Contents

List of Figures

List of Tables

Executive Summary

During the COVID-19 pandemic, social networks have seen a substantial amount of false news [28]. Furthermore, people have been discussing numerous false remedies to cure COVID-19. However, these remedies are extremely dangerous to human health. The director-general of the World Health Organization calls it “infodemic" [31] because of the amount of misinformation, disinformation, and "false news" relating to COVID-19. With the enormous number of news regarding COVID-19 on the Internet, it is difficult for many to assess truthfulness. Moreover, the riots and panic shopping also occurred due to the propagation of “false news" [31]. In this thesis project, I aim to build an automated COVID-19 misinformation detection system and investigate the value of a social network structure compared to the text-based classification approach. I have implemented a variety of techniques to detect fake news and misinformation in tweets related to COVID-19. The research objective is to classify each tweet as either true/fake with various text feature representations techniques and graph structure to compare and evaluate their performances. The project is comprised of two parts which are text-based and graph structure-based fake news detection techniques. For the first part, I conduct five different classification algorithms relying on various embeddings techniques including BoW, TF-IDF and Word2Vec embeddings. For the second part, I represent the data in a graph structure and learn the feature representations for the nodes using the Node2Vec embedding algorithm, which can then be used for the downstream classification task. Different studies [29, 32] revealed that N-grams based features are efficient in identifying false information. The author in [33] uses TF-IDF feature to classify hoax news with 84.67% accuracy. The classifiers used in this paper are more modern KNN, MNB, LSVM, and logistic classification. In this thesis, it was concluded that the use of a combination of features in the GNB classifier has the most accurate performance. Moreover, text-based modelling outperforms graph-based modelling in terms of ROC score, accuracy as well as weighted average F1 score.
The overall aim was to produce a system that can auto-detect fake news in tweets related to COVID-19, improving the results obtained by previous studies.

The main achievements of the project are as follows:

- I learnt how to clean and prepare large scale data in JSON format and convert it into a suitable format to be ready for the study analysis.
- I wrote a total of 18898 lines of source code (in Python), implementing all the experiments analysed in this thesis.
- I learnt how to build an NLP pipeline including word tokenisation and removing stopwords from the tweet text.
- I wrote functions in python code that encode different text feature representation including BoW, TF-IDF and Word2Vec.
- I investigated and trained many classification models with different feature representation combinations.
- I represented the data as a network structure (graph) and modelled it to be used in the context of detecting false information.
- I applied continuous feature representations for the nodes in a graph to be used for the classification task.

Chapter 1

Contextual Background

1.1 Introduction

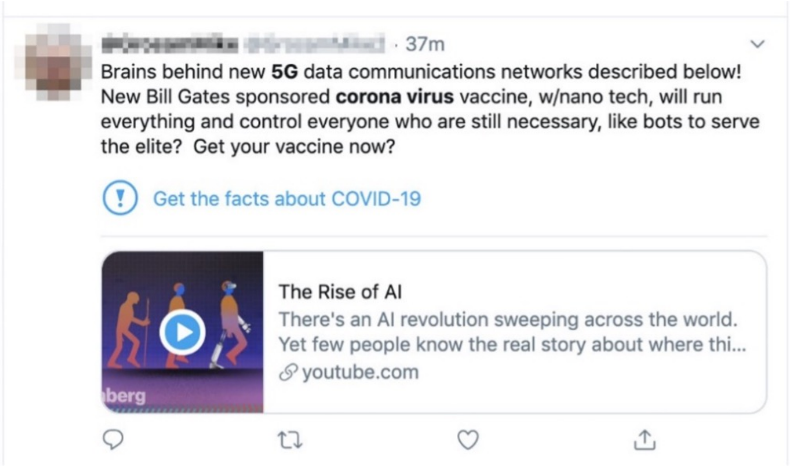
The pandemic has been accompanied by a massive wave of false news that made it more difficult for the public to comprehend the truth. People use different social networks to share information and communicate with each other. However, this also opens a room to propagate fake information on these platforms. The spread of fake news can be extremely dangerous, especially when facing the COVID-19 epidemic. Moreover, fake news about fake remedies to cure COVID-19 can be hazardous to human health and lead to human death [43]. Processing an enormous amount of text manually on these social media platforms, particularly Twitter, is troublesome. Therefore, it is essential to present automatic methods to analyze text and annotate if the news is fake or real. Fake or inaccurate information can also put a human body at risk of exposure to viruses and also cause anxiety, mental stress [40]. Besides, the purpose of spreading fake news can also be to mislead people to gain attention. Believing wrong information can be an easy task; however, refuting fake news is a challenging task – not many people ask for evidence to back up the statement.

