DEMOCRATIC AND POPULAR REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

UNIVERSITY OF MOHAMED BOUDIAF - M'SILA
FACULTY OF MATHEMATICS
AND INFORMATICS
DEPARTMENT OF COMPUTER SCIENCE

A Dissertation in Fulfillment
For the Requirements of the Degree of Master
in Computer Science

DOMAINE: Mathematics and Informatics

FILIERE: Informatics

OPTION: Information Systems and Software Engineering

Entitled

Fake News Detection

By: NOUIOUA Saad **and** OUCIF Tedj Eddine

Presented publicly to the jury:

Bouzaaroura Ahlem	University of M'sila	Superviser
Lakhal Meftah	University of M'sila	Reviewer
Halassa Madiha	University of M'sila	Reviewer

Academic Year: 2021/2022

Acknowledgements

First, Alhamdulillah.
Alhamdulillah for this moment, if you put your trust in Allah, Allah won't let you down.
I want to thank Dr. Bouzaaroura for helping us whenever we get stuck, and guiding us whenever we make a mistake.
To my parents for being by my side since my first day in primary school, I would never reach this level of knowledge without them.
As a big brother, it is a responsibility to be a good example for my sisters to follow my lead, which is what they're doing now, God bless them.
A special thanks to my brother from another mother Saad Nouioua, for being by my side from the first day in school until the last day.
To everyone who helped me in my career of seeking Knowledge.

- Tedj Eddine Oucif

First of all, thanks to Allah for guiding me to the right path ,with his wisdom I'm here now, Alhamdulillah. I would like to thank Dr A.Bouzaaroura for helping and managing us to finish this work, and for all her advice and Instructions.
All Thanks to my parents, my mother and my father, who raised me and took care of me for all of these years.
Special thanks to my older sister, for every beautiful thing she gave me, and my youngest brother and sister. without missing to thank my soulmate Tedj Eddine Oucif, for all the years we spent together.
Finally I would like to thank all my family, friends and all my precious people.

- Saad Nouioua

Content

List of Figures

List of Tables

General Introduction

There are many sources of information in our time, and they are no longer limited to what is broadcast by the radio or television, from news bulletins and others. Rather, it went to web pages, blogs and social media platforms, which became the main source of information, whether true or false.

Social networks have become the main source of information for the majority, due to the ease of access to them and the large number of so-called influencers, who are considered reliable sources of information despite the lack of credibility of most of them, and for this reason we notice a lot of false news and rumors on the pages of social networking sites and the ease of spreading fake news, especially with the availability of the feature Participation, which contributes to facilitating its spread and increasing its impact.

As a result, fake news, regardless of its credibility, has spread quickly and easily, especially in the last decade. Fake news publishers take advantage of criticism. Situations like the Covid-19 pandemic and the US presidential election are having a negative impact on societies. Fake news can seriously affect society in many areas, including politics, finance, sports, etc.

Our work fits into this context, dealing with news and pieces of information is a major challenge in terms of being able to recognize fake and real ones.

1 Objective

So, the objective of our work is the design and the realization of a news classification system capable of detecting fakes and real ones

The work is based on the machine learning methods and the natural language processing techniques .

2 Manuscript Organization

This document is organized into the following chapters:

Chapter 1: Machine Learning and Deep learning

Chapter 2: Natural Language Processing

Chapter 3: Fake News detection on social media

Chapter 4: The realized work and the obtained results.

CHAPTER 1

Machine Learning and Deep learning

1 Introduction

Artificial intelligence (AI) has caught on lately. People from different disciplines are trying to apply AI to make their tasks much easier. For example, economists use AI to predict future market prices for profits, doctors use AI to classify whether a tumor is malignant or benign, meteorologists use AI to predict the weather, recruiters use AI to check applicants' resumes, to determine whether the applicant meets the minimum criteria for the position, etc.

Machine learning is without a doubt one of the most influential and powerful technologies in the world today. More importantly, we are far from realizing its full potential. There's no doubt that it will continue to make headlines for the expected future.

“I think AI is akin to building a rocket ship. You need a huge engine and a lot of fuel. If you have a large engine and a tiny amount of fuel, you won’t make it to orbit. If you have a tiny engine and a ton of fuel, you can’t even lift off. To build a rocket you need a huge engine and a lot of fuel” [1].

2 Machine Learning Definition

Machine Learning (ML) is a method of data analysis—and a type of artificial intelligence—that automates analytical model building, based on the idea that systems can learn from data, identify patterns, and make decisions with little human intervention It is defined as the collection of using various algorithms to teach computers to find patterns in data to be used for future prediction and forecasting or as a quality check for performance optimization. ML provides computers the ability to learn without being explicitly programmed [2].

ML features are found in smartphones and smart devices in the form of , including features like predictive text, speech recognition, computational photography, and so on. ML gives smart devices the ability to become more and more intuitive and proactive. Because ML relies on massive amounts of data, it is most often associated with intelligent devices within the Internet of Things (IoT) [3].

3 Brief Timeline History of Machine Learning

AI and machine learning algorithms aren’t new. Many scientists and researchers tried to find out if computers have real intelligence by self-learning without the intervention of human beans.

The field and the term of Machine Learning dates back to the 1950s. Arthur Lee Samuels, an IBM researcher, developed one of the earliest machine learning programs — a self-learning program for playing checkers. In fact, he coined the term machine learning.

His approach to machine learning was explained in a paper published in the IBM Journal of Research and Development in 1959 [4].

Between the 1950s and the beginning of this century, everything related with this fields was limited with the researches and experiments, like in 1957 Frank Rosenblatt designed the first neural network for computers (the perceptron), which simulate the thought processes of the human brain, and in the 1979 Students at Stanford University invent the "Stanford Cart" which can navigate obstacles in a room on its own, in 1980s Gerald Dejong introduces the concept of Explanation Based Learning (EBL), in which a computer analyzes training data and creates a general rule it can follow by discarding unimportant data.

In the 1990s, Work on machine learning shifted from a knowledge-driven approach to a data-driven approach. Scientists begin creating programs for computers to analyze large amounts of data and draw conclusions or "learn from the results".

Nowadays there is a permanent competition between a lot of tech companies such as Microsoft, IBM, Facebook, Twitter, even the industrial companies such as Tesla and other's one, to develop products and programs based on machine learning, and without human need [5].

4 Types of Machine learning approaches

Many machine learning models are defined by the presence or absence of human influence on the raw data, whether by offering a reward, providing specific feedback, or using labels [6].

- Supervised learning
- Unsupervised learning
- Semi-supervised learning
- Reinforcement learning

4.1 Supervised learning

Supervised learning is a machine learning approach that's defined by its use of labeled datasets. These datasets are designed to train or "supervise" algorithms into classifying data or predicting outcomes accurately. Using labeled inputs and outputs, the model can measure its accuracy and learn over time.

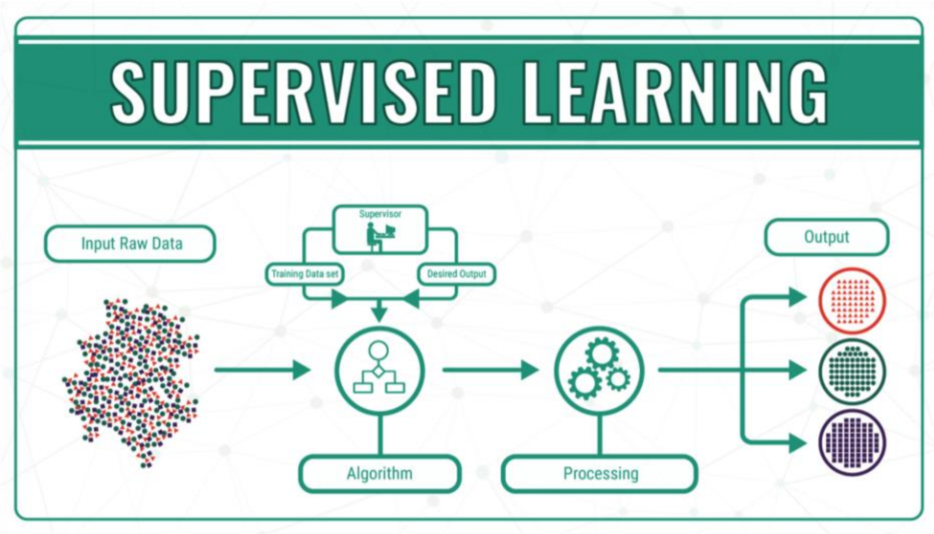

Figure 1.1 Supervised Learning[7].

Supervised learning can be separated into two types of problems when data mining: classification and regression:

4.1.1 Classification

problems use an algorithm to accurately assign test data into specific categories, such as separating apples from oranges. Or, in the real world, supervised learning algorithms can be used to classify spam in a separate folder from your inbox. Linear classifiers, support vector machines, decision trees and random forest are all common types of classification algorithms.

4.1.2 Regression

is another type of supervised learning method that uses an algorithm to understand the relationship between dependent and independent variables. Regression models are helpful for predicting numerical values based on different data points, such as sales revenue projections for a given business. Some popular regression algorithms are linear regression and logistic regression.

- **Linear regression:** Before we know what linear regression is, let's get used to regression. Regression is a method of modeling a target value based on independent predictors. This method is mainly used to predict and discover cause-effect relationships between variables. Regression techniques differ primarily based on the number of independent variables and the nature of the relationship between the independent and dependent variables. Simple linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent (x) and dependent variable (y). The red line in the graph above is known as the straight line of best fit. Based on the given data points, we try to draw a line that best models the points. The line can be modeled based on the linear equation shown below.

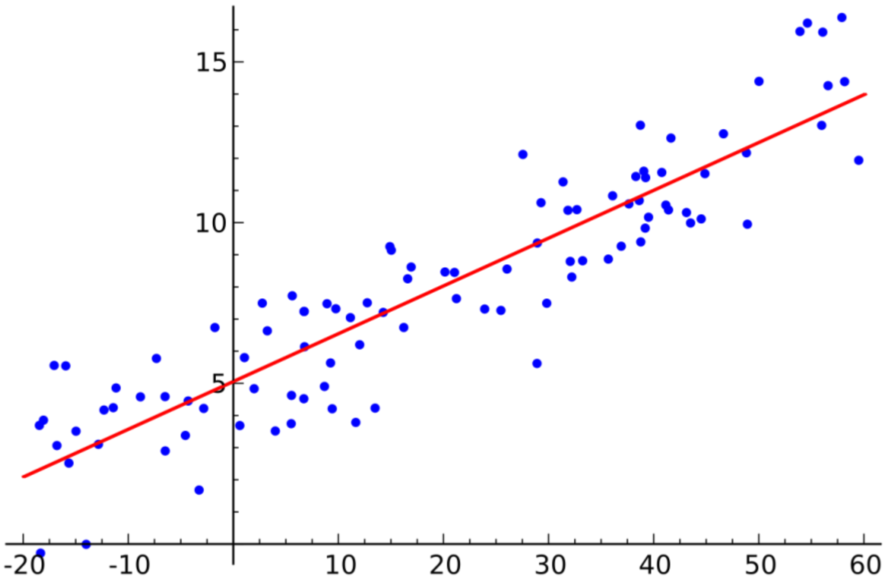

Figure 0.2 Linear regression[8].

- **Logistic regression:** Logistic regression is the most well-known machine learning algorithm after linear regression. In many ways, linear regression and logistic regression are similar. The biggest difference, however, is what they are used for. Linear regression algorithms are used to predict/predict values, but logistic regression is used for classification tasks. If you are unsure about linear regression concepts, check this out. There are many sorting tasks that people routinely perform. For example, classifying whether an email is spam or not, classifying whether a tumor is malignant or benign, classifying whether a website is fraudulent or not, etc. These are typical examples where machine learning algorithms significantly change our lives. A really simple, rudimentary, and useful classification algorithm is the logistic regression algorithm. Now let's take a closer look at logistic regression.

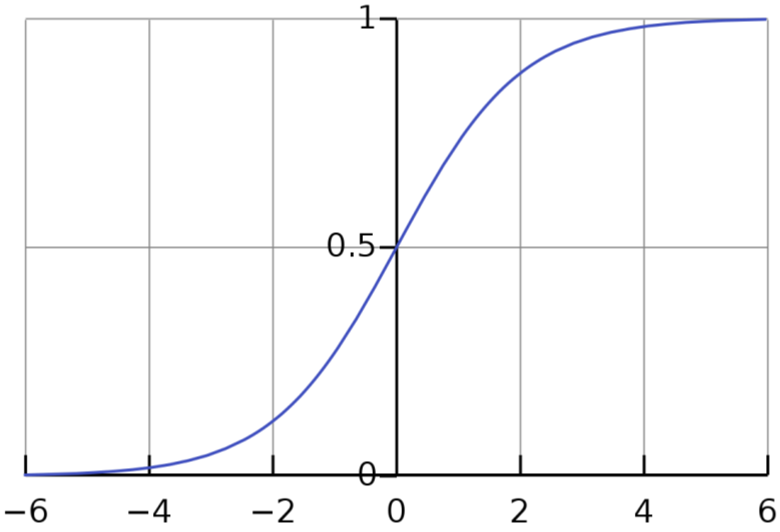

Figure 0.3 Logistic regression[9].

4.1.3 Support Vector Machine:

The Support Vector Machine is another simple algorithm that every machine learning expert should have in their arsenal. The support vector machine is preferred by many because it achieves significant accuracy with less computational power. The Support Vector Machine, abbreviated as SVM, can be used for both regression and classification tasks. However, it is often used for classification purposes. The goal of the Support Vector Machine algorithm is to find a hyperplane in N-dimensional space (N: the number of features) that classifies the data points properly. To separate the two classes of data points, many possible hyperplanes can be chosen. Our goal is to find a plane that has the maximum margin, i.e., the maximum distance between the data points of both classes. Maximizing the edge distance provides some gain so that future data points can be more reliably classified [10].

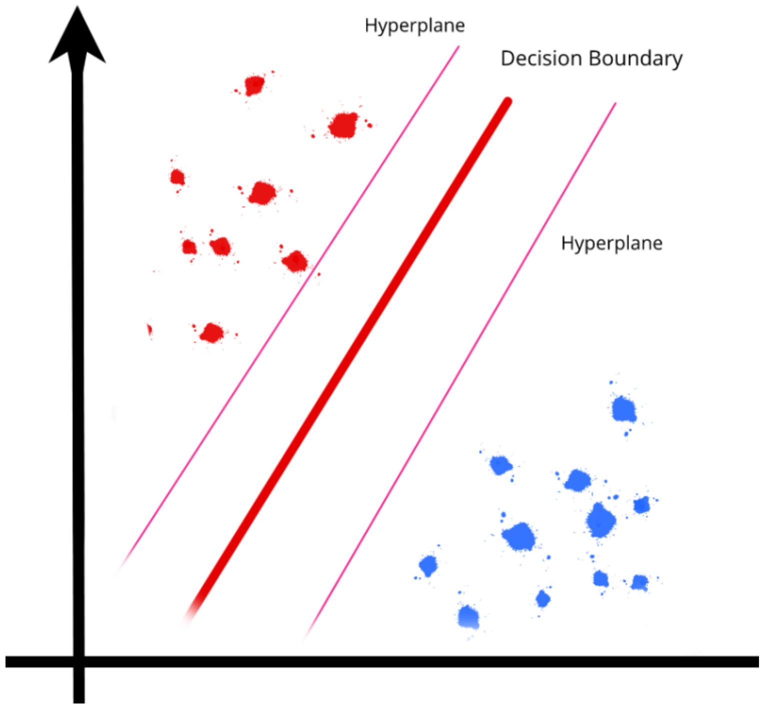

Figure 0.4 Support vector machine[11].

