ECMM 7051 Research Project Report

PSYCHOGRAPHIC PROFILING OF CHARITABLE DONATIONS USING TWITTER DATA AND MACHINE LEARNING TECHNIQUES

By

Carlos Jose Calix Woc

Supervisor Colin Conrad

Submitted in partial fulfilment of the requirements
for the degree of Master of Electronic Commerce

at

Dalhousie University
Halifax, Nova Scotia
March 2020

DALHOUSIE UNIVERSITY

DATE: March 12th, 2020

AUTHOR: Carlos Jose Calix Woc

TITLE: Psychographic Profiling on Charitable Donations Using Twitter Data and Machine Learning Techniques

DEPARTMENT: Faculties of Computer Science, Management, and Law

DEGREE: MEC – Master of Electronic Commerce

_____

Signature of Author

To my parents and sisters, who have made this journey possible and for putting up with me in the times I dreaded being in this freezing hell. For those who are still dreading the cold winters of this abominated place, may this be a proof that creativity and greatness can flow out of you when you let go of the distractions that other places may offer.

Table of Contents

List of Figures

List of Tables

List of Abbreviations Used

CSS	Cascading Style Sheets
CSV	Comma-Separated Values
HTML	Hypertext Markup Language
JSON	JavaScript Object Notation
MNB	Multinomial Naïve Bayes
NLP	Natural Language Processing
SaaS	Software as a Service
SVM	Support Vector Machine
TF-IDF	Term Frequency-Inverse Document Frequency
URL	Uniform Resource Locator

Abstract

This paper describes several content-based approaches to identifying charity donors using machine learning and natural language processing techniques on Twitter data, including profile descriptions and tweets. The explored users belong to a client base from Fundmetric Inc., a donor engagement-focused SaaS company. By evaluating a series of experiments on various combinations of datasets, classification tasks, and vectorizing techniques, this project was able to conclude that machine learning can indeed be extrapolated to the field of charitable donations. Using Multinomial Naïve Bayes, a machine learning model capable of distinguishing donors from ordinary Twitter users attained an accuracy score of 71%. The results of such a project can inspire users in the field of charitable donations to leverage machine learning to optimize resources and apply customer segmentation to create more user-centric experiences for their clients.

Chapter 1

Introduction

The role of social media has increased in importance over the past years as it continues to provide users with a vast number of benefits. From an individual perspective, it has created a platform where users can establish social interactions, microblog and share information in real-time [1]. From a corporate perspective, companies can engage their customers more efficiently by creating more personalized customer experiences. Such strategies ensure the right content is being served to the right person at the right time, thus optimizing overall resources. These personalized experiences are a direct result of the large volumes of data available through these platforms.

According to recent analyses, Twitter has become the social media platform with the third-highest market share along with Reddit and following Facebook [3]. It serves as a platform for users to generate their own content and follow information regarding events, trends, movements, etc. Users generate content in the form of "tweets", Twitter's proprietary messages that have a limit of 280 characters [2]. This voluminous amount of user-generated text data is one of the major sources of information in the era of big data. By applying data mining and machine learning techniques combined with natural language processing techniques on this data, companies can train algorithms that learn consumer patterns and behaviors to ultimately create a more robust customer taxonomy. Identifying patterns on user's words and writing styles have led to valuable insights for customer-centric experiences and ultimately expanded the opportunities for monetization.

1.1 Research Question

The objective of this research is to create a data-driven approach to classifying individuals in the field of charitable donations. The initiative stems out of a partnership between Dalhousie

University and Fundmetric Inc., a Canadian SaaS company that helps charities increase donor engagement through data-driven initiatives [4]. This research project leverages publicly available data from both Fundmetric clients and ordinary Twitter users and serves a twofold purpose. First and foremost, the use of applied machine learning techniques will assist Fundmetric in identifying new customers, who are prone to donate to charities. Several algorithms will evaluate users' social media activity and assess psychographic similarities with the existing Fundmetric customer base. Second, many research initiatives have leveraged the publicly available data on Twitter to create more customer-centric experiences through machine learning and content-based approaches. However, to the best of my knowledge, only statistical-based approaches have explored the domain of charitable donations for user classification [5][6], none have explored content-based approaches for psychographic profiling purposes. The research question to be investigated in this research project is the following:

RQ Do individuals who have previously donated to charities exhibit distinctive social media behavior compared to ordinary social media users?