In the digital era, the general public can be easily misled by fake news. This can be due to various reasons, such as lack of education and lack of interest in news verification. However, the propagation of fake news can be dangerous. Misinformation on COVID-19 can encourage individuals not to wear masks, take certain precautions, or even get a vaccine when these are the factors that can keep them safe. Public health messages should be considered since fake information can be posted as very concise headlines on the internet, especially on Twitter. Therefore, it is crucial to automatically differentiate real and fake news related to COVID-19 to avoid hazardous consequences to public health.

A recent study [39] reported that 1500 tweets were posted on Twitter regarding COVID-19, including 1274 were completely false and 226 were partially false. This implies that it may be unknown to the individuals that they are spreading fake news. If they knew that the shared information was fake, then it is probable that they would have preferred sharing true news. Hence, It is beneficial to understand the essence of propagating misinformation phenomenon on these social media platforms, especially Twitter. Twitter allows its users to share ideas easily, and posting a tweet or re-posting a particular tweet on Twitter does not require much effort. Nevertheless, words can be misused, and stories can be made up to make everyone believe using conspiracy theories. Research [34] shows that the number of conspiracy theories has increased due to the spread of COVID-19 worldwide. Another study [2] linked 5G technology with COVID-19, this has become dangerous since several people have started to burn the 5G towers in the United Kingdom. Finally, another study revealed that fake news on Twitter gets more response and spread faster than real news [2].

1.2 Problem Statement

The COVID-19 pandemic has caused many restrictions and induced different issues that impact people's social and economic lives. Employees have stopped in office work due to the fear of COVID-19. The root of this issue has become even more profound due to social media amplifying fake news. Moreover, the COVID-19 pandemic has become one of the significant events in all of our lives, and the impact has also been enormous due to approximately 199,376,990 million people being infected from the COVID-19 [1].



[redacted] · 37m

Brains behind new **5G** data communications networks described below! New Bill Gates sponsored **corona virus** vaccine, w/nano tech, will run everything and control everyone who are still necessary, like bots to serve the elite? Get your vaccine now?

Get the facts about COVID-19

The Rise of AI

There's an AI revolution sweeping across the world. Yet few people know the real story about where thi...

youtube.com

Figure 1.1: A tweet portraying false information that can mislead and manipulate many individuals who believe this statement is true [20].

1.4 Methodology

1.4.1 Research Question

This project studies the value of both text and a social network structure for an automated misinformation detection system. The main research question to explore in this thesis – is misinformation better detected using text, or can the social network graph structure itself be a better detector?

1.4.2 Research Hypothesis

In this research, the central hypothesis to test is whether an input of a node feature representation in a graph to a supervised classifier algorithm would yield better performance than text-based features in detecting fake tweets. To the best of my knowledge, this approach fills a gap in the literature.

1.4.3 Research Methodology

In this project, both traditional NLP techniques and social network analysis are applied and compared. The project consists of two main parts. The first part is the text-based classification technique, which involves building supervised classification algorithms. These classification models classify tweets as either true/fake based on tweet text content. This research will use various text features, such as BoW, TF-IDF, and Word2Vec, and compare their performances with different classification models.