Supervised Machine Learning Applications

- Bioinformatics
- Quantitative structure
- Database marketing
- Handwriting recognition
- Information extraction
- Object recognition in computer vision
- Optical character recognition
- Spam detection
- Pattern recognition

4.2 Unsupervised learning

Unsupervised learning uses machine learning algorithms to analyze and cluster unlabeled data sets. These algorithms discover hidden patterns in data without the need for human intervention (hence, they are “unsupervised”) [12].

Unsupervised learning models are used for three main tasks: clustering, association and dimensionality reduction:

4.2.1 Clustering

is a data mining technique for grouping unlabeled data based on their similarities or differences. For example, K-means clustering algorithms assign similar data points into groups, where the K value represents the size of the grouping and granularity. This technique is helpful for market segmentation, image compression, etc.

4.2.2 Association rules

It is another type of unsupervised learning method that uses different rules to find relationships between variables in a given dataset. These methods are frequently used for market basket analysis and recommendation engines, along the lines of "Customers Who Bought This Item Also Bought" recommendations.

4.2.3 Dimensionality

reduction is a learning technique used when the number of features (or dimensions) in a given dataset is too high. It reduces the number of data inputs to a manageable size while also preserving the data integrity. Often, this technique is used in the preprocessing data stage, such as when autoencoders remove noise from visual data to improve picture quality.

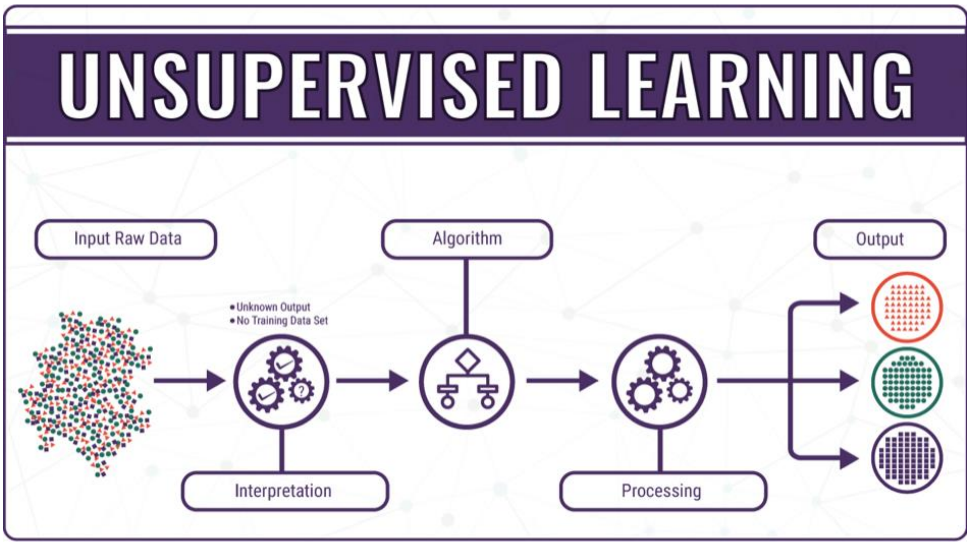

Figure 0.5 Unsupervised learning[13].

Its algorithms:

- K-Means Clustering
- Gaussian Mixture Model
- Hidden Markov Model
- Principal Component Analysis (PCA)
- Unsupervised Machine Learning Application:
- Human Behavior Analysis
- Social Network Analysis to define groups of friends.
- Market Segmentation of companies by location, industry, vertical.
- Organizing computing clusters based on similar event patterns and processes.

4.3 Semi-supervised learning

The dataset contains structured and unstructured data, which guide the algorithm on its way to making independent conclusions. The combination of the two data types in one training dataset allows machine learning algorithms to learn to label unlabeled data. It's particularly useful when it's difficult to extract relevant features from data — and when you have a high volume of data.

Common situations for this kind of learning are medical images like CT scans or MRIs. A trained radiologist can go through and label a small subset of scans for tumors or diseases. It would be too time-intensive and costly to manually label all the scans — but the deep learning network can still benefit from the small proportion of labeled data and improve its accuracy compared to a fully unsupervised model [14].

4.4 Reinforcement learning

Reinforcement machine learning is a behavioral machine learning model that is similar to supervised learning, but the algorithm isn't trained using sample data. This model learns as it goes by using trial and error. A sequence of successful outcomes will be reinforced to develop the best recommendation or policy for a given problem.

Reinforcement learning is the learning of a mapping from situations to actions so as to maximize a scalar reward or reinforcement signal. The learner is not told which action to take, as in most forms of machine learning, but instead must discover which actions yield the highest reward by trying them. In the most interesting and challenging cases, actions may affect not only the immediate reward, but also the next situation, and through that all subsequent rewards. These two characteristics—trial-and-error search and delayed reward—are the two most important distinguishing features of reinforcement learning [15].

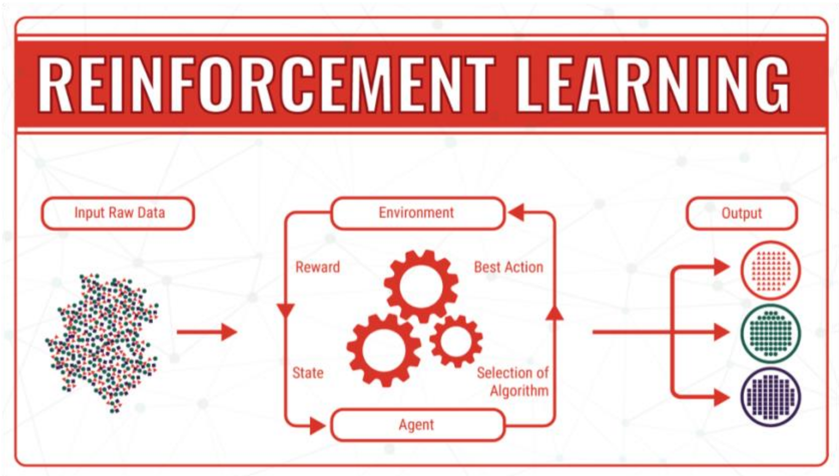

Figure 0.6 Reinforcement learning[16].

5 Deep Learning

Deep learning is a branch of machine learning and solves the feature selection problem above. Deep learning, also called hierarchical learning, achieves this by applying a learning method to spot subtle patterns in the data, then building a mathematical model and integrating it as the final classifier. What makes it special is that we no longer have to go crazy looking for the best features or attributes for each problem, this is done automatically by our algorithm. And that's why [17].

Today, the deep learning approach DL is the new trend in machine learning, as it provides far more advanced pattern recognition and picture categorization than the old machine learning approach ML [18].

6 Neural Network

An artificial neural network is a model of computation inspired by the structure of neural networks in the brain. In simplified models of the brain, it consists of a large number of basic computing devices (neurons) that are connected to each other in a complex communication network, through which the brain is able to carry out highly complex computations. Artificial neural networks are formal computation constructs that are modeled after this computation paradigm. Learning with neural networks was proposed in the mid-20th century. It yields an effective learning paradigm and has recently been shown to achieve cutting edge performance on several learning tasks. Neural network can be described as a directed graph whose nodes correspond to neurons and edges correspond to links between them. Each neuron receives as input a weighted sum of the outputs of the neurons connected to its incoming edges. We focus on feedforward networks in which the underlying graph does not contain cycles [19].

6.1 Activation function

Activation function is nothing but a mathematical function that takes in an input and produces an output. The function is activated when the computed result reaches the specified threshold [20].

6.1.1 Activation function types

- **Linear Activation Function:** The activation function simply scales an input by a factor, implying that there is a linear relationship between the inputs and the output.

  This is the mathematical formula:

$$Output = y * x$$

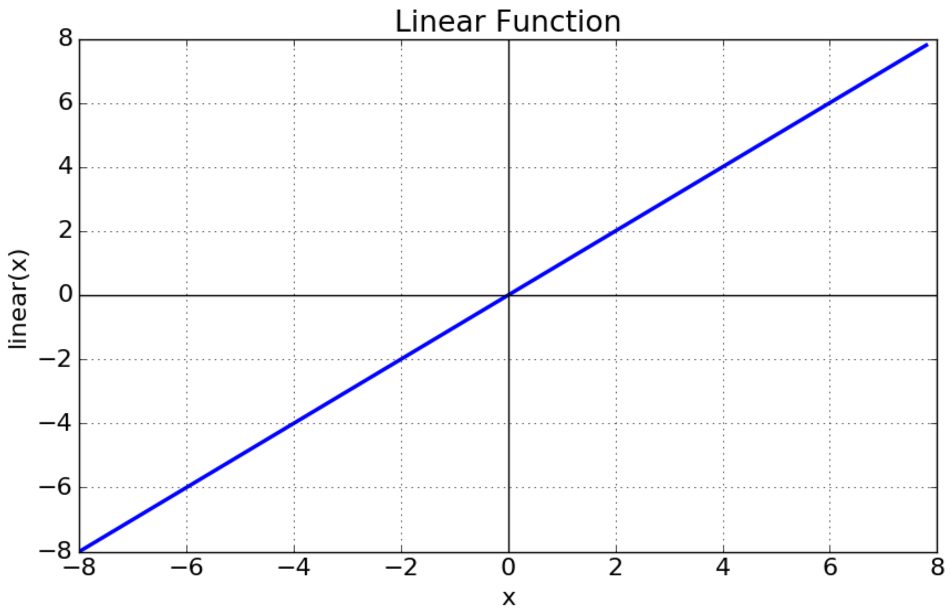

Figure 0.7 Linear Activation Function[21].

- **Sigmoid Activation Function:** The sigmoid activation function is "S" shaped. It can add non-linearity to the output and returns a binary value of 0 or 1.

$$Output = \frac{1}{1 + e^{-x}}$$

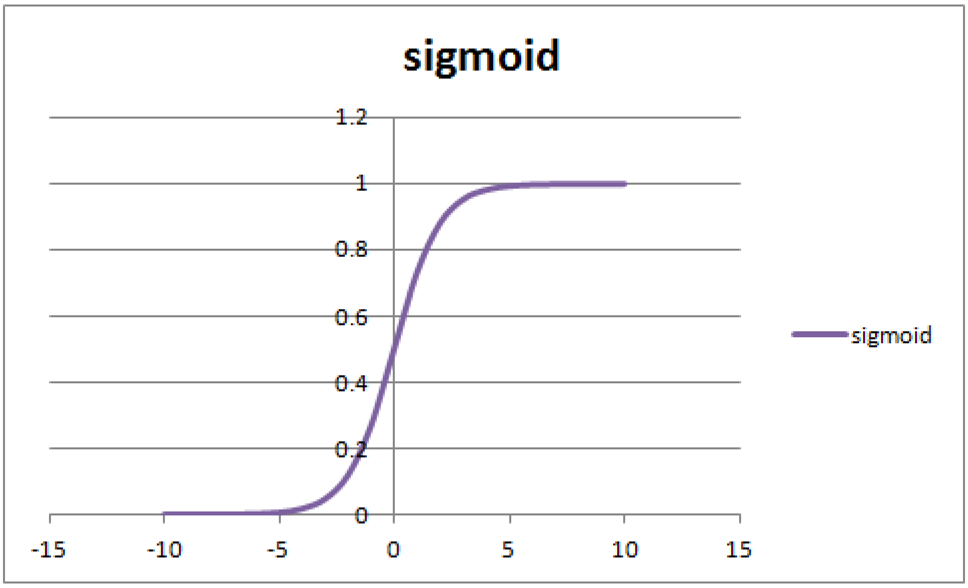

Figure 0.8 Sigmoid Activation Function[22].

- **Tanh Activation Function:** Tanh is an extension of the sigmoid activation function. Hence Tanh can be used to add non-linearity to the output. The output is within the range of -1 to 1. Tanh function shifts the result of the sigmoid activation function:

$$Output = \frac{2}{1 + e^{-2x}} - 1$$

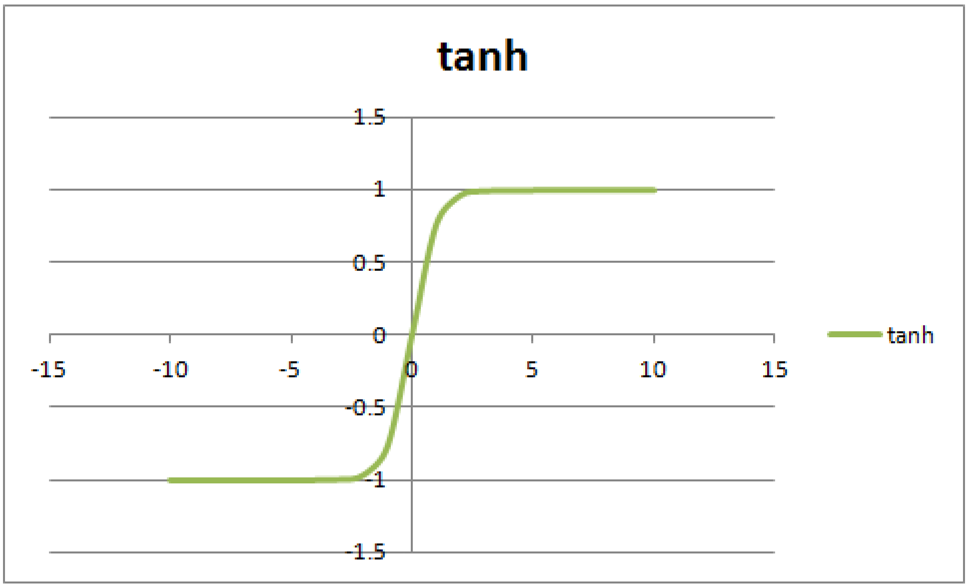

Figure 0.9 Tanh Activation Function[23].