I believe the answer to this research question will set a foundation for more robust research in the field of charitable donations. Customer segmentation in this domain will allow companies such as Fundmetric to optimize their resources to not only engage their clients in a more customized manner but also devise creative ways to target customers who have not previously donated to charities.

Chapter 2

Background and Related Work

The literature explored for this research project has been segmented into two main categories. The first section will explore data mining, machine learning, and NLP techniques and algorithms. The literature will clarify their technicalities and how they were applied to answering the research question. The second section will explore applied machine learning research in different domains, including politics, events and customer demographics from both statistical and content-based approaches. It will demonstrate that content-based approaches can indeed be extrapolated to psychographic profiling in the field of charitable donations.

2.1 Algorithms and Techniques

Machine learning is a discipline that combines computer science and statistics. It focuses on the development of algorithms that allow computers to perform tasks without a human having to explicitly program it. Algorithms rather learn based on patterns found in the data used to construct them [7]. The techniques used to answer the research question will be limited to the branch of supervised learning. Supervised learning is the branch of machine learning that applies algorithms to analyze input variables and learns their patterns in order to map an instance to an output or predefined label [7]. It is said to be supervised because the model is given user-provided labels that the researcher wants the algorithm to predict. Algorithms are not self-guided as it is in the case of unsupervised learning. The supervised learning algorithms that will be explored pertain to classification tasks, where the goal is to classify new instances based on given attributes that the model has previously been trained on. The robustness of the model is later assessed based on the accuracy of labels classified correctly using a testing dataset.

For this research, at three different classification methods were evaluated, because no algorithm that fits perfectly for a particular problem. First, among the most used classification algorithms is the Multinomial Naïve Bayes classifier, which is based on the Bayes theorem [7]. MNB estimates the probability of an instance belonging to a certain class using attributes related to the class. The formula for the classification algorithm is based on the calculation of the conditional probability of a label given the attributes (Figure 1). A key aspect of this algorithm is the assumption of conditional independence among its instances. Essentially, all the features in *x* are independent of one another, when estimating their conditional probability to document *y*. This assumption in combination with the easiness and speed to implement the algorithm has made MNB one of the most common and highly effective classifiers used for text-based tasks [8]. Research by Frank and Bouckaert proved the efficiency and high results achieved through the application of this algorithm on text-based data. Their research focused on successfully classifying news data according to topical categories [9].

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

Figure 1: Multinomial Naïve Bayes Formula from [7]

Second, another commonly used classification algorithm is logistic regression. It is based on the linear regression model; however, it is concerned with estimating the likelihood of categorical variables [7]. It is a probabilistic model that provides the probability of a class belonging to a certain label. Each attribute is given a weighted sum that translates to a probability that contributes to determining the class of an instance. Results are not modeled in a straight line as in the case of linear regression, are reduced to a probability space between the interval of 0 and 1 through a sigmoid function [7]. Brzezinski and Knafl have validated the use of logistic regression for concept and document classification in their research [13].

$$\text{logistic}(\eta) = \frac{1}{1 + exp(-\eta)}$$

Figure 2: Logistic Regression Formula from [11]

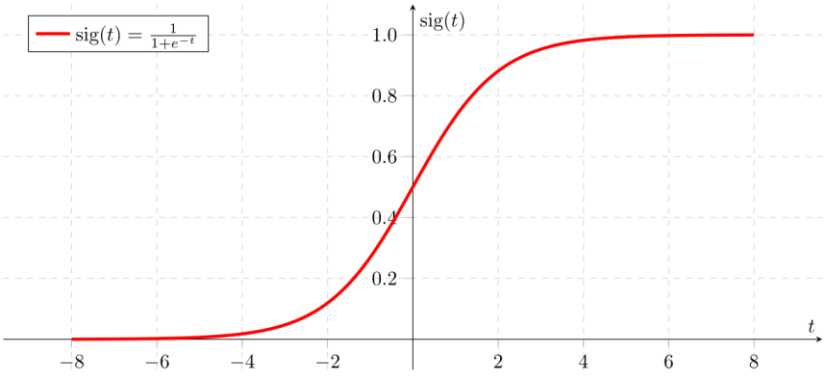


Figure 3: Sigmoid Activation Function from [10]