The second part uses the graph-structure classification technique, where the data is represented in a graph containing nodes and edges (connections between the nodes). Then, this graph representation will be fed as input to supervised ML algorithms. This graph representation learning approach shows the relationships between the tweet contents and the key players in promoting the information, as well as identifying the reasons that promote the information. Therefore, data will be represented in a graph in which nodes represent tweets id and edges represent the relationships between them (retweets/replies). This will allow classifying the tweet as either true/fake based on graph structure and not the tweet text.

Moreover, the node classification technique will be used to classify the tweet in the output layer. Still, it needs further processing to get it working correctly.

1.5 Objectives

A list of objectives was provided to design an automated COVID-19 misinformation detection system that identifies tweets that include fake content. These objectives aim to create such a system that assists users in verifying the truthfulness of any COVID-19 related tweets. The main objectives of this research project are as follows:

2.2.3 Word2Vec

Word2Vec is a yet another popular method of representing the document vocabulary in the form of feature vectors. It is computationally expensive since it requires training a shallow neural network architecture, but the performance gain provided by this method is very significant since it can identify which words are semantically similar and capture the context of every token in the document vocabulary. To train a neural network to learn word features, the features are initially represented in one-hot encoding. In this format, each word is represented by a vector of length equal to the size of the vocabulary. The vector has zero at all indices and one at the index corresponding to the word it is supposed to represent. In this technique, the shallow network is trained for a language-based problem which is not necessarily relevant to the problem we are attempting to solve such as auto completion of text. Once the model is trained for such a problem, it automatically learns the semantic relationship between the words. Consequently, the weights of this model can be used as the representations of tokens. The Word2Vec method has two variations [27]. The variations and their details are mentioned as follows:

- **Common Bag of Words:** In the CBoW method, the context of a given target word is used to predict that word. Context of a word refers to the surrounding n-grams of a word. The input layer has a dimension equal to kxV where k is the number of context words takes and V is the number of unique words in the vocabulary. The output layer has a dimension equal to V. The one-hot representations of each word are fed into a separate input layer. These representations are then multiplied with corresponding weight matrices and fed into the hidden layer. The weight matrix corresponding to the output of the hidden layer serves as the word representation which is extracted after training the shallow neural network.

- **Skip-Gram model:** The input and output dimensions of the skip-gram model is opposite to the CBoW model. In this case, we use a specific word to predict the context. The input layer has a dimension equal to V, where V is a specific word. The output layer has a dimension of kxV where k is the number of tokens in the context and V is the size of vocabulary. In case of this model, the weight matrix between the input layer and the hidden layer is used as the representation for each word after training the model.

2.3 Graph Terminology and Representation

This section illustrates some basic knowledge about data graph representation.

2.3.1 Graph Representation

Graph representation comprises data that are connected to each other. The critical point to highlight is that graph data is similar to any other data type. However, the main difference between graph data and other data types is that graph data is structured data. This means that in addition to defining the data points, the relation between the data points should be defined as well, which provides us with a structure. This extra information is often valuable in analyzing the data, so graphs and graph neural networks represent better performance than other data structures.

2.3.2 Nodes and Edges

Nodes are the data points that are represented in the graph. What makes graph data different from other types of data is the representation of edges that connect to nodes or data points that are similar. Hence, if two nodes are related or identical, there will be an edge connecting the nodes. In this context, the nodes could present the tweet text, and the edges are the communication between Twitter users by retweeting/replying to a given tweet.

2.3.3 Features

The recent papers in the neural network domain [37, 24] allow us to introduce a piece of additional information in the form of features. These features are the extra information that each node could carry on. This information in the form of a feature could be the number of followers, likes, total replies, etc.