- **Rectified Linear Unit Activation Function (RELU):** RELU is one of the most used activation functions. It is preferred to use RELU in the hidden layer. The concept is very straightforward. It also adds non-linearity to the output. However, the result can range from 0 to infinity.

$$Output = \max(0, input)$$

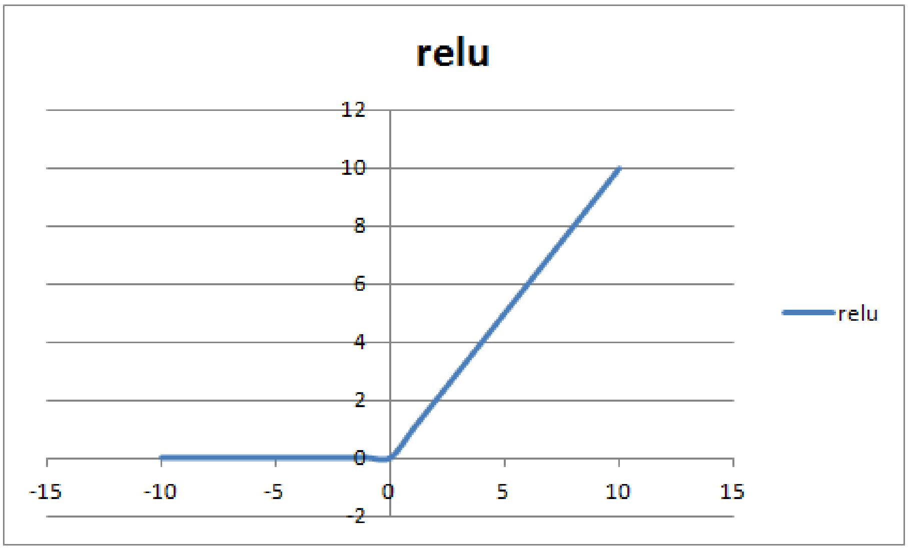

Figure 0.10 Relu Activation Function[24].

If you are unsure of which activation function you want to use then use RELU.

- **Softmax Activation Function:** Softmax is an extension of the Sigmoid activation function. Softmax function adds non-linearity to the output, however it is mainly used for classification examples where multiple classes of results can be computed.

$$Output = \frac{e^x}{Sum(e^x)}$$

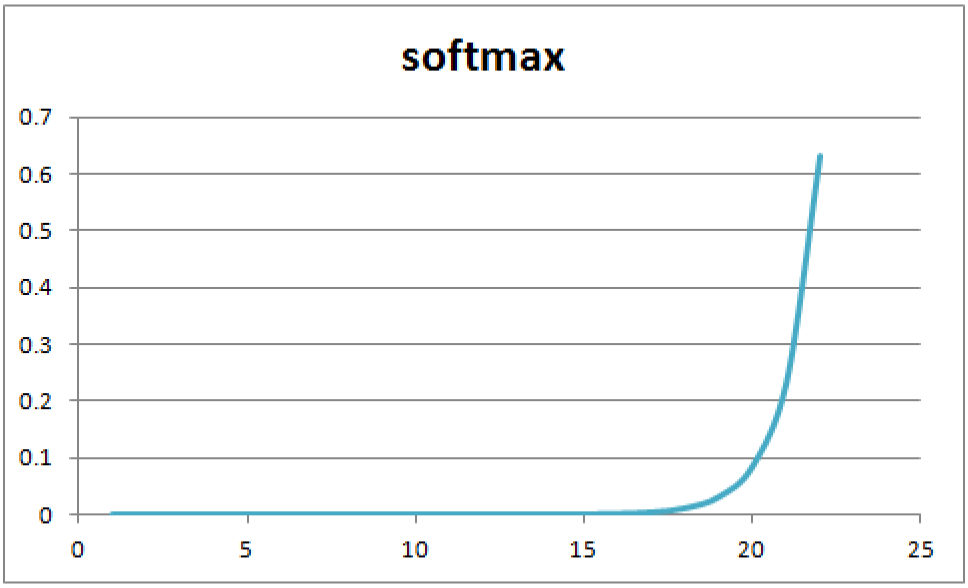

Figure 1.11 Softmax Activation Function[25].

6.2 Deep Neural Network types

Deep Learning has 3 types we'll see them

6.2.1 Artificial Neural Network:

Artificial Neural Network (ANN), is a group of multiple perceptrons/ neurons at each layer. ANN is also known as a Feed-Forward Neural network because inputs are processed only in the forward direction [26].

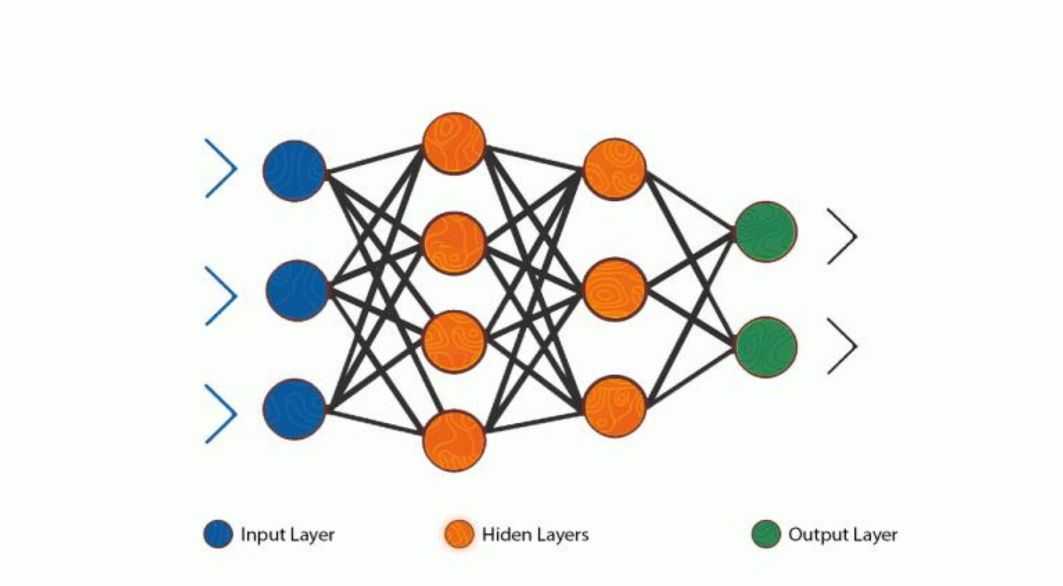

Figure 0.12 Artificial Neural Network[27].

The Neural Network is constructed from 3 type of layers:

1. Input layer — initial data for the neural network.
2. Hidden layers — intermediate layer between input and output layer and place where all the computation is done.
3. Output layer — produce the result for given inputs.

6.2.2 Convolutional Neural Network

A convolutional neural network (CNN) is a feed-forward neural network that is generally used to analyze visual images by processing data with grid-like topology.

CNN is used to detect and classify objects in an image [28].

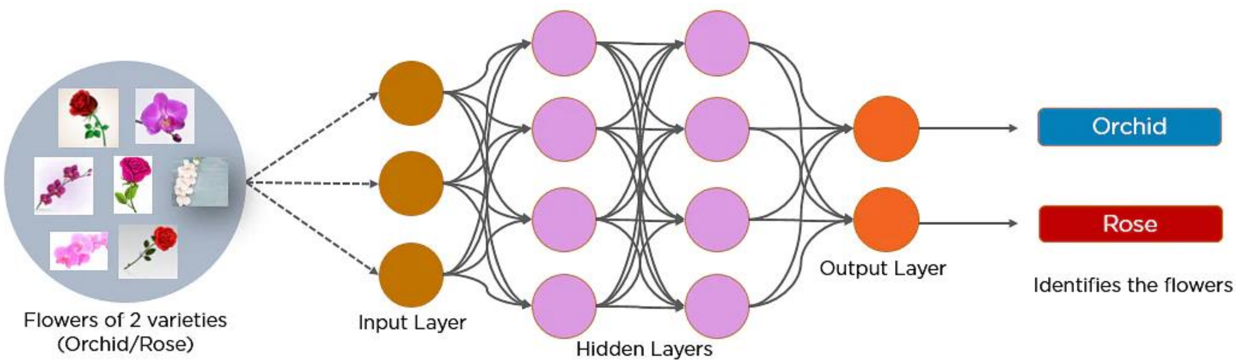

Figure 0.13 Convolutional Neural Network[29].

6.2.3 Recurrent Neural Network

Recurrent Neural Network (RNN) works on the principle of saving the output of a particular layer and feeding this back to the input in order to predict the output of the layer.

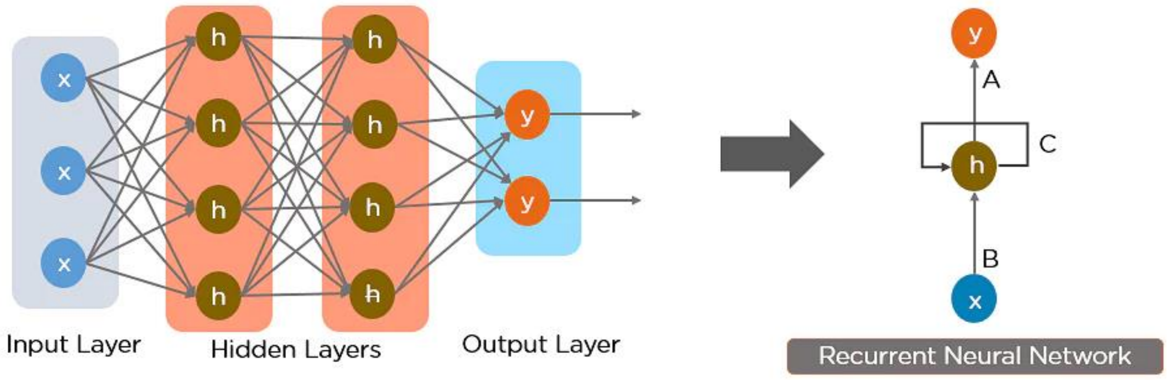

Figure 0.14 Recurrent Neural Network[30].

7 Machine Learning vs Deep Learning:

Properties	Machine learning	Deep Learning
Approach	Machine Learning uses algorithms to parse data, learn from that data, and make informed decisions based on what it has learned	Deep Learning structures algorithms in layers to create an “artificial neural network” that can learn and make intelligent decisions on its own
Time	Quick to train	Computationally intensive
Hardware	Able to function on CPU	Requires significant computing power
Output	The output is in numerical form for classification and scoring applications	The output can be in any form including free form elements such as free text and sound
Capability of training	Can train on lesser training data	Requires large data sets for training
Human invention	a human need to identify and hand-code the applied	tries to learn those features without additional human

	features based on the data type	invention

Table 1.1 Machine Learning Vs Deep Learning

8 Conclusion

This chapter presented the primary thoughts of AI and deep learning and made sense of the connection between them. In total, the action item is that deep learning is a subfield of AI, and man-made brainpower and AI are not equivalent words. Besides, this part presented the three general classifications of AI, that is, working with named information (supervised learning) and unlabeled information (solo endlessly learning complex cycles (unsupervised learning). Eventually, there are a ton of areas of explicit language encompassing profound learning. A ton of specialized terms are from different fields; however, I know that recognizable terms might allude to something else with regards to profound learning.

Chapter 2

Natural Language Processing

1 Introduction

Natural language processing (NLP), often known as computational linguistics, is one of the most essential information-age technologies. Because humans communicate practically everything in language. Deep learning (or neural network) techniques have achieved very high performance across a wide range of NLP tasks in the recent decade, employing single end-to-end neural models that do not need task-specific feature engineering.

In this chapter we will talk about the different approach, definition of NLP, their fields of applications, components of NLP and the basics of NLP for Text.

2 Natural language processing

Natural Language Processing **(**NLP) is a subfield of computer science and artificial intelligence concerned with interactions between computers and human (natural) languages. It is the technology that is used by machines to understand, analyze, manipulate, and interpret human's languages [31].

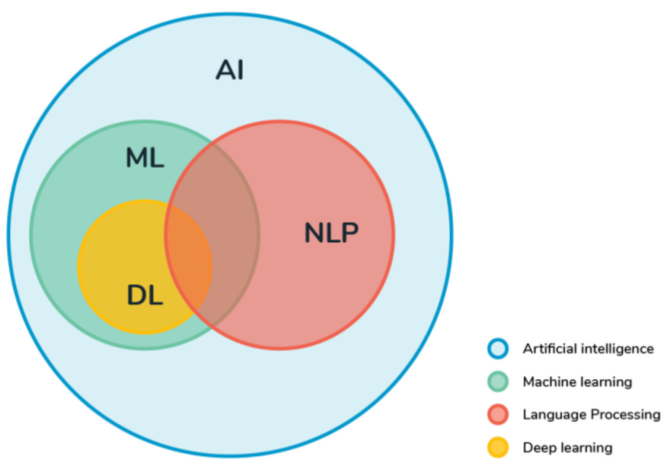

Figure 0.1 Natural Language Processing[32].

3 Difference between Natural language and Computer Language

Natural Language	**Computer Language**
Has a very large vocabulary.	Has a very limited vocabulary.
It’s easily understood by humans.	It’s easily understood by machines.
It’s ambiguous in nature.	It’s unambiguous.

Table 0.1 Difference between Natural language and Computer Language

4 Applications of Natural language processing

NLP helps developers to organize and structure knowledge to perform tasks like translation, summarization, named entity recognition, relationship extraction, speech recognition, topic segmentation, etc.

These technologies help organizations to analyze data, discover insights, automate time-consuming processes and/or gain competitive advantages.

4.1 Machine Translation

Machine translation (MT) is one of the first applications of natural language processing. It is to translate text or speech from one natural language to another natural language. Since each language has grammar rules, the challenge of translating a text is to do so without changing its meaning and style [33].

Examples of Machine Translation tools:

- Google Translate
- Amazon Translate
- Microsoft Translate
- Watson Language Translator
- DeepL Translate

4.2 Speech recognition

Speech recognition is a machine’s ability to identify and interpret phrases and words from spoken language and convert them into a machine-readable format.
Speech recognition systems are an essential part of virtual assistants, like Siri from Apple, Alexa from Amazon, and Google Assistant. And also, we use it with our devices in many daily operations such as searching by voice or transforming some spoken words to written one.

4.3 Sentiment Analysis and opinion mining

Sentiment Analysis and opinion mining. It is used on the web to analyze the attitude, behavior, and emotional state of the sender. This application is implemented through a combination of NLP (Natural Language Processing) and statistics by assigning the values to the text (positive, negative, or natural), and identifying the mood of the context (happy, sad, angry, etc.).
Nowadays sentiment analysis become very important in business, a lot of tools provide a customer feedback analysis such as HubSpot's Service Hub
Repustate, Lexalytics and more. Also, social media depends on this system to study their own users and try to innovate and update their features.

4.4 Text classification

Automatic text classification is another fundamental solution of NLP. It is the process of assigning tags to text according to its content and semantics which allows for rapid, easy retrieval of information in the search phase. This NLP application can differentiate spam from non-spam based on its content.
The most popular use of this tech is in email spam or filtering, email tools such as Gmail, Outlook, provide this feature. And it can automatically tag data by topics like Customer Support, Features, Ease of Use, and Pricing.

4.5 Text Extraction

Text extraction, or information extraction, automatically detects specific information in a text, such as names, companies, places, and more. This is also known as entity recognition. You can also extract keywords within a text, as well as predefined features such as product serial numbers and models.

Applications of text extraction include sifting through incoming support tickets and identifying specific data, like company names, order numbers, and email addresses without needing to open and read every ticket.

We might also want to use text extraction for data entry. we could pull out the information we need and set up a trigger to automatically enter this information in your database.

Keyword extraction, on the other hand, gives us an overview of the content of a text, as this free natural language processing model shows. Combined with sentiment analysis, keyword extraction can add an extra layer of insight, by telling us which words customers used most often to express negativity toward our product or service.

5 Components of NLP

Natural Language Processing can be divided into two major components, understanding and generation.

5.1 Natural Language Understanding

Natural Language Understanding (NLU) helps the machine to understand and analyze human language by extracting the metadata from content such as concepts, entities, keywords, emotion, relations, and semantic roles.

