressive classifier to identify cancer cells, you would want to use the hinge loss function so that the boundaries between cancer cells and healthy cells are as distinct as possible. The squared hinge loss is a nonlinear loss function that is used to minimize the distance between two decision boundaries. The squared hinge loss is a variant of the hinge loss that is typically used when the data has a Gaussian distribution. The squared hinge loss is similar to the hinge loss, but it takes into account the variance of the data. This can be helpful when trying to minimize the error in prediction.

Real-world examples – Passive Aggressive Classifier:

The passive aggressive classifier is used to classify data points into two groups. This algorithm can be used in a variety of different settings, including:

- Spam filtering: The passive aggressive classifier can be used to filter spam emails by training the algorithm on a dataset of known spam emails.
- Fraud detection: The passive aggressive classifier can be used to detect fraudulent transactions by training the algorithm on a dataset of known fraud cases.
- Text classification: The passive aggressive classifier can be used to classify text documents into different categories, such as news articles or blog posts.

These are just a few examples of how the passive aggressive algorithm can be used in the real world. There are many other applications for this powerful machine learning algorithm.

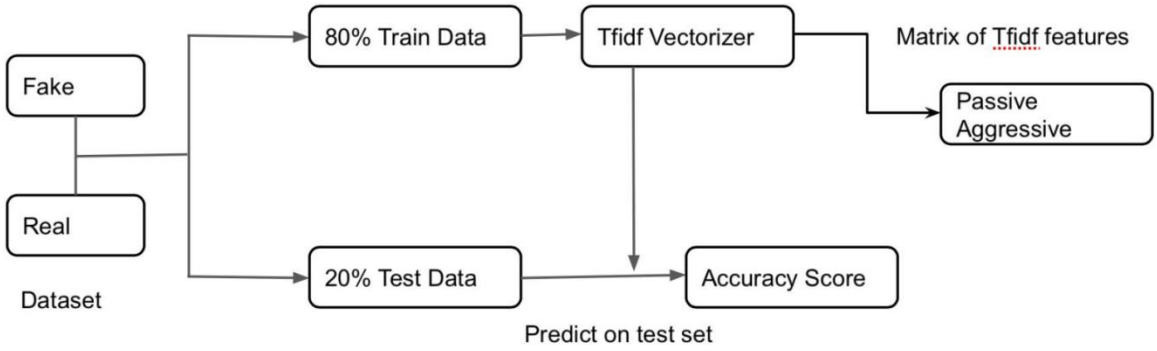

Figure 2 : Passive-aggressive model

Chapter – 6

Process Description and Various Stages of Project Implementation:

You'll need to install the following libraries with pip:

```
pip install numpy pandas sklearn
```

Follow the below steps for detecting fake news and complete your first advanced Python Project

1. Make necessary imports:

import numpy
as np import
pandas as pd
import itertools

from sklearn.model_selection import train_test_split

from sklearn.feature_extraction.text import TfidfVectorizer from sklearn.linear_model import PassiveAggressiveClassifier from sklearn.metrics import accuracy_score, confusion_matrix

```
[ ]  import numpy as np
     import pandas as pd
     import itertools
     from sklearn.model_selection import train_test_split
     from sklearn.feature_extraction.text import TfidfVectorizer
     from sklearn.linear_model import PassiveAggressiveClassifier
     from sklearn.metrics import accuracy_score, confusion_matrix
```

2. Now, let's read the data into a DataFrame, and get the shape of the data and the first 5 recordsand get the labels from the DataFrame.

```
df=pd.read_csv("/content/news.zip")

df.shape
df.head()
```

	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE

```
#DataFlair - Get the labels
labels=df.label
labels.head()
```

```
0    FAKE
1    FAKE
2    REAL
3    FAKE
4    REAL
Name: label, dtype: object
```

3. Split the dataset into training and testing sets.

```
[ ]  #DataFlair - Split the dataset
     x_train,x_test,y_train,y_test=train_test_split(df['text'], labels, test_size=0.2, random_state=7)
```

4. Let's initialize a TfidfVectorizer with stop words from the English language and a maximum document frequency of 0.7 (terms with a higher document frequency will be discarded). Stop words are the most common words in a language that are to be filtered out before processing the natural language data. And a TfidfVectorizer turns a collection of raw documents into a matrix ofTF-IDF features.

Now, fit and transform the vectorizer on the train set, and transform the vectorizer on the test set.

```
[ ]  #DataFlair - Initialize a TfidfVectorizer
     tfidf_vectorizer=TfidfVectorizer(stop_words='english', max_df=0.7)
     #DataFlair - Fit and transform train set, transform test set
     tfidf_train=tfidf_vectorizer.fit_transform(x_train)
     tfidf_test=tfidf_vectorizer.transform(x_test)
```

5. Next, we'll initialize a PassiveAggressiveClassifier. This is. We'll fit this on tfidf_train andy_train.

```
[ ]  #DataFlair - Initialize a PassiveAggressiveClassifier
     pac=PassiveAggressiveClassifier(max_iter=50)
     pac.fit(tfidf_train,y_train)
     #DataFlair - Predict on the test set and calculate accuracy
     y_pred=pac.predict(tfidf_test)
     score=accuracy_score(y_test,y_pred)
     print(f'Accuracy: {round(score*100,2)}%')
```

Then, we'll predict on the test set from the TfidfVectorizer and calculate the accuracy with accuracy_score() from sklearn.metrics.

```
     #DataFlair - Predict on the test set and calculate accuracy
     y_pred=pac.predict(tfidf_test)
     score=accuracy_score(y_test,y_pred)
     print(f'Accuracy: {round(score*100,2)}%')

     Accuracy: 92.66%
```

Finally, let's print out a confusion matrix to gain insight into the number of false and true negatives and positives.

```
     #DataFlair - Build confusion matrix
     confusion_matrix(y_test,y_pred, labels=['FAKE','REAL'])

     array([[588,  50],
            [ 43, 586]])
```

Chapter – 7

Conclusion:

We have achieved the accuracy of 92.66%.Our project can ring the initial alert for fake news. The model produces worse results if the article is written cleverly, without any sensationalization. This is a very complex problem but we tried to address it as much as we could. We believe the interface provides an easier way for the average person to check the authenticity of a news. Projects like this one with more advanced features should be integrated on social media to prevent the spread of fake news

Future Work :

There are many future improvement aspects of this project. Introducing a cross checking feature on the machine learning model so it compares the news inputs with the reputable news sources is one way to go. It has to be online and done in real time, which will be very challenging. Improving the model accuracy using bigger and better datasets, integrating different machine learning algorithms is also something we hope to do in the future.

Also, further research can also be done on images, videos, images containing text which can help in improving the models in future.

References:

- Fake news detection in social media by Kelly Stahl.
- **https://www.kaggle.com/code/barkhaverma/fake-news-detection**
- **https://colab.research.google.com/**
- **https://en.wikipedia.org/wiki/Natural_language_processing**
- **https://www.researchgate.net/publication/343916946_Fake_News_Detection_Using_Machine_Learning_Algorithms**
- **https://www.geeksforgeeks.org/python-programming-language**
- journalstd.com