Lastly, support vector machines are another algorithm that can be used for classification problems. The algorithm consists of creating a hyperplane that divides instances in a binary classification. The shape of this hyperplane is defined by a parameter named kernel and is defined by the user. Moving forward, we will assume this hyperplane is a linear representation. The optimal hyperplane is created by estimating the highest margin between the two different classes. Depending on which side of the hyperplane boundary an instance may find itself, it is classified accordingly as one class or the other [14]. Research by Salamah and Ramayanti demonstrated the successful application of support vector machines to test the classification of complaints and non-complaints to enhance service quality in an organization [12].

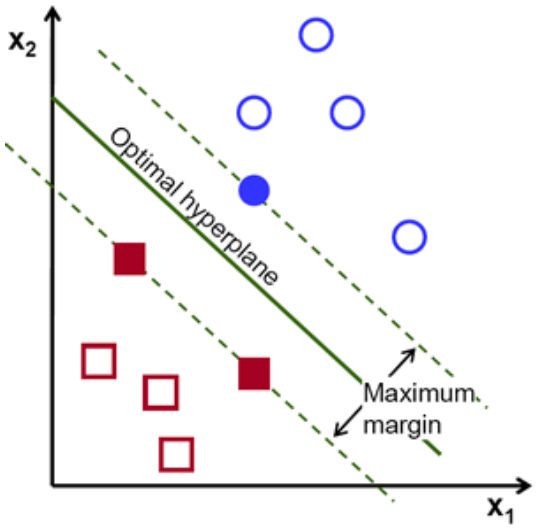


Figure 4: SVM Hyperplane Example from [15]

Prior to applying any classification algorithm to a dataset, the data must be cleaned and preprocessed to ensure it meets the right format. In the case of unstructured text data from social media platforms, it must first be structured to adhere to the format of a data model or schema. Fortunately, natural language processing is the discipline that focuses on making sense of text data and equips users with the tools to structure this data. Input variables must be transformed into a numeric format for machine learning algorithms to process them [16]. To acquire a structured format, text data must first undergo a process of tokenization. Tokenization consists of creating individual text components from the provided text data. The text is broken into smaller parts. These can be individual sentences, words or even characters [16].

One of the most commonly used tokenization methods is n-grams. N-grams are sequences of n-items retrieved from a piece of text usually in the form of words or characters [17]. Tripathy et al. applied the use word n-grams, particularly unigrams, bigrams and trigrams models to classify sentiment on movie reviews [18]. Other research by Wieting et al. explored word embeddings via character n-gram rather than word n-grams and demonstrated this technique outperformed that of state-of-the-art methods such as convolutional neural networks [19]. Both word and character n-grams perform differently according to each scenario, thus it is recommendable to evaluate both formats.

Once the words have been tokenized, features engineering processes need to be applied for machine learning algorithms to process input variables. Two methods will be explored. The first feature extraction process is the bag of words (count vectorizer) model [16]. This simple model takes all the text data and vectorizes each word found in the data. Each vector represents the frequency of each unique word found in the vector space.

Despite its simplicity, the bag of words model only takes into account the absolute frequency of each word. As a result, words that may appear often will outweigh words that appear less frequently but may play a crucial role in the text document. The Term Frequency-Inverse Document Frequency model is the second model that will be explored and aims to solve the problem posed by the bag of words model [16]. By taking the term frequency obtained from the bag of words model and multiplying it with the inverse document frequency, a weight is assigned to each vector. The inverse document frequency determines the relevance of each term and is calculated by obtaining dividing all documents over the term frequency and applying a logarithmic function. Das and Chakraborty applied TF-IDF to product and movie review classification and yielded higher accuracy results than those of the bag of words model [21]. However, this does not necessarily imply that TF-IDF will surpass the bag of words model in model robustness.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right)$$

Figure 5: TF-IDF Formula from [19]