NLU is mainly used in Business applications to understand the customer's problem in both spoken and written language.

NLU involves the following tasks:

- It is used to map the given input into useful representation.
- It is used to analyze different aspects of the language.

5.2 Natural Language Generation

Natural Language Generation (NLG) acts as a translator that converts the computerized data into natural language representation. It mainly involves Text planning, Sentence planning, and Text Realization [34].

5.3 Difference between NLU and NLG

Natural Language Understanding	Natural Language Generation
Is the process of reading and interpreting language.	Is the process of writing or generating language.
It produces non-linguistic outputs from natural language inputs.	It produces natural language outputs from non-linguistic inputs.

Table 2.2 Difference between Natural Language Understanding and Natural Language Generation

6 Phases of NLP

Five main phases of Natural Language processing in AI are:

- Morphological and Lexical Analysis
- Syntactic Analysis
- Semantic Analysis
- Discourse Integration
- Pragmatic Analysis

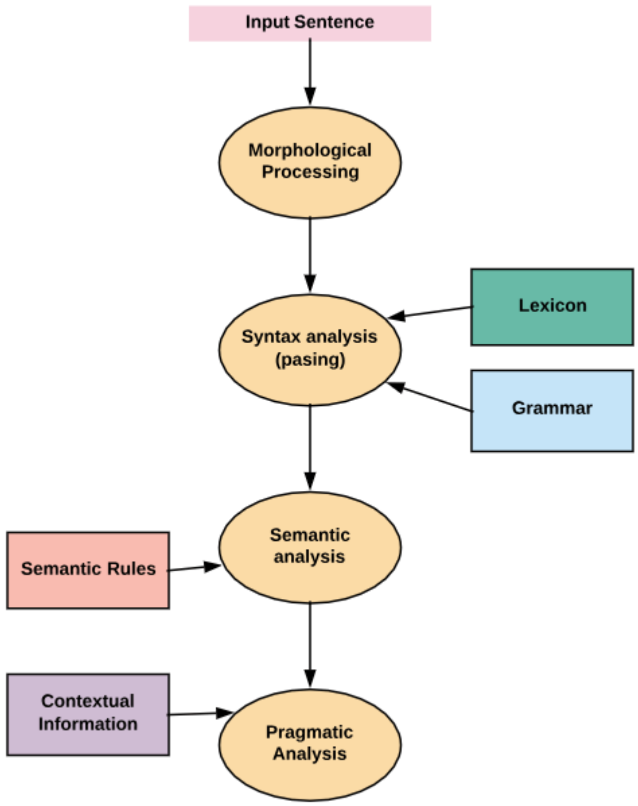

Figure 0.2 Phases of NLP[35].

6.1 Morphological and Lexical Analysis

Lexical analysis is a vocabulary that includes its words and expressions. It depicts analyzing, identifying and describing the structure of words. It includes dividing a text into paragraphs, words and the sentences

Individual words are analyzed into their components, and nonword tokens such as punctuations are separated from the words [36].

6.2 Syntactic Analysis

Semantic Analysis is a structure created by the syntactic analyzer which assigns meanings. This component transfers linear sequences of words into structures. It shows how the words are associated with each other.

Semantics focuses only on the literal meaning of words, phrases, and sentences. This only abstracts the dictionary meaning or the real meaning from the given context. The structures assigned by the syntactic analyzer always have assigned meaning.

6.3 Semantic Analysis

Pragmatic Analysis deals with the overall communicative and social content and its effect on interpretation. It means abstracting or deriving the meaningful use of language in situations. In this analysis, the main focus is always on what was said and reinterpreted on what is meant.
Pragmatic analysis helps users to discover this intended effect by applying a set of rules that characterize cooperative dialogues.

6.4 Discourse Integration

The words are commonly accepted as being the smallest units of syntax. The syntax refers to the principles and rules that govern the sentence structure of any individual language.
Syntax focuses on the proper ordering of words which can affect its meaning. This involves analysis of the words in a sentence by following the grammatical structure of the sentence. The words are transformed into the structure to show how the words are related to each other.

6.5 Pragmatic Analysis

It means a sense of the context. The meaning of any single sentence which depends upon that sentence. It also considers the meaning of the following sentence.

7 The basics of Natural Language Processing for Text

When you begin learning natural language processing, you might feel like it's another language because of all the jargon you'll see. Let's walk through the common terminology

7.1 Terms used in NLP

- **Corpus** refers to the collection of documents or text files. The tweet data from twitter is a corpus.

- A text sample defined as **Documents**. Each tweet is a document.
- Document comprises of **Sentences**. Each tweet has one or more sentences.
- A text gets divided into smaller units called **Tokens**. Each tweet sentence has one or more tokens [37].

7.2 NLP Techniques for Text

The input in natural language processing is text. The data collection for this text happens from a lot of sources. This requires a lot of cleaning and processing before the data can be used for analysis.

These are some of the methods of processing the data in NLP:

- Tokenization
- Normalization
- Stop words
- Regex
- Bag-of-words
- Word Embeddings

7.2.1 Tokenization

Tokenization is breaking the raw text into small chunks. Tokenization breaks the raw text into words, sentences called tokens. These tokens help in understanding the context or developing the model for the NLP. The tokenization helps in interpreting the meaning of the text by analyzing the sequence of the words.

For example, the text “She is playing” can be tokenized into ‘She’, ‘is’, ‘playing’ [38].

Figure 2.3 Tokenization[39].

Tokenization can be done to either separate words or sentences. If the text is split into words using some separation technique it is called word tokenization and the same separation done for sentences is called sentence tokenization.

- **Sentence Tokenization:** Sentence tokenization (also called sentence segmentation) is the problem of dividing a string of written language into its component sentences. The idea here looks very simple. In English and some other languages, we can split apart the sentences whenever we see a punctuation mark.
  However, even in English, this problem is not trivial due to the use of full stop characters for abbreviations. When processing plain text, tables of abbreviations that contain periods can help us to prevent incorrect assignment of sentence boundaries. In many cases, we use libraries to do that job for us, so don’t worry too much about the details for now [40].

- **Word Tokenization:** Word tokenization (also called word segmentation) is the problem of dividing a string of written language into its component words. In English and many other languages using some form of Latin alphabet, space is a good approximation of a word divider.
  However, we still can have problems if we only split by space to achieve the wanted results. Some English compound nouns are variably written and sometimes they contain a space.

In most cases, we use a library to achieve the wanted results, so again don’t worry too much about the details [41].

7.2.2 Normalization

A highly overlooked preprocessing step is text normalization. Text normalization is the process of transforming a text into a canonical (standard) form, by removing inflection from the word. Documents may contain grammars to give meaningful information but these words do not influence text processing. In order to remove it, we use techniques such as Stemming and lemmatization to chop off inflection [42].

- **Stemming:** Stemming is the process of reducing inflection in words (e.g. national, nationality) to their root form (e.g. nation). The “root” in this case may not be a real root word, but just a canonical form of the original word.
  Stemming uses a crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time, and often includes the removal of derivational affixes [43].

- **Lemmatization:** Lemmatization refers to doing things properly with the use of a vocabulary and morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the lemma.

Stemming vs Lemmatization: Both stemming and lemmatization removes inflection from words. However, stemming induces words that are not in the dictionary because it does not use any part of speech tagging, vocabularies and grammar relationship to words whereas lemmatization uses all above mentioned processes step by step and produces desired word.

7.2.3 Stop Words

Stop words are the common words which don't add much meaning to the text. These words need to be removed in the text pre-processing stage, if not it causes noise while processing text.

Stop words usually refer to the most common words such as Conjunctions: ‘for, and, nor, but, or, yet, and so’, or articles ‘a, an, the’. But there is no single universal list of stop words. The list of the stop words can change depending on our application [44].

7.2.4 Regex

A regular expression, regex, or regexp is a sequence of characters that define a search pattern. Regular expressions use the backslash character ('\') to indicate special forms or to allow special characters to be used without invoking their special meaning. We can use regex to apply additional filtering to our text. For example, we can remove all the non-words characters. In many cases, we don’t need the punctuation marks and it’s easy to remove them with regex [45].

7.2.5 Bag of Words

Machine learning algorithms cannot work with raw text directly, we need to convert the text into vectors of numbers. This is called feature extraction.
A bag of words represents the occurrence of words within a document. By implementing Tokenization, ignoring the stop words, applying normalization and applying further pre-processing steps we trail and collect the words. Here, the model concerns whether the word is present in the document and has a unique collection. This approach is a simple and flexible way of extracting features from documents.
To use this model, we need to:

- Design vocabulary of known words (also called Token).
- Choose a measure of the presence of known words.

Any information about the order or structure of words is discarded. That’s why it’s called a bag of words. This model is trying to understand whether a known word occurs in a document, but doesn't know where that word is in the document.
The intuition is that similar documents have similar contents. Also, from the content, we can learn something about the meaning of the document [46].

7.2.6 Word Embeddings

It is an approach for representing words and documents. Word Embedding or Word Vector is a numeric vector input that represents a word in a lower-dimensional space. It allows words with similar meaning to have a similar representation. They can also approximate meaning. A word vector with 50 values can represent 50 unique features.

Word embedding techniques are TF-IDF (Term frequency — Inverse Document Frequency), Word2vec, Glove (Global Vectors).

- **TF-IDF:** TF-IDF, short for term frequency-inverse document frequency, is a statistical measure used to evaluate the importance of a word to a document in a collection or corpus.

  Term frequency is defined as the number of times a word appears in a document divided by the total number of words in the document. Generally, known as Normalization in some sense.

$$TF(\boldsymbol{term}) = \frac{Number\ of\ times\ \boldsymbol{term}\ appears\ in\ a\ document}{Total\ number\ of\ items\ in\ the\ document}$$

  Inverse data frequency defined as the log of the total number of documents divided by the number of documents that contain the word. In below formulae 1 may or may not include it's up to your wish. It is just for standardization.

$$IDF(\boldsymbol{term}) = \log\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ \boldsymbol{term}\ in\ it}\right)$$

  The TF-IDF scoring value increases proportionally to the number of times a word appears in the document, but it is offset by the number of documents in the corpus that contain the word.

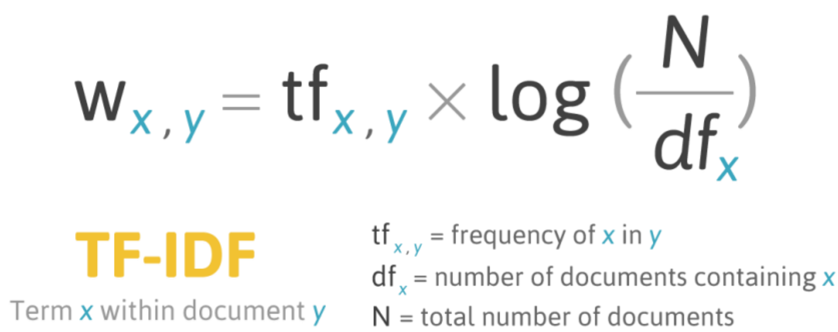

Figure 0.4 TF-IDF[47].

- **Word2Vec:** Word2vec was invented in 2013. It predicts the vectors only by considering the local co-occurrence of the predictive word. The word prediction happens by looking at the range of window size.

  Word2vec incorporated two architectures such as CBOW and Skip-Gram. In CBOW (Continuous bag of words) we try to predict a word by looping many words whereas in Skip-Gram we try to model the contextual words by a given particular word. The below architecture illustrates the same.

  In Word2Vec, the similar words are located together in the vector space and arithmetic operations on word vectors can pose semantic relationships.

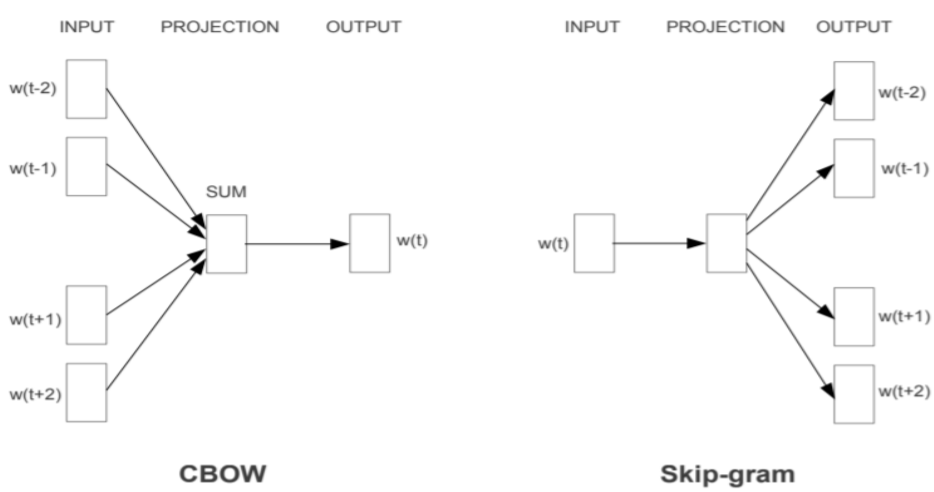

Figure 0.5 Word2Vec[48].

- **GLOVE (Global Vector):** The GloVe method was developed at Stanford by Pennington, et al. Unlike Word2Vec, which creates word embeddings using local context, GloVe focuses on global context to create word embeddings which gives it an edge over Word2Vec. In GloVe, the semantic relationship between the words is obtained using a co-occurrence matrix.

  In this method, we take the corpus and iterate through it and get the co-occurrence of each word with other words in the corpus. We get a co-occurrence matrix through this. The words which occur next to each other get a value of 1, if they are one word apart then 1/2, if two words apart then 1/3 and so on.

  GLOVE and Word2Vec Both are fundamentally the same, they learn vectors from the co-occurrence. However, Word2vec is a predictive model and GloVe is a count-based model. This predictive model learnt the vectors in order to improve the prediction accuracy but count based model overall statistics of the co-occurrence. In Glove, it is easier to train more data than word2vec.

8 NLP APIs

Natural Language Processing APIs allow developers to integrate human-to-machine communications and complete several useful tasks such as speech recognition, chatbots, spelling correction, sentiment analysis, etc.

A list of NLP APIs is given below:

- IBM Watson API
- Chatbot API
- Speech to text API
- Sentiment Analysis API
- Translation API by SYSTRAN
- Text Analysis API by AYLIEN
- Cloud NLP API
- Google Cloud Natural Language API

9 NLP Libraries

- **Natural language Toolkit (NLTK):** NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum [49].
- **Scikit-learn:** It provides a wide range of algorithms for building machine learning models in Python.
- **Pattern**: It is a web mining module for NLP and machine learning.
- **TextBlob:** It provides an easy interface to learn basic NLP tasks like sentiment analysis, noun phrase extraction, or pos-tagging.
- **Quepy:** Quepy is used to transform natural language questions into queries in a database query language.

10 Conclusion

Natural language processing is altering how to analyze and interact with language-based data by teaching computers to make sense of speech and text and perform automated assignments like classification, summarization, translation, extraction, and others.

Chapter 3

Fake News Detection

1 Introduction

The explosive growth of fake news and its erosion to democracy, journalism and economy has increased the demand for fake news detection.

