Classification of Conspiratorial Content on YouTube --Performance based on different feature extractions

(Master's Thesis)

Omer Ahmed
o.ahmed@tilburguniversity.edu
SNR: 2035939

Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Data Science and Society,
Master Track Business,

Thesis Committee
Supervisor: dr. Michal Klincewicz
Second Reader: dr. Travis Wiltshire

Tilburg University

Tilburg, The Netherlands
December 2020.

Preface

I would like to thank Michal Klincewicz for the direction, guidance and time given throughout the thesis journey, as it has always been helpful. Furthermore, I would like to thank Mark Alfano and his colleagues who shared their research and dataset which I used for my own thesis. Lastly, I would like to thank Tolgahan Akyazi , Raf van den Eijnden and Siebe Albers who shared the increased data set which I used for my thesis.

Link to the code and data files : https://github.com/omerrahmed/Thesis-2020.git

Classification of Conspiratorial Content on YouTube -- --Performance based on different feature extractions

Omer Ahmed

Abstract

Amongst the pros of social media there are many negatives that can impact the world in an unthinkable manner. This calls for action to be taken to at least monitor, if not stop, the propagation of potential conspiratorial-content on the social media platform, especially YouTube. The goal of this thesis is two-fold: (1) to investigate which type of features result in better performance of the classifiers and, (2) does increasing the dataset synthetically, improve the performance of the classifiers trained on features extracted from the increased dataset. Van den Eijnden and Akyazi illustrate how the categorization of conspiratorial videos is done using different algorithms. Since the original dataset is small and imbalanced, SMOTE (Synthetic Minority Over-sampling Technique) is used to balance and increase the size of the dataset. This is coupled with extraction of different features to train different models, with the aim to improve the performance of the classifiers. By doing so, this thesis aims at filling the research gap in this field (since this has not been done before on this dataset or this problem). Results show that the Logistic Regression model has the best performance on the original dataset when it is trained on unigram features with a macro averaged F1 score of 0.845. Similarly for the synthetically increased dataset, the Logistic Regression classifier trained on all four types of features shows a relatively better performance. Lastly, it is observable that increasing the dataset's size through SMOTE does indeed increase the performance of the classifiers.

1 -- Introduction

1.1 Context

Social media has become a pervasive online platform, which has no physical borders nor mental ones. It is becoming a source of information for its users, despite the content being generated by the users themselves (Bialy, 2017, p. 73). Furthermore, it's easy accessibility, possible anonymity and lacking gatekeepers make it even more difficult to monitor the content circulated on it (p.74). YouTube, as a social media platform allows content to be produced and displayed from anywhere around the world, which can be targeted towards audiences of all ages, natures and traumas. The conspiratorial content that gets propagated over social media emerges from potentially "epistemic, existential, and social motives" (Douglas, Sutton and Cichocka, 2017) and might promote "significantly more extreme" ideas (H Raab, 2016). In the face of new information, individuals that do not have predispositions, are bound to accept conspiracy theories (Uscinski, Klofstad, & Atkinson, 2016, p. 60). Conspiracy theories can be referred to as rumour theories which are capable of producing large-scale social responses (Del Vicario, Bessi, Zollo, Petroni, Scala, Caldarelli, & Quattrociocchi, 2016, p. 554). As Youtube users involve people of all ages including teenagers (Collins Uduiguomen, Celestine Agwi & Faith Aliu, 2014) conspiratorial videos are capable of impacting them and society negatively, given that they are a malleable audience.

On YouTube, content that is copyright infringement is automatically filtered and can even be reported by its viewers (Tan, 2018, p. 146), however its reporting and flagging mechanism of conspiratorial content is not as developed yet. YouTube has become a host for copious amounts of conspiratorial content, which is not classified sufficiently, given the quantity

of content that exists. A detection of what classifies as conspiratorial content or not, would be helpful in reducing its impact on vulnerable audiences. Furthermore, this will make flagging and reporting of conspiratorial content easier, to bring down the deleterious mental and societal impact such videos bring.

In recent decades, YouTube has risen to become a form of information-warfare given its communicative power (Rantapelkonen & Jaitner, 2014, p. 1). Its capability of shaping emotional, psychological and public behavior, makes the information it uses as conspiracies both a resource and a weapon (Abdyraeva, 2020, p.20). Furthermore, the popularity of any given topic attributes to it a stronger communicative power (Rantapelkonen & Jaitner, 2014, p. 2), allowing conspiracy theories a more impactful role compared to other topics' videos on Youtube, especially if they are negative. Moreover, by impacting gullible audiences, they are bound to produce conflict and violence within society. Hence, a monitoring mechanism on YouTube is required, if not to stop the production of such content then to indicate its conspiratorial nature to secure gullible audiences from believing any information that comes their way. As conspiratorial videos can be classified or categorized using different algorithms, a more effective mechanism to detect them is to use different types of features.

Moreover, the "language of conspiracy" (Spencer, 2011) plays a key part in these theories, as well as their popularity on online platforms. According to Wakabayashi (2019), clickbait videos started to "game the system", meaning that the algorithm was maneuverable with the use of language. The Pizza conspiracy stands as an example of how YouTube can be used as a form of hate speech and other extremist views (Timberg, Dwoskin, Romm, & Ba Tran, 2018). In addition to their proliferation on YouTube, completely rational people are capable of

internalizing "crazy" conspiratories (Sustein, 2016), which makes strengthening their detection-mechanism an important priority.

1.2 Research Questions/Hypothesis

As YouTube is a host for different conspiratorial content that is: not only rampant due its propagators, but is also targeting malleable audiences, thus, detection of such content is vital. In this thesis, different classification algorithms are analyzed to assess which algorithm proves most useful to categorize conspiratorial content on YouTube. Pairing synthetic increase of the dataset with different types of feature extractions, with the aim to improve the classification performance, has not been researched extensively. Furthermore, a comparative analysis using different feature types extracted from the same data-set (YouTube conspiratorial content) to train different models, for assessing the most effective one has not been studied yet.

The main research question employed by this thesis is: how effectively do different Machine Learning models, trained on different features extracted from a small unbalanced dataset, classify conspiratorial Youtube video transcripts? This is then be subdivided into three main subquestions: 1) which feature extraction technique is best suitable for this problem, 2) how accurate can Logistic Regression, SVM and Naive Bayes classify this content, and 3) does synthetically increasing and balancing the data set affect model performance in classification. By embarking upon this approach this thesis aims to contribute uniquely to the research domain regarding text-classification to make detecting such content on YouTube much easier and more accurate. As there should be a balanced set-of-information available for audiences - especially gullible teenagers (Akram & Kumar, 2017. p.351), this thesis helps show which detection mechanism is most effective, so that flagging conspiratorial YouTube videos easier so that their

material is viewed skeptically (rather than as the truth). Ultimately, this thesis aims to illustrate whether classifiers trained on features extracted from synthetically increased dataset help increase the performance of the classifiers.

1.3 Findings

The research shows that overall models trained on unigram features perform better than models trained on other types of features. Furthermore, it is seen that the Logistic Regression model is able to perform better than the other classifiers. Finally, it would not be wrong to say that, indeed, the performance of the classifiers increases when trained on features that are extracted from synthetically increased dataset (as discussed in the results section).

2 -- Related Work

2.1 The world of Conspiratorial Content

The Internet has turned the world into a global village, making mass propagation of information easier. While information and content includes useful knowledge, it also includes content that can be classified as conspiratorial. It is believed that today we live in an “age of conspiracism” (van Prooijen & Douglas, 2017). Conspiratorial content can be found throughout the internet on almost all social media platforms. This includes the most popular platform called Youtube, with 2 billion active users per month (YouTube Revenue and Usage Statistics, 2019). This calls for an action to be taken to at least monitor if not to bring the propagation of such conspiratorial content to a halt.