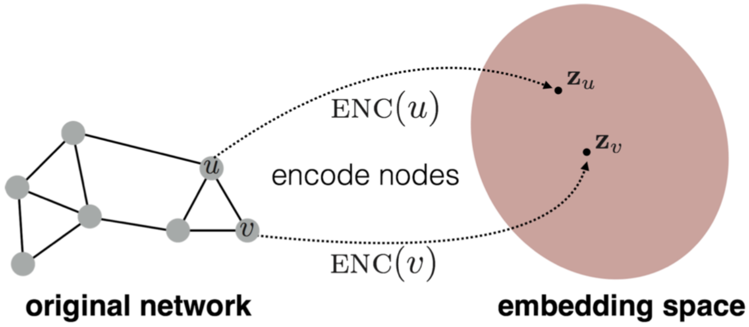


Figure 2.1: A figure illustrating node embedding learning methodology where we have the encoder (ENC) that maps nodes to a low-dimensional embedding space in a way that show the relative positions of the nodes in the original graph. [17] .

2.3.4 Directed Graph

In the case of a directed graph, an arrow shows how the nodes are related in this case. For example, node A is connected to node B, but node B is not connected to node A. So what we do is we draw an arrow from the source (Node A) to the target (Node B). Twitter is an example of a directed graph in the social media network where if the first person follows the second person, the second person does not need to follow back or follow the first person. To mathematically represent graph data, this can be shown in the following notation:

$$G = (V, E, u) \tag{2.10}$$

where G is the graph, V describes vertices (nodes), and E is a vector that describes edges or links, including two types of the matrices adjacency matrix and weights matrix. u represents the feature vector.

2.3.5 Undirected Graph

In an undirected graph, there are two arrows or no arrows at all to define an undirected graph. Moreover, in an undirected graph, the relation between the nodes is symmetric. This implies that when node A is connected to Node C, Node C is also connected to node A.

2.4 Representation Learning on Graphs

This part illustrates the approach behind learning node embeddings. Likewise, demonstrates the Node2Vec embedding algorithm which is used in the project implementation described in chapter 3

2.4.1 Learning Node Embeddings

One of the primary challenges faced in graph representation learning is to select an embedding method for the graph nodes [17]. The aim of learning the appropriate feature representation of nodes is to transform these nodes to an embedding space and use the resultant transformation of each node as a low-dimensional vector as an input to the classification algorithm. One of the most used approaches for learning node embeddings is the use of an encoder-decoder model. Figure 2.1 shows an overview of node embedding learning approach. The encoder part is responsible for mapping the nodes into the embedding space using a well devised methodology. Usually, a shallow embedding approach is employed where embedding function is simply a lookup function which extracts the embedding vector from the embedding matrix when given a unique node id using the strategy that had been devised to generate the embedding matrix. On the other hand, the decoder part of the network is used to map the feature representations or embeddings to the respective node or structural and positional information about that node such that the node and its neighborhood can be reconstructed using this information.