2.2 Machine Learning Applications

The application of machine learning techniques in the field of charitable donations has been previously explored using statistical-based approaches. An example of this work is the research done by Farrokhvar et al [6]. They evaluated demographic factors such as household income, unemployment rate, sex, age, ethnicity, and education level among different

geographical areas in the United States and discovered that education level, previous charity giving and amount of population in a particular area had a significant positive impact in the local charity giving. While their research demonstrates that machine learning techniques can be applied to the field of charitable donations, it is difficult to translate these insights into potential monetization strategies. Companies leveraging this approach would only target population groups at a very generic level as opposed to adopting a user-specific approach. This model fails to encapsulate the dynamics and preferences of the population. Companies must create user-centric experiences to avoid overusing resources.

Liang also applied machine learning techniques to assess the likelihood of identifying donors for high school fundraising based on donation amounts combined with demographic traits from past donors and rejectors dataset [5]. Similar to the research by Farrokhvar et al. [6], this study is limited due to the population sampling approach. This experiment only focused on donations from past alumni, families or relatives of alumni, thus extrapolating the model to other types of donors has not been contemplated. Even though further work can be done to make these models more applicable for industry work purposes, these studies have set a foundation for machine learning in the field of charitable donations.

Twitter is a social media platform that provides a vast amount of user-generated content that has allowed researchers to jumpstart many initiatives. The variety and volume of data provided on this platform have given rise to different approaches to solving problems regarding user classification and ultimately creating more user-centric experiences. The first branch of classification is grounded on statistical-based approaches, where the focus lies on exploring features pertaining to user metadata, such as timestamps, number of followers, number of posts, interactions among users, etc. The research by Daouadi et al., [22] focused on mining Twitter users to classify between individuals and organizations and showed the success of leveraging a statistical-based approach rather than using text data. While they affirm that a content-based approach proves a better alternative than a statistical-based approach due to the demand of high computational power and language dynamics, the success of their results lies primarily on the fact that the behavior between individuals and organizations have different uses for their social media accounts. The former focuses on

microblogging and sharing information with friends, as opposed to the latter, which has a more commercial focus. Naturally, their online behavior varies significantly.

Not all statistical-based approaches exploring user profile metadata have yielded relevant results. Such is the case of the experiment by Rao et al. [23]. They explored users' network structure and communication behavior variables, such as retweet frequency and tweet frequency, to evaluate their impact in classifying users based on several latent variables including age, gender, and political orientation. However, they observed these variables have no significant impact on their classification. The recurred to the use of text data instead, which yielded higher results and a more robust model for user classification purposes.

Given the combined results from the research by Rao et al., Daouadi et al. and recommendations by Pennachiotti and Popescu on leveraging profile features and tweeting behavior [24], content-based approaches need to be explored. Content-based approaches focus on structuring text data coming from text-based sources, such as tweets and profile descriptions, through natural language processing techniques and have demonstrated major results.

The analysis of text data has been applied for user classification across many disciplines. The vast majority of existing literature solely leverages the analysis of tweets. Such is the case of the research by Preoţiuc-Pietro et al. They studied political ideologies where users identified themselves depending on the extremity of their political views [27]. Using logistic regression, they yielded a higher accuracy when classifying tweets by users with more extreme views. Moreover, they discovered that words used in tweets varied among different user groups. For instance, conservative groups made use of religious vocabulary more recurrently in their tweets than liberal groups. Even though the highest accuracy score was approximately 78%, alternative classification and data structure methods could have been explored to test the robustness of the dataset. Unigrams and logistic regressions were the only methods considered for this model. Extended research has explored alternative data sources and techniques. For instance, Kim et al. and Zubiaga et al. have combined the use of tweets and user profile descriptions for identifying users' countries of origins and e-cigarettes users respectively [25][26]. Both pieces of researches found that combining both data sources

led to higher accuracy results than looking at them individually. Another field that has leveraged the use of tweets for classification purposes is the domain of event identification. Stowe et al. analyzed tweets published through hurricane periods to successfully identify evacuation behavior; thus, proving the value found from text analysis [28]. These experiments are evidence that content-based approaches to machine learning can also be extrapolated to the field of charitable donations.