2.2 Efforts to monitor the spread of conspiratorial content:

- *2.2.1 Feature enrichment with word vector models*

In order to tackle the problem of conspiratorial videos' propagation on Youtube, efforts have been made by Albers in their thesis. Albers uses a small unbalanced dataset of Youtube video-transcripts to detect conspiratorial content. They use the Bag of Words (BoW) approach to extract features which inturn are enriched with word vector models. This word vector model is trained on the Reddit conspiracy language dataset, aiming to improve the performance of the models. Multiple models are trained and their performance is evaluated. Word vector models have also been used by Heap, Bain, Wobcke, Krzywicki & Schmeidl (2017) for feature enrichment, and have proved to be effective in increasing the performance of the classifiers. Ultimately, it was concludible that feature enrichment using word vector models indeed increases the performance of the models when working with a small unbalanced dataset. This adds weight to research being done on small unbalanced datasets where enrichment of the features can help improve the performance.

- *2.2.2 Use of sentiment-weighting to improve classification performance*

Similar to Albers, Akyazi in their thesis also aims to develop a model which can flag the conspiratorial videos on Youtube. The same dataset used by Albers is also used by Akyazi in their thesis. However, unlike Albers, Akyazi uses the TF-IDF method to extract features. Furthermore, instead of using word vectors to enrich the features, Akyazi uses sentiment weighting through the IMDB-review dataset, to improve the performance of the classifier. Sentiment analysis is done with both positive and negative sentiments. The results of this thesis

show that using TF-IDF features, coupled with sentiment weighting of the feature vectors, results in an increased performance of the models. This, yet again sheds light on how performance on small unbalanced datasets can be improved by different ways such as sentiment weighting (Akyazi) and feature enrichment through word vectors (Albers). This thesis aims to tackle the unbalanced dataset issue in a different way as compared to both Albers and Akyazi. In this thesis, the dataset is increased using Synthetic Minority Over-sampling Technique (SMOTE) in order to balance the class distribution, which inturn increases the dataset size. The method of increasing the dataset is discussed in the subsequent SMOTE section (2.3).

- *2.2.3 Detecting conspiratorial videos about COVID-19*

Serrano, Papakyriakopoulos, & Hegelich, S. (2020) tackle the issue of identifying videos on YouTube, which are spreading misinformation regarding COVID-19. Comments specifically spreading misinformation, under the Conspiratorial videos related to COVID-19 on Youtube, are used as the feature for the classification task. Similarly, Della Vedova et al. (2018) and Popat, Mukherjee, Yates & Weikum (2018) analyze the language of dataset consisting of articles to detect fake news. Results of both researches indicate that detection of misinformation and conspiracy theories can be done through multiple methods, therefore, hinting towards innovative mechanisms to address this issue.

2.3 Synthetic increase of the data set using SMOTE

The original dataset being used in this is small and unbalanced. An unbalanced dataset refers to a data set which has an underrepresented class as compared to the other classes in the dataset (Lemaˆıtre, Nogueira & Aridas, 2017, p.5). This gives rise to an issue known as “class

imbalance" (Prati, Batista & Monard, 2009) i.e the model faces problems to learn from the minority class. This issue is common in problems relating to fraud detection and bioinformatics (Lemaˆıtre, Nogueira & Aridas, 2017, p.5) among other practical domains. To cater to this problem, different approaches have been used such as "up-sampling" where the minority class data points are duplicated to equal the number of majority class data points, and, "down-sampling" where the majority class data points are reduced to equal the number of minority class data points (Wah, Rahman, He & Bulgiba, 2016).

In addition to the approaches mentioned above, another method to deal with the imbalance dataset problem is by using synthetic minority over-sampling technique (SMOTE). SMOTE deals with the issue of overfitting which is caused by the oversampling technique (Branco, Torgo & Ribeiro, 2015). In SMOTE, new data points are synthetically generated instead of "over-sampling with replacement" (Chawla, Bowyer, Hall & Kegelmeyer, 2002, p.328). A data point is selected and k of its neighbors are looked at. A neighbor is randomly selected and is subtracted from the data point. This difference is then multiplied by a random number between 0 and 1 which is then finally added to the data point under consideration (p.328-330).

Since its advent, SMOTE has proven to be effective in different contexts and practices (Fernandez, Garcia, Herrera & Chawla, 2018). According to Chawla, Bowyer, Hall & Kegelmeyer's conclusion (2002) it is seen that, SMOTE when used on an imbalanced data resulted in an increase in the performance for the minority class. Similarly, it is seen that classifiers trained on a dataset increased through SMOTE result in a better performance as compared to the classifiers trained on "unmodified imbalanced data" (Douzas, Bacao & Last,

2018, p.16). Since SMOTE has proved to be effective, it is used in this thesis in order to balance the dataset, while aiming to improve the classifier's performance. The performance of classifiers trained on the original dataset is compared to the classifiers trained on the dataset increased through SMOTE. The comparison of the performance can be found in the results section.

2.4 TF-IDF

One of the major steps in a model-construction for text classification tasks is feature extraction (Liang, Sun, Sun & Gao, 2017, p.2-3). A number of feature extraction techniques have been used, however TF-IDF has been the "most commonly adopted" technique to extract features for text related problems (Kim, Seo, Cho, Kang, 2019, p.17). Through the TF-IDF approach one can find out "what keywords are important" (Kim & Gil, 2019, p.8). This would utilize the most important features extracted, resulting in a much smaller feature set. The TF-IDF weight is calculated in two steps. First, the TF value (term frequency) and the IDF value (inverse document frequency) is calculated. These two values when multiplied give the TF-IDF value for each word, which can then be used to extract the most important features, reducing the effect of the less important words (Gaydhani, Doma, Kendre & Bhagwat, 2018). The formulas of each of the aforementioned values can be seen in Figure 1 below. The formulas are taken from Christian, Agus & Suhartono (2016).

$$TF = \frac{Totalappearanceofawordindocument}{Totalwordsinadocument}$$

$$\mathrm{IDF} = \log \frac{AllDocumentNumber}{DocumentFrequency}$$

$$TF - IDF = TF \times IDF$$

Figure 1: TF/IDF formulas

2.5 Magic of the features and the research goal

Even though some learning models perform better than the others, it is not always only the type of model which affects the performance. Feature extraction plays a "significant part" in the development of a model (Sammons et al, 2016, 4085). For this very reason it is important to investigate which feature results in the best performance of a model. Pranckevičius & Marcinkevičius run different models on different n-gram features in order to investigate which feature results in the best performance (2017, p.221). From amongst unigram, bigram and trigram features, the results show that models trained on unigram features had the best performance. It is concluded that the performance of the models is dependent on the "n-gram properties" (Pranckevičius & Marcinkevičius, 2017).

Similarly, in a study done by Baygin (2018) on classification of textual documents, multiple n-gram features are used to train the models. This includes 2-gram, 3-gram and 4-gram features. The results of the study show that the models trained on 3-gram features have the best performance and that the performance "drops as n-gram value grows" (Baygin, 2018). Figure 2 below shows the change in performance as n increases.