3. Pre-processing of data to be in the right form to function correctly when feeding it to a classification model.

4. Representing text data using BoW text feature.

5. Representing text data using TF-IDF text feature.

6. Representing text data using Word2Vec text feature.

7. Representing the data in a graph and using Node2Vec algorithm to automatically learn the feature representation.

8. Development of the prediction model.

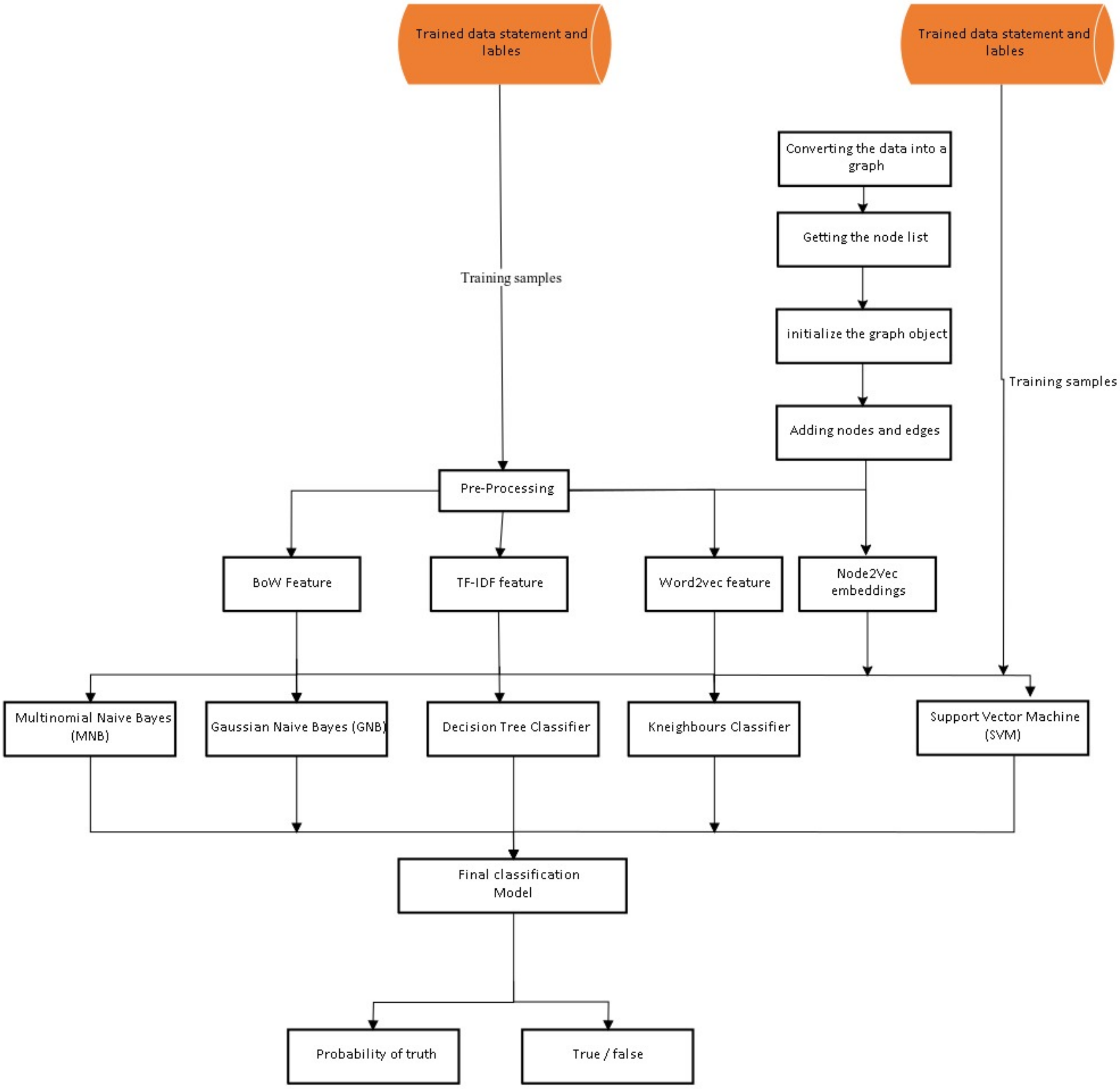

Trained data statement and lables
Trained data statement and lables
Converting the data into a graph
Getting the node list
initialize the graph object
Adding nodes and edges
Training samples
Training samples
Pre-Processing
BoW Feature
TF-IDF feature
Word2vec feature
Node2Vec embeddings
Multinomial Naive Bayes (MNB)
Gaussian Naive Bayes (GNB)
Decision Tree Classifier
Kneighbours Classifier
Support Vector Machine (SVM)
Final classification Model
Probability of truth
True / false

Figure 3.9: The architecture of the proposed pipeline.

Classifier	Features
MNB	BoW TF-IDF Word2Vec
GNB	BoW TF-IDF Word2Vec
KNN	BoW TF-IDF Word2Vec
DT	BoW TF-IDF Word2Vec

Table 3.1: The classifiers with various independent feature representation

4. Taking the experiment a step ahead to try different feature combination with each classifier as the following:

   - **BoW and TF-IDF**
   - **BoW and Word2Vec**
   - **TF-IDF and Word2Vec**
   - **BoW, TF-IDF and Word2Vec**

3.7 Graph Modelling

This part of the research involves representing the data in a graph structure and getting the embeddings of the structure using Node2Vec algorithm. The graph consists of tweet id as source nodes and retweets/replies as connections between those nodes (destination nodes). This will enable us to see how many retweets/replies each tweet receive. Hence, classifying the tweet based on its structure not the text itself. Figure 3.11 shows the built graph.