Social media has become a major source of news consumption in the past decade, around 3.8 billion people in the world use social media to consume news on a regular basis [50]. Just like the rise of news consumption on social media has also seen the spread of fake news on social media as well [51]. The problem of fake news is so widespread on social media that governments of countries like the USA, Singapore, and Malaysia have started initiatives to combat fake news [52]. According to the World Economic Forum (WEF), Davos the spread of fake news and misinformation online is one of the top ten perils of society today [53].

Fake news on social media is being used by individuals, organizations, political parties, and government as a modern age propaganda tool to propagate their idea, ideology, and agenda for ideological, financial, and political gains [54]. The problem of fake news on social media came under the limelight and people started to take it seriously right after the 2016, US Presidential election. Many reports in 2016 suggested that fake news played a major role in the US Presidential election of 2016. The problem of fake news was not limited to the 2016 US presidential only. Many news reports across different countries also suggest that fake news on social media has deadly consequences [55].

2 Fake News Characterization

In this section, we introduce the basic social and psychological propositions related to fake news and discuss more advanced patterns introduced by social media. Specifically, we first discuss colorful definitions of fake news and separate affiliated concepts that are generally misknew as fake news. We also describe different aspects of fake news on traditional media and the new patterns found on social media.

2.1 Definitions of Fake News

Fake news is the deliberate spread of misinformation via traditional news media or via social media. False information spreads extraordinarily fast. This is demonstrated by the fact that, when one fake news site is taken down, another will promptly take its place. In addition, fake news can become indistinguishable from accurate reporting since it spreads so fast. People can download articles from sites, share the information, re-share from others and by the end of the day the false information has gone so far from its original site that it becomes indistinguishable from real news [56].

2.2 Fake News on Traditional News Media

Fake news itself is not a new problem. The media ecology of fake news has changed over time, from newsprint to radio/TV and more recently to online news and social media. We refer to “traditional fake news” as the fake news problem before social media had a major impact on its production and distribution. In the following we describe various basics from psychology and the social sciences that describe the effect of fake news both on an individual level and on the level of the societal information ecosystem [57].

2.2.1 Psychological Foundations of Fake News

Psychologists and cognitive scientists have been studying for many years what’s called the “misinformation effect:” the way false or misleading information, received by subjects after they’ve received correct information, can distort their understanding. A better grasp of this and other related psychological phenomena, such as the “tainted truth effect” can help policymakers and the public better devise and implement solutions to the fake news problem [58].

Psychology studies shows that correction of false information (e.g., fake news) by the presentation of true, factual information is not only unhelpful to reduce misperceptions, but sometimes may even increase the misperceptions, especially among ideological groups [59].

Brendan Nyhan and Jason Reifler. When corrections fail: The persistence of political misperceptions. Political Behavior.

2.2.2 Social Foundations of the Fake News Ecosystem

Considering the entire news consumption ecosystem, we can also describe some of the social dynamics that contribute to the proliferation of fake news. Prospect theory describes decision making as a process by which people make choices based on the relative gains and losses as compared to their current state [60,61]. This desire for maximizing the reward of a decision applies to social gains as well, for instance, continued acceptance by others in a user's immediate social network. As described by social identity theory [62; 63] and normative influence theory [64; 65], this preference for social acceptance and affirmation is essential to a person's identity and self-esteem, making users likely to choose socially safe" options when consuming and disseminating news information, following the norms established in the community even if the news being shared is fake news.

2.3 Fake News on social media

The term fake news was not so popular a few years ago, it became a topic of discussion and debate in 2016 when traditional media houses and researchers reported that fake news shared and distributed on social media in the United States in 2016 most likely has helped determine the outcome of the 2016 US presidential election by influencing the voting decision of the American citizen [66].

2.3.1 Malicious Accounts on social media for Propaganda

While many users on social media are legitimate, social media users may also be malicious, and in some cases are not even real humans. The low cost of creating social media accounts also encourages malicious user accounts, such as social bots, cyborg users, and trolls. A social bot refers to a social media account that is controlled by a computer algorithm to automatically produce content and interact with humans (or other bot users) on social media [67]. Social bots can become malicious entities designed specifically with the purpose to do harm, such as manipulating and spreading fake news on social media. Studies show that social bots distorted the 2016 U.S. presidential election online discussions on a large scale [68] and that around 19 million bot accounts tweeted in support of either Trump or Clinton in the week leading up to election day. Trolls, real human users who aim to disrupt online communities and provoke consumers into an emotional response, are also playing an important role in spreading fake news on social media. For example, evidence suggests that there were 1,000 paid Russian trolls spreading fake news on Hillary Clinton. Trolling behaviors are highly affected by people's moods and the context of online discussions, which enables the easy dissemination of fake news among otherwise “normal” online communities [69]. The effect of trolling is to trigger people's inner negative emotions, such as anger and fear, resulting in doubt, distrust, and irrational behavior. Finally, cyborg users can spread fake news in a way that blends automated activities with human input. Usually, cyborg accounts are registered by humans as camouflage and set automated programs to perform activities on social media. The easy switch of functionalities between human and bot offers cyborg users unique opportunities to spread fake news [70]. In a nutshell, these highly active and partisan malicious accounts on social media become the power sources and proliferation of fake news.

2.3.2 Echo Chamber Effect

Social media provides a new paradigm of information creation and consumption for users. The information-seeking and consumption process is changing from a mediated form (e.g., by journalists) to a more disinter-mediated way [71]. Consumers are selectively exposed to certain kinds of news because of the way news feeds appear on their homepage on social media, amplifying the psychological challenges to dispelling fake news identified above. For example, users on Facebook always follow like-minded people and thus receive news that promotes their favored existing narratives [72]. Therefore, users on social media tend to form groups containing like-minded people where they then polarize their opinions, resulting in an echo chamber effect. The echo chamber effect facilitates the process by which people consume and believe fake news due to the following psychological factors [73]: (1) social credibility, which means people are more likely to perceive a source as credible if others perceive the source is credible, especially when there is not enough information available to access the truthfulness of the source; and (2) frequency heuristic, which means that consumers may naturally favor information they hear frequently, even if it is fake news. Studies have shown that increased exposure to an idea is enough to generate a positive opinion of it [74; 75], and in echo chambers, users continue to share and consume the same information. As a result, this echo chamber effect creates segmented, homogeneous communities with a very limited information ecosystem. Research shows that homogeneous communities become the primary driver of information diffusion that further strengthens polarization [76].

2.4 Examples of well-known fake news

In order to better understand what fake news is, the motivations behind its generation and online circulation, as well as the impact it might have on the audiences, it is helpful to look at a few well-known examples of fake news. As Allcott and Gentzkow point out in their research about the spread of fake news during the US presidential elections. One historical example of fake news is ‘Great Moon Hoax’ of 1835, in which the New York Sun published a series of articles about the discovery of life on the moon. The discovery was falsely attributed by the newspaper to Sir John Herschel, one of the most famous astronomers of that time. The newspaper’s circulation increased dramatically due to the fake story and after a while, it was discovered that the story was nothing more than a hoax. A well-known example of fake news distributed by users who managed to reach a huge audience and earn a lot of money by generating misleading stories is the ‘Veles case'. According to Subramanian (2017), this case refers to a small Balkan town where more than 100 sites posting fake news about the US election were run by teenagers. These websites generated tons of misleading information about the election and the owner’s earned money from the advertisements they had added to their websites. During the US election campaigns these websites were pumping out sensational – and fake – stories to earn money from advertising. Another recent example of a fake news source is the endingthefed.com website, which was responsible for four of the ten most popular fake news stories on Facebook about the US election and was run by a 24-year-old man [77]. A US company called Disinformation owns many fake news sites, including NationalReport.net, USAToday.com.co, and WashingtonPost.com.com and its owner claims to employ between 20 and 25 writers [78].

Another example in 2022 of fake news is the case of the football game of Algeria vs Cameroon, all of us saw how that everyone has something to say, some of them saying that the match will be replayed, and others saying that officially Algeria excluded and not going to play the World Cup Qatar 2022, and others say if Algeria has the proof that the referee Gassama got a bribe by the Cameroonian federation of Football, Algeria will directly qualify to the World Cup of Qatar.

But none of them are true, only the FIFA has the correct information about it, and whenever the FIFA publishes a report the voices get louder again, and everyone understands it as he wants.

3 Fake News Detection

In the previous section, we introduced the conceptual characterization of traditional fake news and fake news in social media. Based on this characterization, we further explore the problem definition and proposed approaches for fake news detection.

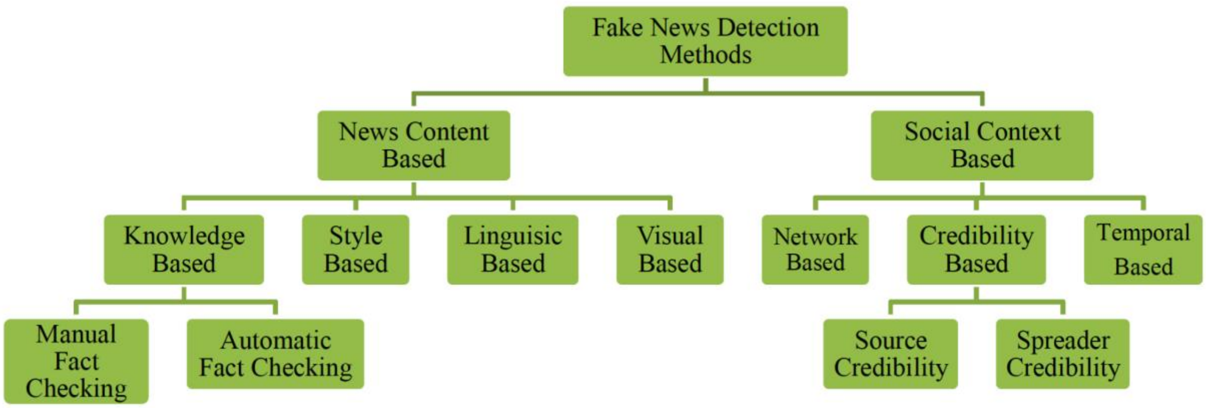

Figure 0.1 Fake News Detection Methods[79]

3.1 Fake News Detection Methods

The generation and propagation of fake news results in detrimental societal consequences. Now more than ever before, it is important to intensify research efforts that will build tools that detect fake news automatically and effectively.

Here we're going to discuss various state-of-the-art studies on fake news detection under the broader umbrella of content and social context of the news article.

3.1.1 Content Based

The content-based fake news detection method aims to detect fake news by analyzing the content [80] of the article, i.e., either the text or image or both within the news article. For automatically detecting the fake news, the researchers often relying on either latent or hand-crafted features of the content.

A list of representative news content attributes is listed below:

- Source: Author or publisher of the news article.
- Headline: Short title text that aims to catch the attention of readers and describes the main topic of the article.
- Body Text: Main text that elaborates the details of the news story; there is usually a major claim that is specifically highlighted and that shapes the angle of the publisher.
- Image/Video: Part of the body content of a news article that provides visual cues to frame the story.

- **Knowledge Based**

The most straightforward way to detect fake news is to check the truthfulness of the statements claimed in news content. Knowledge based approaches are also known as fact checking. The expert-oriented approaches, such as Snopes [81], mainly rely on human experts working in specific fields to help decision making. The crowdsourcing-oriented approaches, such as Fiskkit [82] where normal people can annotate the accuracy of news content, utilize the wisdom of crowd to help check the accuracy of the news articles. The computational-oriented approaches can automatically check whether the given claims have reachable paths or could be inferred in existing knowledge graphs.

- **Style Based**

Style-based approaches attempt to capture the writing style of news content. Mykhailo Granik et al. [83] find that there are some similarities between fake news and spam email, such as they often have a lot of grammatical mistakes, try to affect reader's opinion on some topics in manipulative way and use similar limited set of words. So, they apply a simple approach for fake news detection using naive Bayes classifier due to those similarity. Shlok Gilda [84] applies term frequency-inverse document frequency (TF-IDF) of bi-grams and probabilistic context free grammar (PCFG) detection and test the dataset on multiple classification algorithms. William Yang Wange [85] investigates automatic fake news detection based on surface-level linguistic patterns and design a novel, hybrid convolutional neural network to integrate speaker related metadata with text. Jiang et al. find that some key words tend to appear frequently in the micro-blog rumor. They analyze the text syntactical structure features and presents a simple way of rumor detection based on Language Tool.

- **Linguistic Based**

Since fake news pieces are intentionally created for financial or political gain rather than to report objective claims, they often contain opinionated and inflammatory language, crafted as “clickbait” or to incite confusion [86]. Thus, it is reasonable to exploit linguistic features that capture the different writing styles and sensational headlines to detect fake news. Linguistic based features are extracted from the text content in terms of document organizations from different levels, such as characters, words, sentences, and documents. In order to capture the different aspects of fake news and real news, existing work utilized both common linguistic features and domain-specific linguistic features. Common linguistic features are often used to represent documents for various tasks in natural language processing. Typical common linguistic features are: (i) lexical features, including character level and word-level features, such as total words, characters per word, frequency of large words, and unique words; (ii) syntactic features, including sentence-level features, such as frequency of function words and phrases (i.e., “n-grams” and bag-of-words approaches [87]) or punctuation and parts of-speech (POS) tagging. Domain-specific linguistic features, which are specifically aligned to news domain, such as quoted words, external links, number of graphs, and the average length of graphs, etc. [88] Moreover, other features can be specifically designed to capture the deceptive cues in writing styles to differentiate fake news, such as lying detection features [89]. Visual-based: Visual cues have been shown to be an important manipulator for fake news propaganda. As we have characterized, fake news exploits the individual vulnerabilities of people and thus often relies on sensational or even fake images to provoke anger or other emotional response of consumers.

- **Visual Based**

Visual content is often viewed as evidence that can increase the credibility of the news article [90] and hence the fake news publishers tend to utilize provocative visual content to attract and mislead readers. In [91] various visual and statistical image features are extracted for news authentication.

Verifying Multimedia Use task [92] under the MediaEval-16 benchmark addresses the problem of detecting digitally manipulated (tampered) images.

3.1.2 Social Context Based

The nature of social media provides researchers with additional resources to supplement and enhance News Content Models. Social context models include relevant user social engagements in the analysis, capturing this auxiliary information from a variety of perspectives.
Recent studies have examined different context-based approaches for fake news detection.

- **Network Based**

Network-based fake news detection studies different social networks like friendship, tweet-retweet, post-repost networks to detect fake news. It detects who spreads the fake news, relationships among the spreaders and how fake news propagates on social networks.
Users form different networks on social media in terms of interests, topics, and relations. As mentioned before, fake news dissemination processes tend to form an echo chamber cycle, highlighting the value of extracting network-based features to represent these types of network patterns for fake news detection. Network-based features are extracted via constructing specific networks among the users who published related social media posts. Different types of networks can be constructed.

- **Temporal Based**

Studies have shown that news stories on the Internet are not static but are constantly evolved over time by adding new information or twisting the actual claim. This is very much evident in cases where the rumors resurge multiple times after the original news article is posted. The lifecycle analysis of rumor helps in understanding this phenomenon and [93] examines the recurring rumors at the message level across different time periods. [94] provides deep understanding into the diffusion patterns of rumors over time.