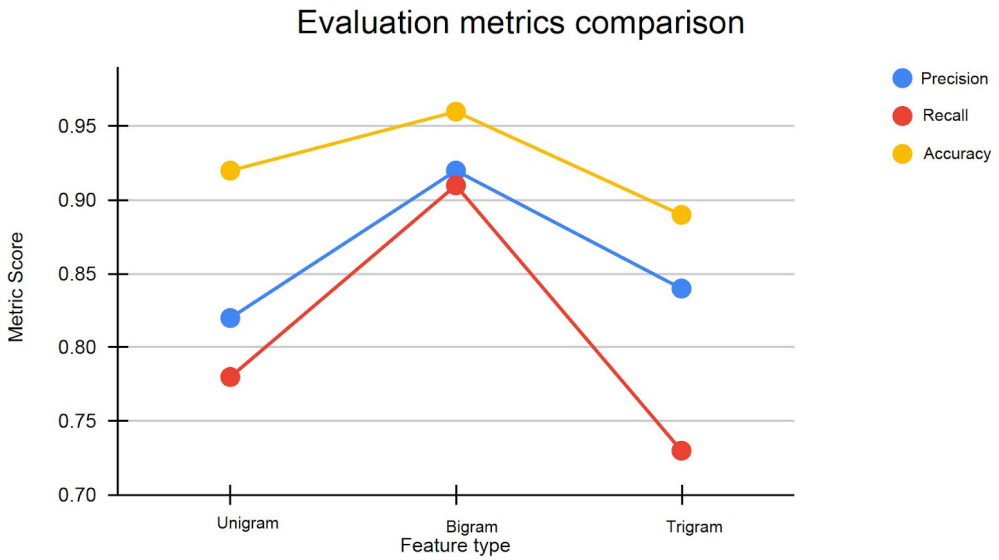

Figure 2: Evaluation metrics for models trained on each type of feature extracted

Experimentation with different n gram features is also done by Krouska, Troussas & Virvou (2016) in their work on twitter sentiment analysis. They extract unigram, bigram and 1-to3 gram features from the dataset. Different models are trained on each type of feature extracted and the results are compared to yet again investigate which type of feature extracted, when used for training a model, results in the best model performance. The results show that unigram and 1-to-3 grams performed better than bigram feature representation.

The above mentioned works show how different feature representations give out different performance for the models. Since the model performance depends on feature extraction (Krouska, Troussas & Virvou, 2016), it is vital to see what feature representation works best for the YouTube transcript dataset that is being used in this thesis. Since this analysis is not done before on the data set being used, this thesis aims to bridge this gap by providing valuable input on the type of features that can be used in order to get the best results. These features will be

extracted through the TF-IDF approach. The working of the TF-IDF approach is as discussed in the previous subsection.

3 -- Methods

The research goal is to develop a Machine Learning model which is able to determine whether a YouTube video is a conspiracy video or a non-conspiracy video based on it's transcript. This problem is tackled as a binary classification problem in this thesis. To tackle this problem, different classification algorithms (Support Vector Machine, Logistic Regression and Naive Bayes.) are trained on different types of features (unigram,bigram,trigram and 1-to-3 gram), which were extracted using the TF-IDF approach. The aim is to make the Youtube detection-mechanism more effective, so that these conspiratorial videos can be flagged more easily.

The performance of different classification models trained on features extracted from the unbalanced dataset, is compared. Finally, the data set is made balanced through SMOTE and the TF-IDF approach is used to extract features from the increased dataset. Different models are trained on the increased dataset to inspect if increasing the dataset, this way, improves performance.

4 -- Experimental setup

The below Figure shows the pipeline for the project, which is explained in the proceeding sections.

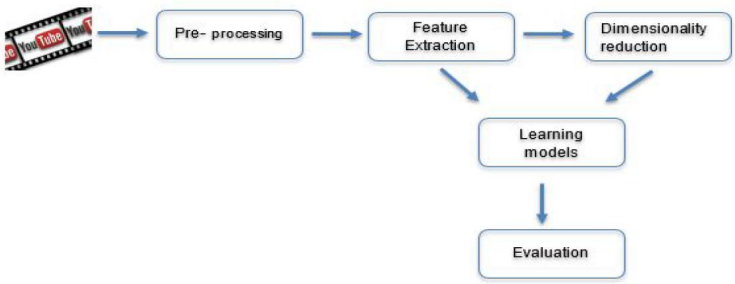

Figure 3: Pipeline for project

4.1 Dataset Description

Alfano, Fard, Carter, Clutton and Klein's (2020) initial data set had 600 videos. They had used a web crawler to get the 100 most recommended videos per topic on 6 topics: natural foods, martial arts, firearms, fitness, gurus and tiny houses. This resulted in a total of 600 videos. There are three labels in total: '1' are videos which have "no conspiratorial content", '2' are videos receiving a "mild conspiracy theory", and '3' are videos containing content with "severe conspiracy theory" (p.26). Out of these 600 videos, 480 videos had transcripts available to be processed and 120 were removed since their transcripts did not exist. Further, 101 videos turned out to be duplicates and hence were removed. As a result, there became 378 videos in total. In addition to these 378 transcripts, 50 more were added and ranked in the same way (van den Eijnden, p. 10-11). Out of these 50 videos 4 were removed because of being deleted by YouTube

later. As a result, there were a total of 424 videos in the data set to be worked with. The final data set is unbalanced as there are more non-conspiratorial videos; 335 as compared to conspiratorial videos; 89 videos (van den Eijnden, p.10-11).

This thesis contributes in increasing the size of the dataset. Two teams each consisting of 3 and 4 thesis-students respectively have worked to increase the dataset. A total of 59 videos have been added to the dataset. While increasing the data set, it is observable that conspiratorial videos are difficult to find on YouTube. Therefore, it has been decided that for video-searching purposes a broad topic such as 'corona' would be coupled with seed words such as 'china', 'america', 'vaccine', to make the search for the targeted videos narrower. These videos have been watched and labelled by three raters. The real challenge lay in distinguishing label 2 videos from label 3 videos. The final label has been selected based on majority voting i.e the label which two out of three raters selected. For videos where all three raters gave a different label, one of the three raters watched the video and labeled it again.

Finally, in order to balance the dataset SMOTE class from scikit-learn has been used. Initially there were 347 label 0 data points and 136 label 1 data points. After transforming the data through SMOTE, the number of label 1 data points has now increased to 347, making the data set balanced.

4.2 Preprocessing

The first step in the pipeline for this thesis is preprocessing. Preprocessing is an important step for data preparation (Krouska, Troussas & Virvou, 2016). As Kadhim (2018, p.23) illustrates, the goal of preprocessing is to convert a document, made of sentences, into a feature

vector (where each word is a feature). Firstly, the document is split into features. He indicates that these features are called “tokens, words, terms or attributes”(p.24). After tokenization the next step is to remove the features such as “stop words, numbers and special characters” (p.24). Furthermore, these features are “non-informative” as they do not provide any important information when it comes to classifying texts (p.23). Therefore, these words are removed in the preprocessing phase and results in lower dimensional data, decreasing both the time and speed-resources required in the processing phase (p.22).

For the purpose of this thesis the following preprocessing steps are taken:

1. **Removing punctuations and non-alpha numeric characters:**

   Punctuations and other non-alpha numeric characters are frequently used in almost all sentences. As Kowsari et al, (2019) illustrate, these characters can negatively affect the performance of the classifier. Therefore, all non-alpha numeric characters and punctuations were removed from the text.

2. **Lemmatization:**

   Lemmatization is the step where different words having a common root are converted to their root word. For example, the words “heard” and “hearing”, after lemmatization, will be converted to the common word “hear”. Since all words do not occur in their root form, this is done to reduce the feature set (Camacho-Collados & Pilehvar, 2018, section 2.2). However, as Manning, Raghavan & Schütze (2009) illustrate, lemmatization may increase recall and reduce precision.

3. **Removing Stop words :**

   There are several words such as “an”, “the” ,”it” etc. which are frequent in almost

all sentences, and therefore do not hold much importance in text classification (Kadhim, 2018, p.24, section 3.3). Therefore, such words will be removed in order to reduce the feature set dimensionality. In addition to the words in the stop words list provided by nltk package, the word "music" is added in order to remove it from the transcripts since it appears wherever there is background music in the video.

4. **Lower casing words:**

Lower casing although simple is yet an effective way to reduce the dimensionality of this thesis' feature set. This holds true because when words are lowercase, the same words are merged, resulting in a lower dimensional feature set (Symeonidis, Effrosynidis & Arampatzis, 2018, p.301, section 3.10). For example before lowercasing the words "Jump" and "jump" will be seen as different words. However, after lower casing, both these words will be seen as the same word. In-turn reducing the size of the feature set.