The application of machine learning algorithms has revolutionized how businesses approach their daily processes across different industries and domains. In the domain of finance, Paypal has used classification algorithms for fraud detection purposes, thus reducing the number of false-positive cases and mitigating costs [29]. In the domain of retail, Under Armour uses the data collected from their fitness apps to create customer segments that allow the company to create customized fitness routines, thus catering to each to the needs of each unique customer segment [30]. In the domain of car manufacturing, Mazda also leveraged machine learning models to optimize marketing spend. By analyzing social media posts from marketing influencers that have yielded a high return on investment, Mazda is able to identify new marketers with similar attributes thus reducing costs and increasing revenue [31]. At the foundation of these innovative strategies lies supervised learning models. Machine learning-based strategies are necessary to ensure new monetization and resource optimization opportunities.

Although no research regarding user classification through content-based approaches on charitable donations has been done in the past, the evaluated methods prove that such classification can be extrapolated to a variety of domains. By identifying unique linguistic characteristics on users who have donated to charities and comparing them to users who have not donated, machine learning techniques can be applied and allow companies to create more user-centric experiences. The result will allow charities to optimize resources by focusing their efforts on the right people at the right time.

Chapter 3

Theory and Methodology

The focus of this chapter is to synthesize the theory behind the algorithms, techniques, and technologies necessary to build this proof of concept. It will uncover the intricacies of how they function to assist the reader in understanding the implementation of each technology. As evidenced by the provided literature review, user classification through machine learning and NLP-based techniques can be extrapolated to distinct domains, not limited to that of charitable donations. To best address the research question, different experiments were run using different datasets and NLP tokenization methods:

E1 Classifying between donors and non-donors within Fundmetric users using word unigrams on profile descriptions

E2 Classifying between Fundmetric users and a randomized sample of Twitter users using word unigrams on profile descriptions

E3 Classifying between Fundmetric users and a randomized sample of Twitter users using word and character n-grams on profile descriptions

E4 Classifying between Fundmetric users and a randomized sample of Twitter users using word and character n-grams on tweets

This chapter will explore two types of technologies. The first group will elaborate on the technologies used to construct the visual aesthetics of the website prototype. The second group will dive into the intricacies of the backend technologies to construct the machine learning model and retrieve the necessary data to build it. Finally, this chapter will explore the necessary preprocessing steps to clean and format the data in the right structure and the final datasets. The website prototype is a combination of all these steps, where the algorithm that yielded the highest accuracy was selected and implemented in the prototype. The prototype is capable of assessing and classifying new instances introduced by a user.

3.1 Website Development

The construction of a website prototype has two distinct facets. First, it is composed of a user interface that is created using frontend technologies. This is rendered on a web browser on the client-server. Hypertext Markup Language 5 is the latest version of the markup language that was used to create the prototype. The prototype is made of pages that are rendered on the World Wide Web [32]. HTML5 uses a different range of tags and attributes to divide a page's content such as text and images into containers, thus giving users the ability to segment a document according to the project's needs. The main use of HTML5 is to provide structure to a website.

Following the structure definition of the prototype, styling was applied through the fourth release of Cascading Style Sheets. CSS4 modifies the different aesthetic elements in the prototype, such as the colors, fonts, and buttons [33]. Unobtrusive design is one of the challenges faced when applying aesthetics to an HTML document. Users own devices of different dimensionality and as a result, we must recur to responsive website design. Elements within an HTML document must be adjusted according to each device. While this strategy is necessary for adequate user experience and flow, this proof of concept will only consider desktop devices, as the main goal is to demonstrate the functionality of the machine learning model, rather than showcasing visual aesthetics.

External code frameworks allow software programmers to reduce the amount of code that needs to be written because these include functionalities and design templates that can be applied to the different elements. The Bootstrap 4 library is an open-sourced CSS framework that has been used to easily incorporate aesthetic elements into the prototype [34]. By applying the concept of element inheritance software programmers can override changes on elements that have already been predefined through this framework, thus allowing for customization according to the prototype requirements.

The second aspect to the website prototype is the backend technologies, which are responsible for handling the application logic of the website prototype. Python is an open-sourced server-side language that was used to create the logic behind the prototype [35]. It is

also the language in which the microframework Flask was made. Flask was incorporated into the project because it serves as a template engine that creates a local server to run the application [36]. Not only is it able to render the document, but it can incorporate Python functionality to run the machine learning model within the application.