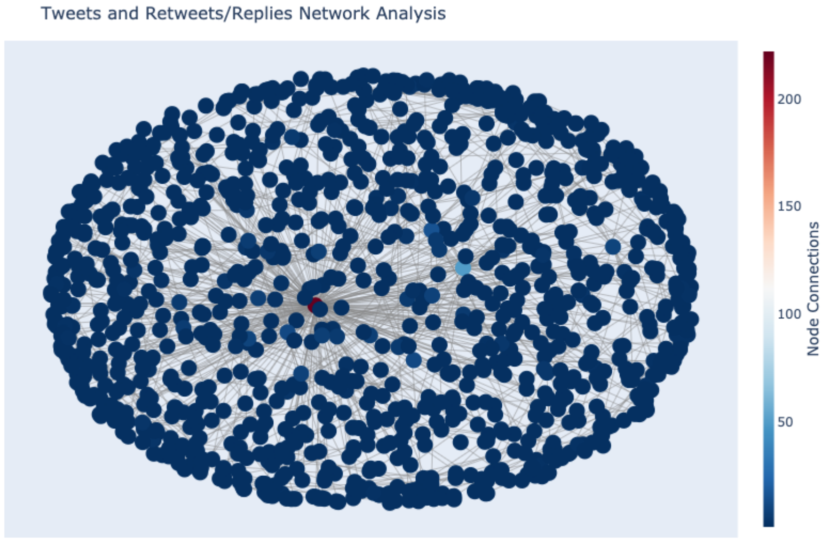


Figure 3.11: A graph representing the tweet ID and its retweets/replies connections

values are compared to those obtained using a combination of features , Table 4.13, the highest values of ROC score, accuracy, micro average f1 score and weighted average f1 scores are 0.82, 0.81, 0.69 and 0.79 respectively. These results are 12.33%, 6.58%, 15% and 11.27% higher than those achieved by using independent features allowing us to concretely conclude that using a combination of text-based features improves classification results of machine learning models significantly.

Another observation made through the comparison of independent and combined text features was that out of all machine learning models used, MNB performs the best with the use of independent features whereas GNB works well with a combination of features. This helps improve our results further and hence it can for a baseline to be used in further research's in the future.

	ROC	**Accuracy**	**Micro avg F1**	**Weighted avg F1**
MNB	**0.73**	**0.76**	0.55	**0.71**
GNB	0.66	0.68	**0.6**	0.69
DT	0.47	0.58	0.49	0.63
KNN	0.56	0.71	0.51	0.68

Table 4.12: Average performance for all classifiers using independent features

	ROC	**Accuracy**	**Micro avg F1**	**Weighted avg F1**
MNB	0.77	0.76	0.43	0.65
GNB	**0.82**	**0.81**	**0.69**	**0.79**
DT	0.60	0.74	0.60	0.72
KNN	0.65	0.75	0.57	0.72

Table 4.13: Average performance for all classifiers using a combination of features

4.10.2 Comparison of text and graph modelling

An important purpose of this research is to determine which method is more accurate, text-based modelling or graph-based. Since many experiments were performed in the text-based part, it was decided to use combination of features in GNB classifier for comparison since it had the finest performance. The average of all values for each evaluation metrics was taken and a separate table was constructed. The comparison can be seen in Figure 4.2.