However, according to Manning, Raghavan & Schütze (2009) this affects precision and recall as a proper noun can be seen as a concrete noun after lowercasing, e.g "Bill" is seen as "bill".

4.3 Feature Extraction:

In this step, features which are most suitable for classifying the conspiratorial videos and non-conspiratorial videos are selected. This is done through the TF-IDF approach. Through the TF-IDF approach "core words" (most important features) are extracted" (Kim & Gil, 2019, p.8). This results in a drastic decrease in the size of the feature space. Hence, making the feature set more relevant to the classification task as well as making the feature set smaller.

The data set provided by Alfano, Fard, Carter, Clutton and Klein (2020) had 3 categories of videos: labelled ‘1’ are videos which have “no conspiratorial content”. Videos labelled ‘2’ are videos receiving a “mild conspiracy theory”, and lastly videos labelled ‘3’ are videos which contain content with ”severe conspiracy theory” (p.26). This data set is merged with an additional 59 videos which are gathered by 2 groups of thesis students. Each video, from these 59 videos, is labelled as one of the three aforementioned labels. Figure 4 shows the Label distribution for the complete data set of videos.

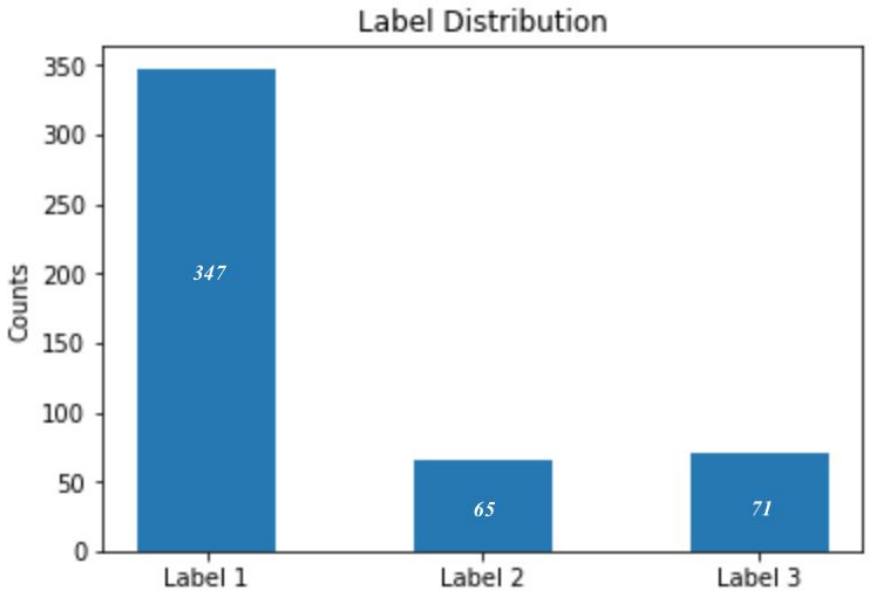

Figure 4: Label distribution

For the purpose of this thesis, all videos labelled as ‘1’ are relabelled to ‘0’: non-conspiratorial videos. All other videos labelled as ‘2’ and ‘3’ are merged and relabelled to ‘2’: conspiratorial videos. This is done in order to make the problem a binary classification problem. This has also been done in the thesis of Albers, making the results comparable to the

results of this thesis. Figure 5 below shows the label distribution for these videos after the relabelling is done.

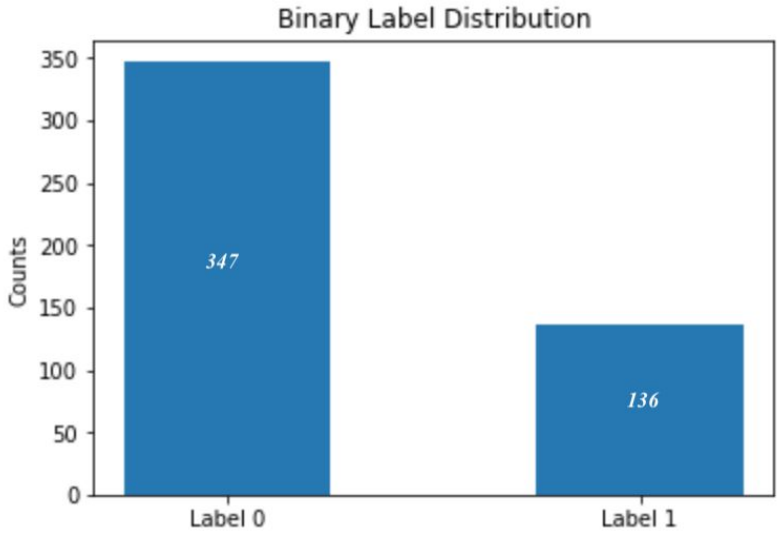

Figure 5: Binary label distribution.

The dataset that is used for this thesis consists of 483 videos with each video either labelled as “0’ non-conspiracy video or as “1” conspiracy video. After merging the datasets the transcripts of the videos are first tokenized to convert the documents into features. These transcripts also contain stop words (words contained in the nltk stopword collection). Furthermore, punctuations and non-alpha numeric characters are also removed in the preprocessing step in order to reduce the size of the feature set. After preprocessing the final feature set had a mean word count of 1802 per transcript, with the largest transcript having 17936 words.

4.4 Experimental Procedure and evaluation method

For this thesis all the experiments and code have been run on Google colaboratory using python 3.6.9 as the interpreter. To load the data in addition to the csv package, pickle package has also been used. For the purposes of visualisations, Python and Excel were used.

The first step taken in this thesis is preprocessing. Firstly, preprocessing is performed to clean the data resulting in a smaller feature set. For preprocessing one function had been created. This function uses the nltk package, which first removed all the punctuations and non-alpha numeric characters, as these have shown to negatively affect the model's performance (Kowsari et al., 2019). Secondly, the text has been lemmatized. Lemmatization reduces the size of the feature set (Camacho-Collados & Pilehvar, 2018) since it converts all the words to their root words, resulting in a smaller feature set. Thirdly, stop words have been removed and all the text has been made lower case. Similar to lemmatization, lower casing reduces the size of the feature set, resulting in a lower dimensional feature set (Symeonidis, Effrosynidis & Arampatzis, 2018, p.301). The resulting output of the function is used to extract the features for training and testing the classifiers.

After the preprocessing, features have been extracted using the TF-IDF approach. Through TF-IDF every word in a transcript is converted into a numeric value. This numeric value indicates the importance of that particular word to the transcript it appears in enabling extraction of the most important words (Kim & Gil, 2019, p.8). Different features are extracted through the TF-IDF method. This includes unigram models, bigram models, trigram models and 1-to-3 gram models (Krouska, Troussas & Virvou, 2016).

Since the individual words in a text are considered as features when dealing with a textual dataset, it can result in a high dimensional feature set. For the data set being used in this thesis it is seen that the total number of unique features in the data set are 33026. Using a large number of features for training and learning them perfectly, during training, can result in overfitting (Dietterich, 1995). Therefore, for the purpose of this thesis the maximum features of the feature set are set to 4000. This has been done for mainly three reasons. Firstly, to cater to the problem of overfitting. Secondly, to cater to the memory and computational limitation, since a smaller feature set requires lesser memory and results in faster computations. Baradaran & Golpar-Raboki (2019) show that compressing the number of features results in increased processing speed, catered to the memory limitations and improved the accuracy of the model. Lastly, even though in the thesis by Akyazi, the size of the feature set had been set to 5000, the feature set of size 4000 resulted in better performance.(For the comparison of all the results see Appendix A).The final resulting training set is split into training (80%) and testing (20%) data sets. The training model is used to train all the models and the test data set is used for the evaluation of the models' performance.