- **Credibility Based**

The basic assumption is that the credibility of a news event is highly related to the credibility of relevant social media posts. Both homogeneous and heterogeneous credibility networks can be built for propagation process. Homogeneous credibility networks consist of a single type of entities, such as post or event [95]. Heterogeneous credibility networks involve different types of entities, such as posts, sub-events, and events [96; 97].

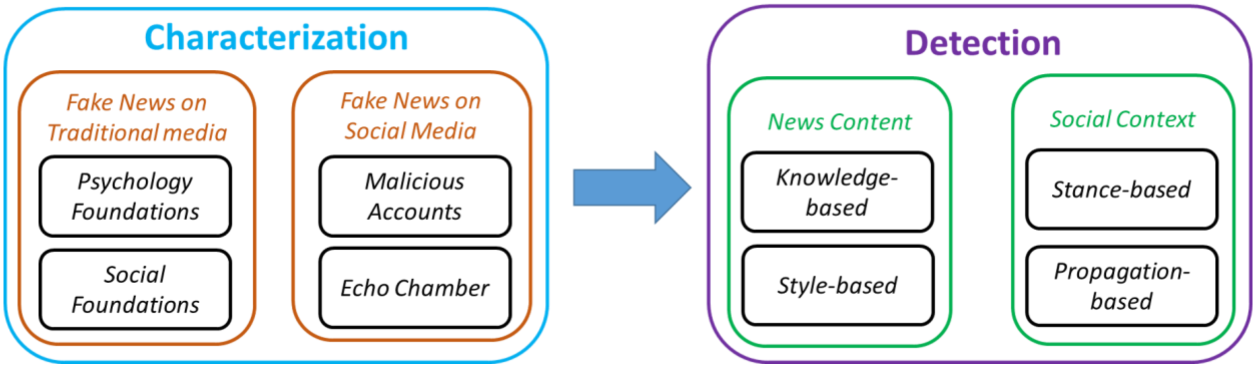

Figure 0.2 Fake news on social media: from characterization to detection [98].

4 Fake News Detection Examples

In the last years, many fake news detection platforms based on the Ai has launched, thanks to smart people who help detect and combat fake news. We can name few of them.

4.1 Factmata

This London-based AI startup was founded by NLP researchers Dhruv Ghulati, the startup does automated content scoring and verification, specifically targeting the dissemination of fake and misleading news. It is now working on building an AI-based platform for fact-checking and news aggregation.

The process of fact-checking is to be powered by people with interest and concern towards fake news, and in time is expected to be completely automated and powered by AI. In the initial model, users will serve as watchdogs and flag content that is false.

4.2 Full Fact

Full Fact is a media company founded in 2009. It offers several fact-checking tools, including ones that are automated through the use of artificial intelligence. It is building AI tools to help fact checkers understand what is the most important, and check-worthy, information of the day. It also aims to design an algorithm that can identify when somebody knowingly repeats something they know to be false.

4.3 Misbar

Misbar is the most popular Arabic platform for detecting fake news, was launched to verify news and detect lies in the public space, without being satisfied with this task, as it undertakes to spread awareness among people, enhance their critical view and train them to verify the news themselves.

5 Conclusion

In conclusion, fake news on social media is becoming a major issue and the consequences of it are borne by the social media users. The study shows and suggests that a large number of social media users are being affected by fake news and a large number of social media users do not verify the news they read online and the main reason behind this is the lack of information about news Verification and fact-checking sources.

With the increasing popularity of social media, more and more people consume news from social media instead of traditional news media. However, social media has also been used to spread fake news, which has strong negative impacts on individual users and broader society. In this chapter, we explored the fake news problem by reviewing existing literature in two phases: characterization and detection. In the characterization phase, we introduced the basic concepts and principles of fake news in both traditional media and social media. In the detection phase, we reviewed existing fake news detection approaches from a data mining perspective, including feature extraction and model construction.

Chapter 4

The realized work and obtained results

1 Introduction

This chapter presents the experimental part, so it contains the preprocessing of the used dataset including cleaning the data and applying the NLP techniques, also we are going to propose a model using a machine learning algorithm. After that, we make a predictive system.

2 Related Work

Here are some works related to Fake news detection

2.1 Fake news detection within online social media using supervised artificial intelligence

In this paper, a two-step method for identifying fake news on social media has been proposed, focusing on fake news. In the first step of the method, a number of pre-processing is applied to the data set to convert unstructured structured data sets into the structured data set. The texts in the data set containing the news are represented by vectors using the obtained TF weighting method and Document-Term Matrix. In the second step, twenty-three supervised artificial intelligence algorithms have been implemented in the data set transformed into the structured format with the text mining methods. In this work, an experimental evaluation of the twenty-three intelligent classification methods has been performed within existing public data sets and these classification models have been compared depending on four evaluation metrics.

In this paper, three different data sets have been used to evaluate intelligent classification algorithms. As declared in previous sections, twenty-three different supervised artificial intelligence algorithms; BayesNet, JRip, OneR, Decision Stump, ZeroR, Stochastic Gradient Descent (SGD), CV Parameter Selection (CVPS), Randomizable Filtered Classifier (RFC), Logistic Model Tree (LMT), Locally Weighted Learning (LWL), Classification Via Clustering (CvC), Weighted Instances Handler Wrapper (WIHW), Ridor, Multi-Layer Perceptron (MLP), Ordinal Learning Model (OLM), Simple Cart, Attribute Selected Classifier (ASC), J48, Sequential Minimal Optimization (SMO), Bagging, Decision Tree, IBk, and Kernel Logistic Regression (KLR) have been adapted to detect fake news. [99]

2.2 Fake news detection on social networks using Machine learning techniques

The constant circulation of fake news directly or indirectly produces a huge negative impact on vast majority of the society. Users throughout the world are actively visiting well-known social networking sites such as Facebook, Twitter, Instagram, LinkedIn, and others. ML techniques investigated and compared with three different evaluation models namely Countvectorizer, TF-IDF Vectorizer and N-gram and four machine learning techniques were used namely Naive Bayes, SVM, Random Forest, Logistic Regression. The proposed strategy achieves the maximum accuracy % when using TF-IDF highlights and an SVM classifier. The highest level of precision is 93%. [100]

2.3 Fake News Detection in social media using Graph Neural Networks and NLP Techniques

The paper presents our solutions for the MediaEval 2020 task namely Fake News: Corona Virus and 5G Conspiracy Multimedia Twitter-Data-Based Analysis. The task aims to analyze tweets related to COVID-19 and 5G conspiracy theories to detect misinformation spreaders. The task is composed of two sub-tasks namely (i) text-based, and (ii) structure-based fake news detection. For the first task, we propose six different solutions relying on Bag of Words (BoW) and BERT embedding. Three of the methods aim at binary classification tasks by differentiating in 5G conspiracy and the rest of the COVID-19 related tweets while the rest of them treat the task as ternary classification problem. In the ternary classification task, our BoW and BERT based methods obtained an F1-score of .606% and .566% on the development set, respectively. On the binary classification, the BoW and BERT based solutions obtained an average F1-score of .666% and .693%, respectively. On the other hand, for structure-based fake news detection, we rely on Graph Neural Networks (GNNs) achieving an average ROC of .95% on the development set. [101]

2.4 A comparative analysis of Graph Neural Networks and commonly used machine learning algorithms on fake news detection.

in this paper, we present a comparative analysis among some commonly used machine learning algorithms and Graph Neural Networks for detecting the spread of false news on social media platforms. In this study, we take the UPFD dataset and implement several existing machine learning algorithms on text data only. Besides this, we create different GNN layers for fusing graph-structured news propagation data and the text data as the node feature in our GNN models. GNNs provide the best solutions to the dilemma of identifying false news in our research. Index Terms—Fake news detection, Graph Neural Network, Text classification, social media analysis, GNN.

In the Gossipcop dataset, GraphSAGE has a higher accuracy of 96.99 % for the 768-dimensional Bert technique and 96.52% for the 300-dimensional spacy technique, whereas GAT has a higher accuracy of 93.27% for the 10-dimensional profile technique. On the other hand, In the PolitiFact dataset, GIN performs better accuracy of 85.07% for the 768-dimensional Bert technique while the 10- dimensional profile technique outperforms the GraphSAGE model by 78.28% and the 300-dimensional spacy technique outperforms the GCN model by 82.81%. It is noticeable that for both of the datasets the endogenous techniques (Bert and Spacy) typically outperform the profile feature, which only holds user profile data. [102]

2.5 Unified Fake News Detection using Transfer Learning of BERT Model

This paper attempts to develop a unified model by combining publicly available datasets to detect fake news samples effectively. Our experiments are conducted on three publicly available datasets, namely ISOT and others from the Kaggle website. The model is designed with the help of Google's Bidirectional Encoder Representation from Transformers (BERT) base uncased model using a transfer learning approach by using pre-trained weights without changing them during training with preprocessing steps like removing the words whose lengths are less than three. To develop the final model, we fine-tune the pre-trained Google BERT base uncased model on each dataset and select the model with better performance on all three datasets. Our final model is constructed from the hyperparameters obtained from the individual models, which show better performance and are trained on the combined dataset. The results indicate that our proposed finalized model works better than existing machine learning and deep learning models like Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM), etc., with an F1-score of 0.97 and accuracy of 97% when combined with all three datasets. [103]

3 The Used Programing Languages, Libraries and Tools

Here are the programming language and the libraries we used in our work

3.1 Programing Languages

We decided to use Python programming Language due to it popularity in the Artificial Intelligence field and to the huge number of Libraries, also due to the big community that Python has.

- **Python** is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation.

3.2 Libraries

- **Numpy:** is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
- **Pandas:** is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three clause BSD license.
- **SKLearn:** Scikit-learn is probably the most useful library for machine learning in Python. The sklearn library contains a lot of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction.
- **Re:** Regular Expression specifies a set of strings that matches it; the functions in this module let you check if a particular string matches a given regular expression (or if a given regular expression matches a particular string, which comes down to the same thing).
- **NLTK:** Natural Language Toolkit, is a platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for

classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

3.3 Tools

In this work we relied entirely on a very useful cloud technology provided by google which is Google Colaboratory.

Google Colaboratory (also known as Colab) is a free Jupyter notebook environment and a web IDE for python that runs in the cloud without any required, it allows us to write, run and share code within google drive.

Colab provides a lot of benefits such as:

- Free virtual machines for you to use: with about 12GB RAM and 50GB hard drive space, with common dependencies such as numpy, pandas, and even TensorFlow pre-installed.
- Free GPU access
- Supports Python3
- There is integration with Google Drive, you can share and control permissions and you'll be able to see other collaborators work instantly.
- There is a revision history — an extremely useful feature for teams.
- You can add comments on cells — someone for example could be given 'comment-only' permissions to review code. You can also resolve, reply and target comments to others.
- You can import existing Jupyter/Python notebooks.
- It also supports connecting to a Jupyter runtime on your local machine.

4 Our Approach

The framework of the proposed method as shown in Fig4.1 consists of six main phases namely; Data preparation, Text Pre-processing, Train the model, Evaluation, and Creating a predictive system.

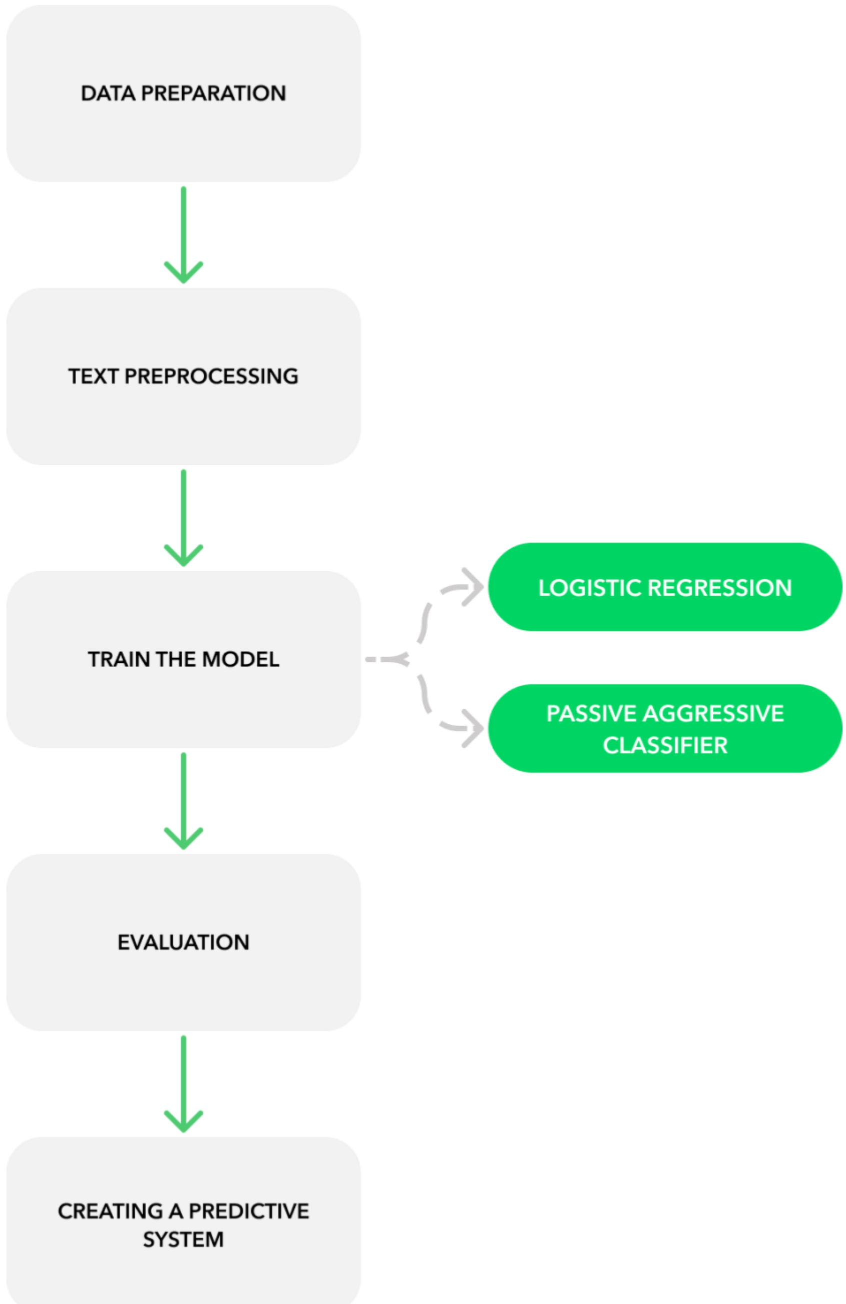

Figure 0.1 Our approach

4.1 Importing dependencies

Here we imported the required dependencies for our experiment, we imported the python libraries (numpy and pandas), also Sklearn to split the train data from the test data, apply the logistic regression and measure the accuracy score. NLTK to apply NLP techniques.

```python
import numpy as np
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from nltk.stem.porter import PorterStemmer
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
```

Figure 0.2 Importing dependencies

4.2 Preprocessing the Dataset

Data preparation is the process of cleaning and transforming raw data prior to processing and analysis

4.2.1 Printing the stop words

After we imported the stop words from NLTK, the line below is for printing them.

```
# printing the stopwords in English

nltk.download('stopwords')
print(stopwords.words('english'))
```

Figure 4.3 printing the stop words in English

The list of the stop words

```
# the stopwords in English

['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're",
"you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he',
'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's",
'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which',
'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are',
'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do',
'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because',
'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against',
'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to',
'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again',
'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all',
'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no',
'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can',
'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o',
're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't",
'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn',
"isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't",
'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't",
'won', "won't", 'wouldn', "wouldn't"]
```

Figure 0.4 Stop words in English

4.2.2 Loading the Dataset

This data set was collected from e-newspapers like New York Times, Breitbart, News Tral, CNN…based on tweets from trusted reporters and official authorities from Kaggle. And the line of code below is for loading the dataset to a pandas Data Frame.