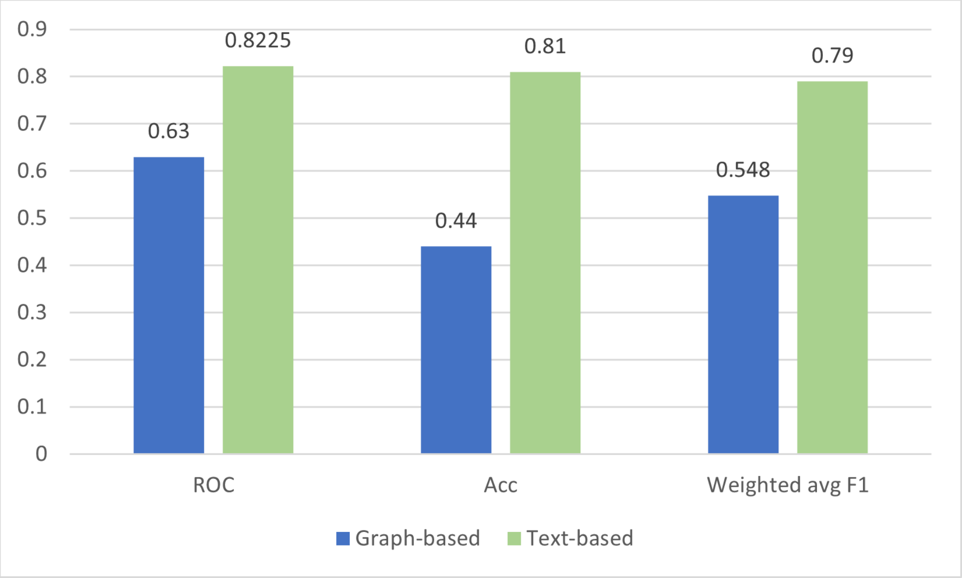


Figure 4.2: Comparison of graph-based and text-based modelling

In the figure above, it can be clearly seen that text-based modelling outperforms graph-based modelling in terms of ROC score, accuracy as well as weighted average F1 score. In the graph-based experiment,

5.1 Summary

This thesis can be summarized in the following manner:

Chapter 1 provides an introduction to the non-technical aspects of this thesis. It explains the importance of this research as well as the motivation behind it whilst also discussing the various challenges linked with the project.

Chapter 2 contains a detailed explanation of all technical aspects which need to be understood for this research. A sufficient amount of literature review is provided for various machine learning techniques used for textual data and necessary terminologies linked with this research are explained. This chapter also includes details regarding the dataset to be used for our experiments.

Chapter 3 provides detailed descriptions of the experiments performed in this research. All steps performed to achieve the output are listed in a systematic manner and are explained conspicuously. This chapter also includes explanations of the machine learning algorithms to be used and lastly all evaluation metrics used are discussed in-depth.

Chapter 4 is an extremely crucial chapter since it contains all results obtained in this research and a thorough comparison of text-based data and graph-based data using 5 different machine learning algorithms is provided as well. The results are analyzed and any shortcomings are mentioned.

5.2 Project Status

The initial objectives of this project were to:

- Labelling the dataset, adding binary labels to the dataset as true/fake is required. (Complete)
- Building a baseline model that classify each tweet and improving its performance with coming experiments. (Complete)
- Encoding BoW, TF-IDF and Word2Vec feature representation from the tweet text. (Complete)
- Training classifiers with independent feature representation. (Complete except SVM classifier with independent features)
- Training more than one feature to represent the text. (Complete)
- Represent the tweet text related information in a graph structure. (Complete)
- Get input vectors from the graph. (Complete)
- Training classifiers with graph feature representation. (Complete)
- Compare and analyse the performance of both text and graph approaches. (Complete)
- Applying graph feature representation to a GNN algorithm. (Incomplete)

Some obstacles I faced during the execution of the research implementation include the following:

- Running the following experiments:
  - SVM algorithm with independent features including BoW, TF-IDF and Word2Vec : This could not be achieved because the model took more than 48 hours to train on local machine with great difficulty.
  - Combination of BoW, TF-IDF, Word2Vec based features: This also proved to be problematic as training took a long time.