A total of three models, using their default parameters, are trained. This included Support Vector Machine, Naive Bayes and Logistic Regression. These models are trained on different features extracted through the TF-IDF approach and then are exposed to unseen data to be classified. In order to evaluate these models Confusion matrix, F1 score and recall are used. Even though accuracy matrix is a widely used evaluation matrix, for imbalanced data sets, it is misleading, because it gives more weight to the majority class/label (Bekkar, Kheliouane Djemaa and Akrouf Alitouche, 2013, p. 28). Instead, evaluation is done through Precision, recall and F1-scores as they take into account the imbalance of the dataset (Omary & Mtenzi, 2010).

Since the dataset being used is small and unbalanced, the dataset has been made balanced by synthetically generating data points for the minority class, which inturn increased the size of the dataset. This is done through SMOTE. Increasing the size of the dataset through SMOTE has shown to increase the performance of the model as seen from the experiments carried out by Fan & Qin (2018) and therefore in order to investigate this effect, the data set used in this thesis has been increased through SMOTE. For comparison purposes, the same features (unigram, bigram, tri gram and 1-to-3 gram models) are extracted from the increased dataset to train the same models (Support Vector Machine, Naive Bayes, Logistic Regression and Neural Networks). The matrices associated with the original and the increased datasets are reported and compared in the subsequent results section.

For all machine learning tasks, overfitting is seen as a major problem (Feng et al., 2017). Overfitting refers to a model fitting to the data with almost zero error (Bilbao & Bilbao, 2017). Since the goal of developing models is to maximize the predicting capability in the unseen test data, learning the training data perfectly would not help achieve this goal. Therefore, it is necessary to take steps to overcome this issue. To deal with this issue, in this thesis a subset (400 features) from the whole feature space was selected to train the models.

5 – Results

Table 1 shows the results for the models' performances on each type of feature extracted from the original data set (not synthetically increased). Additionally, Table 2 shows the results for the models' performances on each type of feature extracted from the synthetically increased data set. The classifiers have been trained with the Sklearn modules, using their default settings.

For the ease of the reader, the interesting results for both the datasets are separated into 2 tables: Table 1 and Table 2. All the results, for all the experiments are found in Appendix B (original data set) and Appendix C (Synthetically Increased Dataset).

Since the original dataset was small and had an unbalance in the class distribution, the dataset has been synthetically increased and made balanced using SMOTE. Using SMOTE to increase the dataset size can "improve the accuracy" of the classifiers being used (Chawla, Bowyer, Hall, & Kegelmeyer, 2002). As expected, using SMOTE to increase the size of the dataset used in this thesis results in better classification performance. The performance of the models, trained on unigram and bigram features, increases after synthetically increasing the dataset. However, surprisingly the performance of models trained on trigram and 1-to-3 gram features did not increase. It would be interesting to investigate why this was the case, however the direction taken by this thesis was different, otherwise it would have been embarked by it.

Overall, the models trained on unigram features perform better than the models trained on all the other features. This is the case for both the original dataset and the synthetically increased dataset, as it can be seen from the macro averaged F1 values in Table 1 and Table 2. Surprisingly, the models trained on 1-to 3 gram feature failed to perform better than the unigram and bi-gram models, which is contradictory to the results shown in the works of Krouska, Troussas & Virvou (2016) as it can be seen here that models trained on 1-to-3 gram features outperformed the models trained on bigram features.

Analysing the models shows that the Logistic Regression model trained on unigram features had the best performance on the original data set used in this thesis, with a macro averaged F1 score of *0.84*. From amongst the models trained on bigram features, Support Vector

Machine outperforms both the Logistic Regression model as well as the Naive Bayes model,with a Macro averaged F1 score of *0.82*. However, when looking at the results for the models trained on trigram and 1-to-3 gram features, Naive Bayes model performs better than the other two models, with an F1 Score of *0.61* on both types of features.

Examining the results on the synthetically increased dataset, it is seen that the Logistic Regression model has the best performance on all four types of features extracted (unigram, bigram, trigram and 1-to-3 gram). On the unigram features, Logistic Regression had a macro averaged F1 score of *0.88.* Similarly, on bigram features, Logistic regression performed better than the other two models with a macro averaged F1 score of *0.87.* Lastly, for Trigram and 1-to-3 gram features the model yet again had a better performance that the other two models with macro averaged F1 scores of *0.58* and *0.58* respectively. Logistic Regression model has also proved to outperform on other datasets as seen in the work by Pranckevičius & Marcinkevičius (2017) where logistic regression achieved the highest classification accuracy as compared to the other models. Table 1 and table 2 depict that, even though the models trained on unigram features and bigram features perform better when the data set is synthetically increased, it is seen that the performance for the models trained on trigram and 1-to-3 gram features decreases when the data is synthetically increased. Therefore, it is not necessary that the performance will be improved after synthetically increasing the dataset.

	Macro Average F1-Score			
	Uni-gram	**Bi-gram**	**Tri-gram**	**1-to-3 gram**
Logistic Regression	0.845	0.812	0.602	0.602
SVM	0.829	0.819	0.589	0.589
Naive Bayes	0.693	0.649	0.613	0.613

Table 1: Performance of models on features extracted from the original dataset.

	Macro Average F1-Score			
	Uni-gram	**Bi-gram**	**Tri-gram**	**1-to-3 gram**
Logistic Regression	0.879	0.874	0.578	0.578
SVM	0.875	0.849	0.563	0.563
Naive Bayes	0.803	0.747	0.522	0.522

Table 2: Performance of models on features extracted from the Synthetically increased dataset.

To conclude, after looking at the results from Appendix B and Appendix C, it would be safe to say that synthetically increasing the data set using SMOTE increases the performance of the models for unigram and bigram features. Furthermore, it is evident that overall the Logistic regression performed the best from amongst all three models on both the original dataset and the synthetically increased dataset. Lastly, it can be concluded that the models trained on unigram features outperform all models trained on other features.

6 – Discussion

The goal of this thesis is to compare the performance of different classifiers, trained on different features extracted from a small dataset. Subsequently, it compares the performance of the same classifiers trained on the same type of features extracted from a synthetically increased dataset. Even though it was expected that the models trained on 1-to-3 gram feature would result in a better performance as compared to models trained on other types of features, surprisingly, the results are contradictory to that of the work by Krouska, Troussas & Virvou (2016) where 1-to-3 gram feature models outperformed bigram feature models. It was seen that bigram features resulted in a better performance of the models as compared to models trained on trigram and 1-to-3 gram features.

Furthermore, it was expected that features extracted from the synthetically increased dataset would help improve the performance of the classifiers, as seen in the work by Chawla, Bowyer, Hall, & Kegelmeyer (2002), as compared to the classifiers trained on features extracted from the original dataset. This is indeed the case as synthetically increasing the dataset shows a visible improvement in the classification performance.

The results of the experiments carried out in this thesis show that unigram models perform better than the other types of features extracted. This is the case for both the original dataset as well as the synthetically increased dataset, as seen from Table 1 and Table 2. Looking at the results of the classifiers trained on both types of datasets, it is seen that indeed synthetically increasing the dataset improved the performance of the classifiers trained on unigram and bigram features. However, the performance of the models trained on trigram and

1-to-3 gram features did not increase. Furthermore, as discussed above in the results section, it is seen that, overall, the Logistic Regression model had the best performance. From amongst the models that were trained on features extracted from the synthetically increased dataset, Logistic Regression had the best performance for all types of features that were extracted. However, for the original dataset, Logistic regression had a better performance as compared to the other models when trained on unigram features. Despite these engrossing results, there are certain limitations that need to be taken under consideration since they affect the reliability of the results.

Even though the dataset used for this thesis is bigger than the dataset used by Akyazi (see Figure 6 below for comparison), in terms of big data it is still small, since it only consists of 483 data points. In addition to the data set being relatively small, the distribution of the classes in the dataset is imbalanced, with a total of 347 non-conspiratorial videos as compared to 136 conspiratorial videos as seen from Figure 5. This makes it interesting and thought-provoking to work with, since techniques to balance and counter the effect of the imbalance are needed to be considered. The overall results of the experiments carried out in this thesis show that Logistic Regression performs better than the other classifiers. For future works, it is suggested that research be conducted using a bigger and slightly balanced dataset, which might illustrate an increase in the performance and more robust results.