Loading the Dataset

```
# loading the dataset to a pandas DataFrame
news_dataset = pd.read_csv('/content/train.csv')
```

Figure 0.5 Loading the dataset

The line of code below to see the dimension of the table of the dataset, the output is (20800,5), which means 5 columns (id, title, author, text, label) and 20800 rows.

Loading the dataset

```
news_dataset.shape
```

Figure 0.6 Dimension of the table of the dataset

The instruction below shows the first 5 rows (by default) of our dataset using the function “head()”.

```
Print the head of the dataset

# print the first 5 rows of the dataframe
news_dataset.head()
```

Figure 0.7 Print the first 5 rows of the data frame

The first 5 rows of our dataset

	id	title	author	text	label
0	0	House Dem Aide: We Didn't Even See Comey's Let...	Darrell Lucus	House Dem Aide: We Didn't Even See Comey's Let...	1
1	1	FLYNN: Hillary Clinton, Big Woman on Campus - ...	Daniel J. Flynn	Ever get the feeling your life circles the rou...	0
2	2	Why the Truth Might Get You Fired	Consortiumnews.com	Why the Truth Might Get You Fired October 29, ...	1
3	3	15 Civilians Killed In Single US Airstrike Hav...	Jessica Purkiss	Videos 15 Civilians Killed In Single US Airstr...	1
4	4	Iranian woman jailed for fictional unpublished...	Howard Portnoy	Print \nAn Iranian woman has been sentenced to...	1

Figure 0.8 The result of printing the first 5 rows of the data frame

4.2.3 Cleaning Data

It is a part of data preprocessing practice, usually the bad data comes from the human mistakes, in order to create an efficient model, we need to have clean data.

Here's the process we did to clean our data

- Counting the number of missing values using the function "isnull()" to detect the missing value, and the function "sum()" to return the total number.

Cleaning the Data

```python
# counting the number of missing values in the dataset
news_dataset.isnull().sum()
```

Figure 0.9 Cleaning the data

This is the result of the instruction above

```
id           0
title      558
author    1957
text        39
label        0
dtype: int64
```

- Replacing the missing values with empty string.

Cleaning the Data

```python
# replacing the null values with empty string
news_dataset = news_dataset.fillna('')
```

Figure 0.10 Replacing the null values with empty string

4.2.4 Making the Content

Next, we add a new column “content” which is a combination of the “author” and “title”. Th reason why we combined them is we’re going to use them for the training to reduce the time of training the model instead of working on the “text”.

Composing the Content

```
# merging the author name and news title
news_dataset['content'] = news_dataset['author']+'
'+news_dataset['title']
```

Figure 0.11 Composing the content

The instruction below prints the new column “content”.

Printing the Content

```
print(news_dataset['content'])
```

Figure 0.12 Printing the content

The result of the content

the stopwords in English

```
0        darrel lucu hous dem aid even see comey letter...
1        daniel j flynn flynn hillari clinton big woman...
2                   consortiumnew com truth might get fire
3        jessica purkiss civilian kill singl us airstri...
4        howard portnoy iranian woman jail fiction unpu...
                               ...
20795    jerom hudson rapper trump poster child white ...
20796    benjamin hoffman n f l playoff schedul matchup...
20797    michael j de la merc rachel abram maci said re...
20798    alex ansari nato russia hold parallel exercis ...
20799                         david swanson keep f aliv
Name: content, Length: 20800, dtype: object
```

Figure 0.13 The result of the content

4.3 Text pre-processing

After the phase of cleaning the data now we move to the text preprocessing phase by applying the NLP techniques.

4.3.1 Stemming the Dataset

In this instruction, we call the “PorterStemmer()” in variable “port_stem”.

Stemming

```
port_stem = PorterStemmer()
```

Figure 0.14 Stemming

We define a function called “stemming” which has these instructions below:

- Exclude everything that’s not presenting the alphabets and replace it by space using the “sub()” function.
- Convert all the alphabets to lowercase letters by applying the “lower()” function.
- “split()” function for splitting the content as a list of strings separated by whitespaces.
- Take each word from the “content” out of the stop words and apply the “stem()” function to extract the root word.
- Before we return the stemmed content, we convert the array of each row to a sequenced word using “join()”

Creating the function of Stemming

```
def stemming(content):
    stemmed_content = re.sub('[^a-zA-Z]',' ',content)
    stemmed_content = stemmed_content.lower()
    stemmed_content = stemmed_content.split()
    stemmed_content = [port_stem.stem(word) for word in stemmed_content if not
word in stopwords.words('english')]
    stemmed_content = ' '.join(stemmed_content)
    return stemmed_content
```

Figure 0.15 Creating the function of stemming

We apply the "stemming()" function on the content column from the dataset, and print the result. I took a while to apply that function on our dataset.

```
news_dataset['content'] = news_dataset['content'].apply(stemming)
print(news_dataset['content'])
```

Figure 0.16 Applying the Stemming function on the content

The result of printing the new column “Content”

```
darrel lucu hous dem aid even see comey letter...
1        daniel j flynn flynn hillari clinton big woman...
2                    consortiumnew com truth might get fire
3        jessica purkiss civilian kill singl us airstri...
4        howard portnoy iranian woman jail fiction unpu...
                                ...
20795    jerom hudson rapper trump poster child white s...
20796    benjamin hoffman n f l playoff schedul matchup...
20797    michael j de la merc rachel abram maci said re...
20798    alex ansari nato russia hold parallel exercis ...
20799                        david swanson keep f aliv
Name: content, Length: 20800, dtype: object
```

4.3.2 Labeling the data

In this step, we replace the variable “X” by “content” values, and keep the “Y” value.

In this step, we remove the “label” column from the dataset and put it in variable “X”, and the variable “Y” contains just the “label” column, we print the result of “X” and “Y”.

Separating the data and label

```
#separating the data and label
X = news_dataset['content'].values
Y = news_dataset['label'].values
```

Figure 0.17 separating the data and label

Print X:

```
['Darrell Lucus House Dem Aide: We Didn’t Even See Comey’s Letter Until
Jason Chaffetz Tweeted It'
 'Daniel J. Flynn FLYNN: Hillary Clinton, Big Woman on Campus - Breitbart'
 'Consortiumnews.com Why the Truth Might Get You Fired' ...
```

```
 'Michael J. de la Merced and Rachel Abrams Macy’s Is Said to Receive
Takeover Approach by Hudson’s Bay - The New York Times'
 'Alex Ansary NATO, Russia To Hold Parallel Exercises In Balkans'
 'David Swanson What Keeps the F-35 Alive']
```

Print Y:

```
[1 0 1 ... 0 1 1]
```

4.3.3 Vectorization of Data

TF-IDF stands for “Term Frequency – Inverse Document Frequency”

- TF-IDF is a numerical statistic which measures the importance of the word in a document.
- Term Frequency: Number of times a word appears in a text document.
- Inverse Document Frequency: Measure the word is a rare word or common word in a document.

```
# converting the textual data to numerical data
vectorizer = TfidfVectorizer()
vectorizer.fit(X)

X = vectorizer.transform(X)
print(X)
```

Figure 0.18 TF-IDF Vectorizer

The new X

```
  (0, 23355)	0.18006497451107856
  (0, 22649)	0.26575278886038384
  (0, 22289)	0.3484071341454308
  (0, 19171)	0.22537992364975484
  (0, 12902)	0.3024224900242886
```

```
(0, 12528)      0.24883399099107747
(0, 11409)      0.20615188166061463
(0, 11307)      0.1532265401605094
(0, 10387)      0.1844880289323935
:   :
(20798, 1324)   0.2955941555358824
(20798, 1009)   0.2706299600743188
(20799, 23493) 0.2683870404159613
(20799, 21564) 0.10106058584391787
(20799, 21101) 0.4480459367054237
(20799, 11815) 0.45575108674851145
(20799, 5537)   0.2993058137514979
(20799, 1043)   0.4480459367054237
(20799, 270)    0.4679442365402834
```

4.4 Training The model

In this phase, we planned to create our training model, using a logistic regression algorithm.

4.4.1 Logistic Regression

Logistic regression is a statistical analysis method to predict a binary outcome, such as yes or no, based on prior observations of a data set. A logistic regression model predicts a dependent data variable by analyzing the relationship between one or more existing independent variables.

- Splitting the dataset to training & test data

Here we split our data to “training” and “test” data, (80% from the data for training, and 20% for the test).

Splitting the dataset to training & test data

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2,
stratify=Y, random_state=2)
```

Figure 0.19 Splitting the dataset to training & test data

Training the Model: Logistic Regression

```
model = LogisticRegression()

model.fit(X_train, Y_train)
```

Figure 0.20 Training the model: Logistic Regression

- **Evaluation**

Here we evaluate the accuracy score of Logistic Regression model

- **Accuracy score of the training data**

Logistic Regression Accuracy Score

```
# accuracy score on the training data
X_train_prediction = model.predict(X_train)
training_data_accuracy = accuracy_score(X_train_prediction, Y_train)
print("Accuracy score of the training data : %0.3f" % training_data_accuracy)
```

Figure 0.21 Logistic regression accuracy score on the training data

The Accuracy score of the training data: **0.987,** which means **98,7%**

- **Accuracy score of the test data**

Logistic Regression Accuracy Score

```
# accuracy score on the test data
X_test_prediction = model.predict(X_test)
test_data_accuracy = accuracy_score(X_test_prediction, Y_test)
print("Accuracy score of the test data : %0.3f" % test_data_accuracy)
```

Figure 0.22 Logistic regression accuracy score on the test data

The Accuracy score of the training data: **0.979,** which means **97,9%**

4.4.2 Passive Aggressive Classifier

Passive Aggressive Classifier is one of the binary Classification Algorithms, Passive-Aggressive algorithms are generally used for large-scale learning. It is one of the few online-learning algorithms.

Fake news detection on social media is a good example to apply it. Because new data is being added every second. To dynamically read data from Twitter continuously, the data would be huge, and using an online-learning algorithm would be ideal.

- Splitting the dataset to training & test data

Here we split our data to “training” and “test” data, (67% from the data for training, and 33% for the test).

```
Passive Aggressive Classifier

from sklearn.linear_model import PassiveAggressiveClassifier
from sklearn import metrics

X2_train, X2_test, Y2_train, Y2_test = train_test_split(X, Y, test_size=0.33,
random_state=42)

linear_clf = PassiveAggressiveClassifier(max_iter=50)
linear_clf.fit(X2_train, Y2_train)
```

Figure 0.23 Training the model: Passive Aggressive Classifier

- **Evaluation**

Here we evaluate the accuracy score of Logistic Regression model

- **Accuracy score of the training data**

```
Passive Aggressive Classifier accuracy score

X_train_prediction2 = linear_clf.predict(X2_train)
training_data_accuracy2 = metrics.accuracy_score(Y2_train, X_train_prediction2)
print("Accuracy score of the training data:   %0.3f" % training_data_accuracy2)
```

Figure 0.24 Passive Aggressive Classifier accuracy score on the training data

The Accuracy score of the training data: **1,0** which means **100%**

- **Accuracy score of the testing data**

```
Passive Aggressive Classifier accuracy score

X_test_prediction2 = linear_clf.predict(X2_test)
testing_data_accuracy2 = metrics.accuracy_score(Y2_test, X_test_prediction2)
print("Accuracy score of the training data:   %0.3f" % testing_data_accuracy2)
```

Figure 0.25 Passive Aggressive Classifier accuracy score on the testing data

The Accuracy score of the training data: **0,99** which means **99%**

4.5 Building a Predictive System

Building a predictive system in order to find that the initial word in the dataset is real or fake.

4.5.1 Logistic Regression model

```
Making a predictive system

X_new = X_test[5]

prediction = model.predict(X_new)
print(prediction)

if (prediction[0]==0):
  print('The news is Real')
else:
  print('The news is Fake')
```

Figure 0.26 Building a Predictive System: Logistic Regression

4.5.2 Passive Aggressive Classifier model

Making a predictive system

```
X2_new = X2_test[0]

prediction = model.predict(X2_new)

print(prediction)

if (prediction[0]==0):
  print('The news is Real')
else:
  print('The news is Fake')
```

Figure 0.27 Building a Predictive System: Passive Aggressive Classifier

4.6 Classification Report:

A Classification report is used to measure the quality of predictions from a classification algorithm.

It is used to show the precision, recall, F1 Score, and support of your trained classification model.

4.6.1 Logistic Regression Model

Classification report

```
from sklearn.metrics import classification_report
print(classification_report(Y_test, X_test_prediction))
```

Figure 0.28 Classification Report: Logistic Regression

	precision	recall	f1-score	support
0	0.99	0.96	0.98	2077
1	0.97	0.99	0.98	2083
accuracy			0.98	4160
macro avg	0.98	0.98	0.98	4160
weighted avg	0.98	0.98	0.98	4160

4.6.2 Passive Aggressive Classifier Model

Classification Report