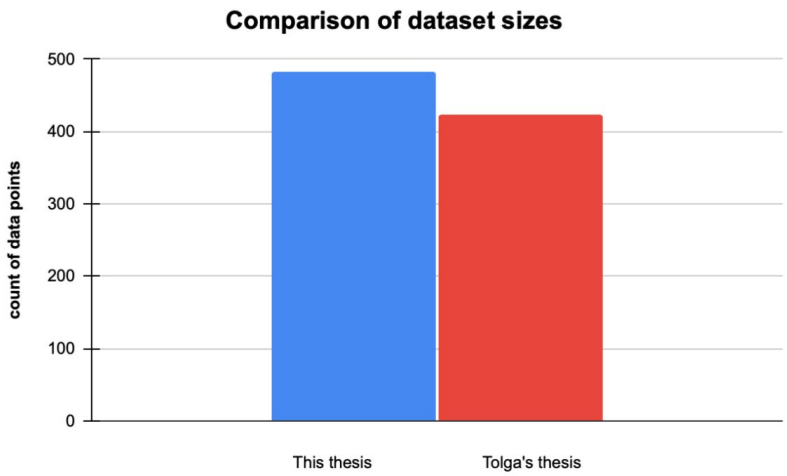

Figure 6: Comparison of dataset sizes between the two theses

Furthermore, even though SMOTE has proved effective in dealing with the imbalance dataset problem, it comes with some shortcomings. The SMOTE algorithm randomly generates a synthetic data point between two data points. The algorithm cannot tell if a point is an outlier or not. Therefore, this can result in synthetically generated outliers (Liang, Jiang, Li, Xue & Wang, 2020). Hence, using methods other than SMOTE, in order to deal with the imbalance dataset problem, shows potential of giving interesting results and are worth being explored.

Additionally, even though the TF-IDF approach is widely used in text mining problems and has shown impressive results, it is still not the best option to be used in text classification problems (Chen, Zhang, Long & Zhang, 2016, p.245). Attempts have been made to improve the TF-IDF method. As evident from the works of Fan & Qin (2018) it can be seen that the features extracted through the improved TF-IDF approaches indeed improve the performance of the classifier. Therefore, for future works to be done on this dataset, a comparison of TF-IDF approach with the improved versions of TF-IDF may prove interesting.

Figure 7 below shows the number of words in each transcript. The variability and the

high number of features shows how noisy the data is. The works of Wang, Wang & Chang (2016) show that noisy data has a negative effect on the performance of the classifiers. Chen, Sun, Bing & Yang (2017) further show that Recurrent Neural networks are better when working with noisy data. It is suggested that noise from the data be removed as much as possible. Furthermore, using Recurrent Neural Networks on this dataset might give interesting results, and may be worth being analyzed.

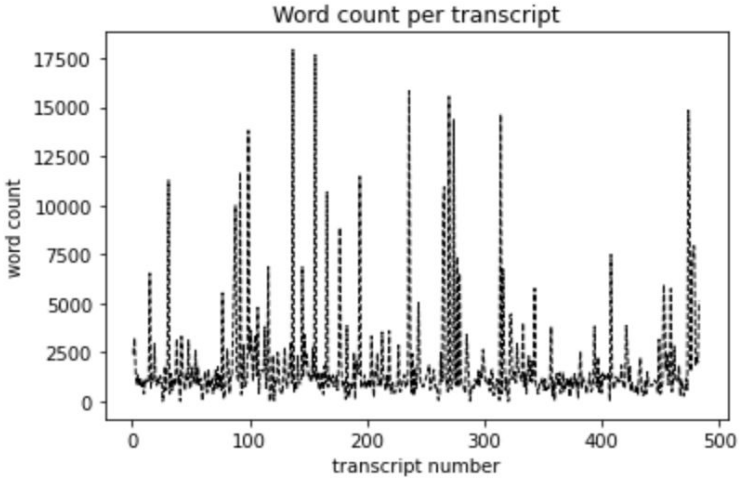

Figure 7: Word count per transcript

7 – Conclusion

The main objective of this study has been to firstly, compare the performance of several classifiers (Logistic regression, Support Vector machines and Naive Bayes) which were trained on different features (unigram, bigram,trigram, 1-to-3 gram) extracted from transcripts of videos on YouTube. Secondly, to investigate whether synthetically increasing a small, unbalanced dataset improves the classification performance of classifiers as compared to the classification performance of models trained on the original small and unbalanced dataset.

The research question of this study has been: How effectively do different Machine Learning models, trained on different features extracted from a small unbalanced dataset, classify conspiratorial Youtube video transcripts? The outcome of this research shows that firstly, models trained on unigram features, extracted from a small and unbalanced dataset, perform better than models trained on either of bigram, trigram and 1-to3 gram features. Secondly, adequate improvement in the performance of the classifiers trained on features extracted from synthetically increased dataset is seen, paving way for future experimentation in this area of research.

Even though flagging conspiratorial content on YouTube was the focus of this research, it would not be wrong to say that conspiracy theories are widespread throughout the social media. The effect of being exposed to such content can be detrimental as it may promote "significantly more extreme" ideas (H Raab, 2016). Furthermore, conspiratorial content emerges from a detection-mechanism that has not taken into account deliberate efforts by propagators to utilize the YouTube algorithm to consciously impact individuals and society, and hence needs monitoring. This calls for an action to be taken to mitigate this effect even if it is only focused on a single part of social media (YouTube), and that was the focus of this thesis. Although this mechanism might be able to help prevent only a single person rather than the whole society from being radicalized, it illustrates the need for further attempts to be taken and shows how this can be done, to stop the spread of conspiratorial content at large.

8 – Acknowledgements

I once again thank my supervisor, Michal Klincewicz for his guidance and help during the complete phase of this thesis, including his relentless efforts to motivate me. Furthermore, I highly appreciate Mark Alfano and his colleagues for sharing their research and dataset which formed the foundation for my thesis. Lastly, I thank my fellow thesis colleagues in working together to increase the dataset, as well as being there for any discussion whenever needed.

References

Abdyraeva, C. The Use of Cyberspace in the Context of Hybrid Warfare.: Means, Challenges and Trends (pp. 20-28, Rep.). OIIP - Austrian Institute for International Affairs. doi:10.2307/resrep25102.8

Akyazi, T. Classification of Conspiratorial Content on YouTube - Measuring Influence of Sentiment Weighting on Classification Performance (Graduate). Tilburg University Tilburg University, School of Humanities and Digital Sciences, Department of Cognitive Science & Artificial Intelligence.

Akram, W. and Kumar, R.. (2017). A Study on Positive and Negative Effects of Social Media on Society. International Journal of Computer Sciences and Engineering, 5(10), pp.351-354.

Albers, S. Detecting conspiratorial content using word vector models trained on Reddit conspiracy language for feature enrichment (Graduate). Tilburg University School of Humanities and Digital Sciences Department of Cognitive Science & Artificial Intelligence.

Alfano, M., Fard, A., Carter, J., Clutton, P., and Klein, C. (2020). Technologically scaffolded atypical cognition: the case of YouTube's recommender system. Synthese. doi: 10.1007/s11229-020-02724-x

Baradaran, R., & Golpar-Raboki, E. (2019). Feature Extraction and Efficiency Comparison Using Dimension Reduction Methods in Sentiment Analysis Context. Signal And Data Processing, 16(3), 88-79. doi: 10.29252/jsdp.16.3.88

Baygin, M. (2018). Classification of Text Documents based on Naive Bayes using N-Gram Features. 2018 International Conference On Artificial Intelligence And Data Processing (IDAP). doi: 10.1109/idap.2018.8620853

Bekkar, M., Kheliouane Djemaa, D., and Akrouf Alitouche, D. (2013). Evaluation Measures for Models Assessment over Imbalanced Data Sets. Algeria: ENSSEA, National School of Statistics and Applied Economics, Algiers,.

Biały, B. (2017). Social Media—From Social Exchange to Battlefield. *The Cyber Defense Review, 2(2)*, 69-90. Retrieved November 25, 2020, from http://www.jstor.org/stable/26267344

Bilbao, I., & Bilbao, J. (2017). Overfitting problem and the over-training in the era of data: Particularly for Artificial Neural Networks. 2017 Eighth International Conference On Intelligent Computing And Information Systems (ICICIS). doi: 10.1109/intelcis.2017.8260032

Branco, P., Torgo, L., & Ribeiro, R. (2015). A Survey of Predictive Modelling under Imbalanced Distributions. Cornell University.

Camacho-Collados, J., & Pilehvar, M. (2018). From Word To Sense Embeddings: A Survey on Vector Representations of Meaning. *Journal Of Artificial Intelligence Research, 63*, 743-788. doi: 10.1613/jair.1.11259

Chawla, N., Bowyer, K., Hall, L., & Kegelmeyer, W. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal Of Artificial Intelligence Research, 16*, 321-357. doi: 10.1613/jair.953

Chen, K., Zhang, Z., Long, J., & Zhang, H. (2016). Turning from TF-IDF to TF-IGM for term weighting in text classification. Expert Systems With Applications, 66, 245-260. doi: 10.1016/j.eswa.2016.09.009

Chen, P., Sun, Z., Bing, L., & Yang, W. (2017). Recurrent attention network on memory for aspect sentiment analysis. *Proceedings of the 2017 conference on empirical methods in natural language processing*. 452-461.

Christian, H., Agus, M., & Suhartono, D. (2016). Single Document Automatic Text Summarization using Term Frequency-Inverse Document Frequency (TF-IDF). Comtech: Computer, Mathematics And Engineering Applications, 7(4), 285. doi: 10.21512/comtech.v7i4.3746

Collins Uduiguomen, U., Celestine Agwi, U., and Faith Aliu, N. (2014). Social Networks: A Curse or a Blessing? (A Case Study of Selected Students from Auchi Polytechnic). Transactions On Networks And Communications, 2(4). doi: 10.14738/tnc.24.405

Della Vedova, M., Tacchini, E., Moret, S., Ballarin, G., DiPierro, M., & de Alfaro, L. (2018). Automatic Online Fake News Detection Combining Content and Social

Signals. 2018 22Nd Conference Of Open Innovations Association (FRUCT). doi: 10.23919/fruct.2018.8468301

Del Vicario, M., Bessi, A., Zollo, F., Petroni, F., Scala, A., Caldarelli, G., . . . Quattrociocchi, W. (2016). The spreading of misinformation online. *Proceedings of the National Academy of Sciences of the United States of America*, 113(3), 554-559. Retrieved November 25, 2020, from https://www.jstor.org/stable/26467425

Dietterich, T. (1995). Overfitting and under computing in machine learning. ACM computing surveys

Douglas, K., Sutton, R., and Cichocka, A. (2017). The Psychology of Conspiracy Theories. *Current Directions In Psychological Science, 26(6),* 538-542. doi: 10.1177/0963721417718261

Douzas, G., Bacao, F., & Last, F. (2018). Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. *Information Sciences, 465*, 1-20. doi: 10.1016/j.ins.2018.06.056

Fan, H., & Qin, Y. (2018). Research on Text Classification Based on Improved TF-IDF Algorithm. Proceedings Of The 2018 International Conference On Network, Communication, Computer Engineering (NCCE 2018). doi: 10.2991/ncce-18.2018.79

Feng, X., Liang, Y., Shi, X., Xu, D., Wang, X., & Guan, R. (2017). Overfitting Reduction of Text Classification Based on AdaBELM. Entropy, 19(7), 330. doi: 10.3390/e19070330

Fernandez, A., Garcia, S., Herrera, F., & Chawla, N. (2018). SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary. *Journal Of Artificial Intelligence Research, 61*, 863-905. doi: 10.1613/jair.1.11192

Gaydhani, A., Doma, V., Kendre, S., & Bhagwat, L. (2018). Detecting Hate Speech and Offensive Language on Twitter using Machine Learning: An N-gram and TFIDF based Approach. IEEE International Advance Computing Conference 2018. doi: 1809.08651v1

Ghosh, S., & Desarkar, M. (2018). Class Specific TF-IDF Boosting for Short-text Classification. Companion Of The Web Conference 2018 On The Web Conference 2018 - WWW '18. doi: 10.1145/3184558.3191621

Guo, H., Zhou, J., & Wu, C. (2018). Imbalanced Learning Based on Data-Partition and SMOTE. *Information*, *9(9)*, 238. doi: 10.3390/info9090238

Heap, B., Bain, M., Wobcke, W., Krzywicki, A., & Schmeidl, S. (2017). Word Vector Enrichment of Low Frequency Words in the Bag-of-Words Model for Short Text Multi-class Classification Problems. School of Computer Science and Engineering and School of Social Sciences, The University of New South Wales Sydney NSW 2052, Australia.

H Raab, M. (2016). Manipulation, Exaggeration and Conspiracy. Experimental Approaches to a Better Understanding of the Belief in Conspiracy Theories. (Ph.D). Otto-Friedrich-Universität Bamberg.

Kadhim, A. (2018). An Evaluation of Preprocessing Techniques for Text Classification. *International Journal Of Computer Science And Information Security (IJCSIS), 16(6).*

Kim, D., Seo, D., Cho, S. & Kang, P. (2019). Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec. Information sciences, 477, s. 15–29. doi:10.1016/j.ins.2018.10.006

Kim, S.-W. & Gil, J.-M. (2019). Research paper classification systems based on TF-IDF and LDA schemes. Human-centric computing and information sciences, 9 (1). doi:10.1186/s13673-019-0192-7

Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information, 10(4),* 150.

Krouska, A., Troussas, C., & Virvou, M. (2016). The effect of preprocessing techniques on Twitter sentiment analysis. *7th International Conference On Information, Intelligence, Systems & Applications* (IISA). doi: 10.1109/IISA.2016.7785373

Lemaˆıtre, G., Nogueira, F., & Aridas, C. (2017). Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning. *Journal Of Machine Learning Research, 18,* 1-5.

Liang, H., Sun, X., Sun, Y., & Gao, Y. (2017). Text feature extraction based on deep learning: a review. EURASIP Journal On Wireless Communications And Networking, 2017(1). doi: 10.1186/s13638-017-0993-1

Liang, X., Jiang, A., Li, T., Xue, Y., & Wang, G. (2020). LR-SMOTE — An improved unbalanced data set oversampling based on K-means and SVM. Knowledge-Based Systems, 196, 105845. doi: 10.1016/j.knosys.2020.105845

Manning, C., Raghavan, P., & Schütze, H. (2009). An Introduction to Information Retrieval. Cambridge: Cambridge University Press.

Omary, Z., & Mtenzi, F. (2010). Machine learning approach to identifying the dataset threshold for the performance estimators in supervised learning. I*nternational Journal for Infonomics (IJI), 3(3).*

Popat, K., Mukherjee, S., Yates, A., & Weikum, G. (2018). DeClarE: Debunking Fake News and False Claims using Evidence-Aware Deep Learning. Proceedings Of The 2018 Conference On Empirical Methods In Natural Language Processing. doi: 10.18653/v1/d18-1003

Pranckevičius, T., & Marcinkevičius, V. (2017). Comparison of Naive Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Reviews Classification. *Baltic Journal Of Modern Computing, 5(2)*. doi: 10.22364/bjmc.2017.5.2.05

Prati R. C., Batista G. E., & Monard M. C. (2009). Data mining with imbalanced class distributions: concepts and methods. *Indian International Conference Artificial Intelligence,* 59–376.