```
from sklearn.metrics import classification_report
print(classification_report(Y2_test,X_test_prediction2))
```

Figure 0.29 Classification Report: Passive Aggressive Classifier

	precision	recall	f1-score	support
0	0.99	0.99	0.99	3449
1	0.99	0.99	0.99	3415
accuracy			0.99	6864
macro avg	0.99	0.99	0.99	6864
weighted avg	0.99	0.99	0.99	6864

4.7 Model Comparison

Hence, in this dataset we can observe that

- Logistic Regression: Accuracy is 0.98
- Passive Aggressive Classifier: Accuracy is 0.99

5 Conclusion

In this chapter, we saw how we handled the experimental part by presenting our data set, and passing by the data pre-processing which includes importing the required dependencies, removing the stop words, and cleaning the data. Then we moved to the text pre-processing phase where we applied some of the NLP techniques like the stemming and the vectorization of the data. After that we moved to using the Logistic regression and Passive Aggressive Classifier machine learning algorithm to train our model and we evaluated it by measuring the accuracy score. Lastly, we created a predictive system so we can add info to see if it is fake or real.

In our experiment we used two machine learning algorithms (Logistic Regression and Passive Aggressive Classifier) to see which one has the best accuracy and as we saw, the result was 97,9% for the Logistic regression algorithm, and 99% for the Passive Aggressive Classifier algorithm, so we can say that the Passive Aggressive Classifier is the best Algorithm we can use to detect the fake news in our case.

General Conclusion

In recent years, it has become difficult for users to access accurate and reliable information because of the increased amount of information on social media. In this study, a model is proposed to detect fake news in social media by supervised artificial intelligence algorithm.

Social media has become a huge part of our lives, since most of the people gave up on watching TV the social media became the first source of news, and that means we receive a huge amount of information when we are browsing which can be real or fake.

In this study we worked on creating a machine learning model that can detect if the received information fake or real.

We struggled in collecting dataset and finding good resources and related works concerning this subject because it is a new subject started after the US presidential elections 2016.

References

[1] Andrew Ng. (2018). *Stanford CS229: Machine Learning | Autumn 2018.* Stanford, California, USA.

[2] Hoss Belyadi, Machine Learning Guide for Oil and Gas Using Python, 2021, p3

[3] Arm, https://www.arm.com/, Visited on April 1, 2022

[4] Judith Hurwitz and Daniel Kirsch, Machine Learning for dummies, IBM Limited Edition, 2018, page 05.

[5] Bernard Marr, https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read, Visited on April 1, 2022

[6] Julianna Delua, march 2021, https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning, visited on April 2, 2022

[7] Nwishop, www.nwishop.com, visited on April 2, 2022

[8] Linear Regression Explained for Beginners in ML, www.medium.com, visited on April 2, 2022

[9] Logistic regression using PyTorch, www.linuxhint.com, visited on April 3, 2022.

[10] Rohith Gandhi, Jun 7, 2018, www.towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms, visited on April 3, 2022.

[11] Satyam Mishra, breaking down the Support vector machine algorithm, www.towardsdatasciene.com, visited on April 3, 2022.

[12] Julianna Delua, march 2021, https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning, visited on April 3, 2022.

[13] Image Charts, www.image-charts.com, visited on April 3, 2022.

[14] ISHA SALIAN, August 2, 2018, https://blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/, visited on 8 april 2022

[15] Richard S. Sutton, Introduction: The Challenge of Reinforcement Learning, Kluwer Academic Publishers, 1992, p1.

[16] Nwishop, www.nwishop.com, visited on April 8, 2022

[17] Victor Roman, an 18, 2020, https://medium.datadriveninvestor.com/deep-learning-a-comprehensive-introduction, visited on April 8, 2022.

[18] Khelil, Mohamed Imed, and Mohamed Ladjal. "Hybrid Predictive Models for Water Quality Assessment Based on Water Quality Index Using ANN, LSSVM and multivariate statistical Methods." 9th (Online) International Conference on Applied Analysis and Mathematical Modeling (ICAAMM21) June 11-13, 2021, Istanbul-Turkey.

[19] Shai Shalev-Shwartz and Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, 2014, p268.

[20] Farhad Malik, May 18, 2019, https://medium.com/fintechexplained/neural-network-activation-function-types-a85963035196, visited on April 9, 2022

[21] Sagar Sharma, Activation functions in Neural Network, www.towardsdatascience.com, visited on April 9, 2022

[22] Sagar Sharma, Activation functions in Neural Network, www.towardsdatascience.com, visited on April 9, 2022

[23] Sagar Sharma, Activation functions in Neural Network, www.towardsdatascience.com, visited on April 9, 2022

[24] V7 Labs, www.v7labs.com, visited on April 10, 2022.

[25] An overview of the activation functions used in neural networks, Nov 13,2017, www.github.io, visited on on April 12, 2022.

[26] aravindpai, February 17, 2020, https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/, visited on April 12, 2022.

[27] Analytics Vidhya, www.analyticsvidhya.com, visited on April 12, 2022.

[28] Avijeet Biswal, feb 21, 2022 https://www.simplilearn.com/tutorials/deep-learning-tutorial/convolutional-neural-network, visited on April 12, 2022.

[29] Analytics Vidhya, www.analyticsvidhya.com, visited on April 12, 2022

[30] Analytics Vidhya, www.analyticsvidhya.com, visited on April 13, 2022

[31] Ventsislav Yordanov, Nov 17, 2018, https://towardsdatascience.com/introduction-to-natural-language-processing-for-text-df845750fb63, visited on April 13, 2022.

[32] Devopedia, www.devopedia.org, visited on April 16, 2022.

[33] Java point, https://www.javatpoint.com/nlp, visited on April 16, 2022.

[34] Paramita (Guha) Ghosh on January 7, 2022, https://www.dataversity.net/fundamentals-natural-language-processing-natural-language-generation/, visited on April 19, 2022.

[35] Natural Language Processing Tutorial, www.guru.com, visited on April 19, 2022

[36] Daniel Johnson , march 08 2022, https://www.guru99.com/nlp-tutorial.html, visited on April 22, 2022.

[37] Keerthana Durai, Jul 06 2021, https://medium.com/geekculture/basics-of-natural-language-processing-for-beginners, visited on April 23, 2022.

[38] Srinivas Chakravarthy, June 19, 2019, https://towardsdatascience.com/tokenization-for-natural-language-processing, visited on April 25, 2022.

[39] Srinivas Chakravarthy, Jun 19, 2020, Tokenization for NLP, towardsdatascience.com, visited on April 25, 2022.

[40] Ventsislav Yordanov, Nov 17 2018, https://towardsdatascience.com/introduction-to-natural- language-processing-for-text-df845750fb63, visited on April 25, 2022.

[41] Keerthana Durai, Jul 06 2021, https://medium.com/geekculture/basics-of-natural-language-processing-for-beginners, visited on April 27, 2022.

[42] NLP Stanford Education, https://nlp.stanford.edu/, visited on April 28, 2022.

[43] Diego Lopez, Feb 17, 2021, Text Normalization for NLP, www.towardsdatascience.com, visited on April 29, 2022.

[44] Ventsislav Yordanov, Nov 17 2018, https://towardsdatascience.com/introduction-to-natural-language-processing-for-text-df845750fb63, visited on April 29, 2022.

[45] Ventsislav Yordanov, Nov 17 2018, https://towardsdatascience.com/introduction-to-natural-language-processing-for-text-df845750fb63, visited April 30, 2022.

[46] GeeksforGeeks, https://www.geeksforgeeks.org, visited on April 30, 2022

[47] Ted Mei, Demystify TF-IDF in Indexing and Ranking, www.medium.com, visited on May 1, 2022

[48] Data Scientest, www.datascientest.com, visited on May 1, 2022

[49] NLTK, https://www.nltk.org/, visited on May 1, 2022

[50] Nova, O., Oxford, L., 7amp; Curry, H. (2020). Digital 2020: 3.8 billion people use social media.

[51] Denise-Marie Ordway, J. (2019). Fake news and the spread of misinformation: A research roundup.

[52] Burns, K. S. (2017). Social media: a reference handbook: a reference handbook. ABC-CLIO.

[53] World Economic Forum. (2014). 10. The rapid spread of misinformation online.

[54] Patton, D. U., Eschmann, R.D., & Butler, D. A. (2013). Internet banging: New trends in social media, gang violence, masculinity and hip hop. Computers in human behaviours, 29(5), A54-A59.

[55] Khan, S. A., Alkawaz, M. H., & Zangana, H. M. (2019, june). The use and abuse of social media for spreading fake news. In 2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)(pp. 145-148). IEEE.

[56] Kelly Stahl, Fake news detection in social media, University Circle, Turlock, CA 95382, May 2018.

[57] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu, Fake News Detection on Social Media: AData Mining Perspective, Volume 19, 2017, p24.

[58] Helen Lee Bouygues, 2019, https://reboot-foundation.org/en/misinformation-effect, visited on 6 May 2022.

[59] Brendan Nyhan and Jason Reifler. When corrections fail: The persistence of political misperceptions. Political Behavior, 32(2):303-330, 2010.

[60] Daniel Kahneman and Amos Tversky. Prospect theory: An analysis of decision under risk. Econometrica: Journal of the econometric society, pages 263291, 1979.

[61] Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. Journal of Risk and uncertainty, 5(4):297–323, 1992.

[62] Henri Tajfel and John C Turner. An integrative theory of intergroup conflict. The social psychology of intergroup relations, 33(47):74, 1979.

[63] Henri Tajfel and John C Turner. The social identity theory of intergroup behavior. 2004.

[64] Solomon E Asch and H Guetzkow. Effects of group pressure upon the modification and distortion of judgments. Groups, leadership, and men, pages 222–236, 1951.

[65] Jean-Noel Kapferer. Rumors: Uses, Interpretation and Necessity. Routledge, 2017.

[66] Richard Gunther, Paul A. Beck, and Erik C. Nisbet. Fake News May Have Contributed to Trump's 2016 Victory. Ohio State University. Retrieved from https://www.documentcloud.org/documents/4429952-Fake-News-May-Have-Contributed-to-Trump-s-2016.html.

[67] Emilio Ferrara, Onur Varol, Clayton Davis, Filippo Menczer, and Alessandro Flammini. The rise of social bots. Communications of the ACM, 59(7):96–104, 2016.

[68] Alessandro Bessi and Emilio Ferrara. Social bots distort the 2016 us presidential election online discussion. First Monday, 21(11), 2016.

[69] Justin Cheng, Michael Bernstein, Cristian DanescuNiculescu-Mizil, and Jure Leskovec. Anyone can become a troll: Causes of trolling behavior in online discussions. In CSCW '17.

[70] Zi Chu, Steven Gianvecchio, Haining Wang, and Sushil Jajodia. Detecting automation of twitter accounts: Are you a human, bot, or cyborg? IEEE Transactions on Dependable and Secure Computing, 9(6):811–824, 2012.

[71] Michela Del Vicario, Gianna Vivaldo, Alessandro Bessi, Fabiana Zollo, Antonio Scala, Guido Caldarelli, and Walter Quattrociocchi. Echo chambers: Emotional contagion and group polarization on facebook. Scientific Reports, 6, 2016.

[72] Walter Quattrociocchi, Antonio Scala, and Cass R Sunstein. Echo chambers on facebook. 2016.

[73] Christopher Paul and Miriam Matthews. The russian rehose of falsehood propaganda model.

[74] Robert B Zajonc. Attitudinal e ects of mere exposure. Journal of personality and social psychology, 9(2p2):1, 1968.

[75] Robert B Zajonc. Mere exposure: A gateway to the subliminal. Current directions in psychological science, p228, 2001.

[76] Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabio Petroni, Antonio Scala, Guido Caldarelli, H Eu-gene Stanley, and Walter Quattrociocchi. The spreading of misinformation online. Proceedings of the National Academy of Sciences, 113(3):554{559, 2016.

[77] Townsend, T 2016, 'Meet the Romanian Trump Fan behind a Major Fake News Site', Retrieved from https://www.inc.com/

[78] Sydell, L 2016, 'We Tracked Down a Fake-News Creator in the Suburbs. Here's What We Learned', National Public Radio, November 23, Retrieved from http://www.npr.org/

[79] Sakshini Hangloo, Bhavna Arora, Fake News Detection Tools And Methods – A Review, India, p4.

[80] X. ZHOU, A. JAIN, V. V. PHOHA, and R. ZAFARANI, "Fake news early detection: A theorydriven model," arXiv, vol. 1, no. 2, pp. 1–25, 2019.

[81] Snopes, http://www.snopes.com, visited on May 20, 2022

[82] Fiskkit, http://fiskkit.com, visited on May 20, 2022

[83] Mykhailo Granik and Volodymyr Mesyura. Fake news detection using naive bayes classi er. In Electrical and Computer Engineering (UKR-CON), 2017 IEEE First Ukraine Conference on, pages 900-903. IEEE, 2017.

[84] Shlok Gilda. Evaluating machine learning algorithms for fake news detection. In Research and Development (SCOReD), 2017 IEEE 15th Student Conference on, pages 110–115. IEEE, 2017.

[85] William Yang Wang. "Liar, liar pants on fire": A new benchmark dataset for fake news detection. arXiv preprint arXiv:1705.00648, 2017.

[86] Yimin Chen, Niall J Conroy, and Victoria L Rubin. Misleading online content: Recognizing clickbait as false news. In Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection, pages 15–19. ACM, 2015.

[87] Johannes F¨urnkranz. A study using n-gram features for text categorization. Austrian Research Institute for Artifical Intelligence, 3(1998):1–10, 1998.

[88] Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. A stylometric inquiry into hyperpartisan and fake news. arXiv preprint arXiv:1702.05638, 2017.

[89] Sadia Afroz, Michael Brennan, and Rachel Greenstadt. Detecting hoaxes, frauds, and deception in writing style online. In ISSP'12.

[90] J. Cao, P. Qi, Q. Sheng, T. Yang, J. Guo, and J. Li, "Exploring the role of visual content in fake news detection," arXiv, no. April, 2020, doi: 10.1007/978-3-030-42699-6_8.

[91] Z. Jin, J. Cao, Y. Zhang, J. Zhou, and Q. Tian, "Novel Visual and Statistical Image Features for Microblogs News Verification," IEEE Trans. Multimed., vol. 19, no. 3, pp. 598–608, 2017, doi: 10.1109/TMM.2016.2617078.

[92] C. Boididou et al., "Verifying Multimedia Use at MediaEval 2016," CEUR Workshop Proc., vol. 1739, pp. 4–6, 2016.

[93] J. Shin, L. Jian, K. Driscoll, and F. Bar, “The diffusion of misinformation on social media: Temporal pattern, message, and source,” 2018, doi: 10.1016/j.chb.2018.02.008.

[94] S. Kwon, M. Cha, and K. Jung, “Rumor detection over varying time windows,” PLoS One, vol. 12, no. 1, pp. 1–19, 2017, doi: 10.1371/journal.pone.0168344.

[95] Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. News verification by exploiting conflicting social viewpoints in microblogs. In AAAI’16.

[96] Zhiwei Jin, Juan Cao, Yu-Gang Jiang, and Yongdong Zhang. News credibility evaluation on microblog with a hierarchical propagation model. In ICDM’14.

[97] Manish Gupta, Peixiang Zhao, and Jiawei Han. Evaluating event credibility on twitter. In PSDM’12.

[98] Kai Shu, Amy Silva, Suhang Wang, Jilian Tang, and HuanLiu, Fake news detection on social media: a Data mining perspective, volume 19, p3.

[99] Ozbay, F. A., & Alatas, B. (2020). Fake news detection within online social media using supervised artificial intelligence algorithms. *Physica A: Statistical Mechanics and its Applications*, *540*, 123174.

[100] Raja, M. S., & Raj, L. A. (2022). Fake news detection on social networks using Machine learning techniques. *Materials Today: Proceedings*.

[101] Hamid, A., Shiekh, N., Said, N., Ahmad, K., Gul, A., Hassan, L., & Al-Fuqaha, A. (2020). Fake news detection in social media using graph neural networks and NLP Techniques: A COVID-19 use-case. *arXiv preprint arXiv:2012.07517*.

[102] Mahmud, F. B., Rayhan, M. M. S., Shuvo, M. H., Sadia, I., & Morol, M. K. (2022, March). A comparative analysis of Graph Neural Networks and commonly used machine

learning algorithms on fake news detection. In *2022 7th International Conference on Data Science and Machine Learning Applications (CDMA)* (pp. 97-102). IEEE.

[103] Tida, V. S., Hsu, D., & Hei, D. (2022). Unified fake news detection using transfer learning of bidirectional encoder representation from transformers model. *arXiv preprint arXiv:2202.01907*.