Rantapelkonen, J., & Jaitner, M. (2014). YouTube Wars: In and Out of Control. Journal of Information Warfare, 13(1), 1-18. Retrieved November 25, 2020, from https://www.jstor.org/stable/26487007

Sammons, M., Christodoulopoulos, C., Kordjamshidi, P., Khashabi, D., Srikumar, V., Vijayakumar, P., Bokhari, M., Wu, X., & Roth, D. (2016). Edison: Feature extraction for NLP, simplified. In N. Calzolari, K. Choukri, H. Mazo, A. Moreno, T. Declerck, S. Goggi, M. Grobelnik, J. Odijk, S. Piperidis, B. Maegaard, & J. Mariani (Eds.), *Proceedings of the 10th International Conference on Language Resources and Evaluation, LREC 2016* (pp. 4085-4092). (Proceedings of the 10th

International Conference on Language Resources and Evaluation, LREC 2016). European Language Resources Association (ELRA).

Serrano, J., Papakyriakopoulos, O., & Hegelich, S. (2020). NLP-based Feature Extraction for the Detection of COVID-19 Misinformation Videos on YouTube. Technical University Of Munich, Germany. Retrieved from https://www.aclweb.org/anthology/2020.nlpcovid19-acl.17.pdf

Spencer, W. (2011). Conspiracy Rhetoric in Cicero's Verrines. *Illinois Classical Studies, (35-36)*, 121-141. doi:10.5406/illiclasstud.35-36.0121

Sunstein, C. R. (2014). Conspiracy theories and other dangerous ideas: Simon and Schuster.

Symeonidis, S., Effrosynidis, D., & Arampatzis, A. (2018). A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. *Expert Systems With Applications, 110*, 298-310. doi: 10.1016/j.eswa.2018.06.022

Tan, C. (2018). Influence of the technological features. In Regulating Content on Social Media: Copyright, Terms of Service and Technological Features (pp. 137-164). London: UCL Press. Retrieved November 25, 2020, from http://www.jstor.org/stable/j.ctt2250v4k.9

Timberg, C., Dwoskin, E., Romm, T., and Ba Tran, A. (2018). The Washington Post, p. Two years after #Pizzagate showed the dangers of hateful conspiracies, they're still rampant on YouTube. Retrieved from https://www.washingtonpost.com/business/technology/hateful-conspiracies-thrive-on-youtube-despite-pledge-to-clean-up-problematic-videos/2018/12/10/625730a8-f3f8-11e8-9240-e8028a62c722_story.html

Uscinski, J., Klofstad, C., & Atkinson, M. (2016). What Drives Conspiratorial Beliefs? The Role of Informational Cues and Predispositions. *Political Research Quarterly, 69(1),* 57-71. Retrieved November 25, 2020, from http://www.jstor.org/stable/44018529

van den Eijnden, R. A Keywords-base Approach to Conspiracy Video Classification (Msc.). School of Humanities and Digital Sciences, Tilburg University.

van Prooijen, J., & Douglas, K. (2017). Conspiracy theories as part of history: The role of societal crisis situations. Memory Studies, 10(3), 323-333. doi: 10.1177/1750698017701615

Wah, Y., Rahman, H., He, H., & Bulgiba, A. (2016). Handling imbalanced dataset using SVM and k-NN approach. *Advances In Industrial And Applied Mathematics*. doi: 10.1063/1.4954536

Wang, L., Wang, Y., & Chang, Q. (2016). Feature selection methods for big data bioinformatics: A survey from the search perspective. *Methods, 111,* 21-31.

Wakabayashi, D. (2019). YouTube Moves to Make Conspiracy Videos Harder to Find. The New York Times.

YouTube Revenue and Usage Statistics (2019). (2019). Retrieved from https://www.businessofapps.com/ data/youtube-statistics/

Appendix A:

Original Dataset

	Macro Average F1-Score (Feature set size 5000)				**Macro Average F1-Score (Feature set size 4000)**			
	Unigram	**Bigram**	**Trigram**	**1-to-3 gram**	**Unigram**	**Bigram**	**Trigram**	**1-to-3 gram**
Logistic Regression	0.787	0.721	0.487	0.487	0.845	0.812	0.602	0.602
SVM	0.795	0.699	0.461	0.461	0.829	0.819	0.589	0.589
Naive Bayes	0.684	0.688	0.619	0.619	0.693	0.649	0.613	0.613

Synthetically increased dataset

	Macro Average F1-Score (Feature set size 5000)				**Macro Average F1-Score (Feature set size 4000)**			
	Unigram	**Bigram**	**Trigram**	**1-to-3 gram**	**Unigram**	**Bigram**	**Trigram**	**1-to-3 gram**
Logistic Regression	0.863	0.777	0.537	0.537	0.880	0.874	0.578	0.578
SVM	0.843	0.781	0.494	0.494	0.875	0.849	0.563	0.563

Naive Bayes	0.837	0.788	0.712	0.712	0.803	0.747	0.522	0.522

Appendix B:

	Precision(0)	Recall(0)	F1-Score(0)	Precision(1)	Recall(1)	F1-Score(1)	Macro averaged F1-Score
Results on original Dataset							
Logistic Regression (uni-gram model)	0.906	0.923	0.914	0.795	0.756	0.775	0.845
Logistic Regression (bi-gram model)	0.862	0.962	0.909	0.862	0.610	0.714	0.812
Logistic Regression (trigram model)	0.761	0.981	0.857	0.818	0.220	0.346	0.602
Logistic Regression (1-to-3 gram model)	0.761	0.981	0.857	0.818	0.220	0.346	0.602

SVM (uni-gram model)	0.883	0.942	0.912	0.824	0.683	0.747	0.829
SVM (bi-gram model)	0.863	0.971	0.914	0.893	0.610	0.725	0.819
SVM(trigram model)	0.757	0.990	0.858	0.889	0.195	0.320	0.589
SVM(1-to-3 gram model)	0.757	0.990	0.858	0.889	0.195	0.320	0.589

Naive Bayes (uni-gram model)	0.802	0.933	0.862	0.708	0.415	0.523	0.693
Naive Bayes (bi-gram model)	0.780	0.952	0.857	0.722	0.317	0.441	0.649
Naive Bayes (trigram model)	0.765	0.971	0.856	0.769	0.244	0.370	0.613
Naive Bayes (1-to-3 gram model)	0.765	0.971	0.856	0.769	0.244	0.370	0.613

Appendix C:

	Precision(0)	Recall(0)	F1-Score(0)	Precision(1)	Recall(1)	F1-Score(1)	Macro averaged F1-Score
Results on Increased Dataset							
Logistic Regression (uni-gram model)	0.850	0.923	0.885	0.916	0.837	0.874	0.880
Logistic Regression (bi-gram model)	0.820	0.962	0.885	0.954	0.789	0.863	0.874
Logistic Regression (trigram model)	0.576	0.981	0.726	0.936	0.279	0.430	0.578
Logistic Regression (1-to-3 gram model)	0.576	0.981	0.726	0.936	0.279	0.430	0.578

SVM (uni-gram model)	0.842	0.923	0.881	0.915	0.827	0.869	0.875
SVM (bi-gram model)	0.787	0.962	0.866	0.951	0.740	0.832	0.849
SVM(trigram model)	0.570	0.981	0.721	0.931	0.260	0.406	0.563
SVM(1-to-3 gram model)	0.570	0.981	0.721	0.931	0.260	0.406	0.563

Naive Bayes (uni-gram model)	0.735	0.962	0.833	0.944	0.654	0.773	0.803
Naive Bayes (bi-gram model)	0.680	0.981	0.803	0.966	0.539	0.691	0.747
Naive Bayes (trigram model)	0.554	0.990	0.710	0.955	0.202	0.333	0.522
Naive Bayes (1-to-3 gram model)	0.554	0.990	0.710	0.955	0.202	0.333	0.522