3.2 Data and Machine Learning

To create the machine learning model that was incorporated using Flask, a series of Python libraries were imported. Libraries are a collection of associated function code that is stored in files, intending to reduce the volume of required code and thus facilitate the construction of applications. The first imported library was NumPy, which stands for Numerical Python [37]. It allowed the data to be structured in arrays and matrices. These formats are required in order to apply machine learning techniques to the data. Pandas is another imported library that is used to structure the data easily. Even though NumPy is able to handle multidimensional data and complex data structures, Pandas structures the data in a two-dimensional table-like format that is easier to handle for users [38]. These structures called data frames represent the data in a format based on columns and rows, similar to that of a relational database. The nature of the two-dimensional data of the project allowed the data to be structured as a data frame. Third, scikit-learn was imported, because it contains a vast range of functions pertaining to machine learning, such as classification algorithms and preprocessing techniques [39]. It also provides functions to assess the robustness of the models. Fourth, the Natural Language Toolkit NLTK library was imported to process text data and structure to a format processable by the machine learning models [40]. Some of the techniques applied include lemmatizing the words, Finally, the library pickle was used to serialize the machine learning model [41]. Training the model on the website prototype for new instances would result in inefficient use of resources. Consequently, pickle saves the previously trained model and loads it to the website prototype. Pickle allows files to be transferred across Transmission Control Protocol, thus allowing multiple users to use and work on one machine learning model, without the need to retrain the model. Lastly, the Tweepy library was imported to establish a connection to Twitter's API. It allowed scraping the API for data on profile descriptions and tweets [42].

Table 1: Summary of Technologies Used

Tool	Developers	Description
HTML 5	Hoy [32]	A markup language for structuring website documents.
CSS 4	Wium Lie [33]	A styling language for modifying the aesthetic elements of HTML documents.
Bootstrap 4	Otto and Thornton [34]	An open-sourced CSS framework with predefined aesthetic elements.
Python	van Rossum [35]	An open-sourced server-side language for creating the logic behind the prototype.
Flask	Grinsberg [36]	A Python microframework for developing web applications.
NumPy	van der Walt et al. [37]	A library for arrays and matrices calculations.
Pandas	Mckinney [38]	A library for manipulating and analyzing data.
Scikit-learn	Pedregosa et al. [39]	A machine learning library.
NLTK	Loper and Bird [40]	A natural language processing library.
Pickle	Pickle Library [41]	A library for serializing objects.
Tweepy	Roesslein and Hill [42]	A library for accessing Twitter API.

3.3 Dataset and Data Retrieval

The data utilized for this research was provided by Fundmetric through a CSV file shared with the Dalhousie team. It pertains to a client organization that has approximately 98,000 individual donors. The dataset includes demographics features such as client age and location client engagement details such as donation type, donation frequency, etc. Although the different features provided great details on user behavior, the nature of a linguistic-based research did not require the use of all the present variables. For this research, I have focused only on two variables within this dataset:

V1 URL – A string variable that consists of the donor's Twitter URL profile (e.g. https://twitter.com/username123)

V2 Maximum Donation – A numeric variable that represents the maximum monetary donation that a client has given in their lifetime.

Prior to starting this linguistic-based approach study, a dataset had to be constructed. The URL variable was the key feature to expand the dataset for NLP purposes. Using personal research credentials, a connection to Twitter's API was established in order to begin retrieving the data. By trimming Twitter's domain from each individual URL, the username was used as an input parameter in the GET_USER function that was provided by the Tweepy library. The first run of data retrieval focused on obtaining profile descriptions only. The subsequent steps for data retrieval were programmed in a loop to improve task efficiency. Twitter is limited in the number of requests that a user is allowed to make to the API. Consequently, once this limit has been reached, users must wait for a short limit of 15 minutes prior to continuing any request calls. The data retrieval process began by verifying this limit. If this rate limit had not been reached, the algorithm verifies whether the input username existed. Even if an input username is valid, a profile description is not a required feature. Consequently, users without a description would be omitted and no data would be stored.

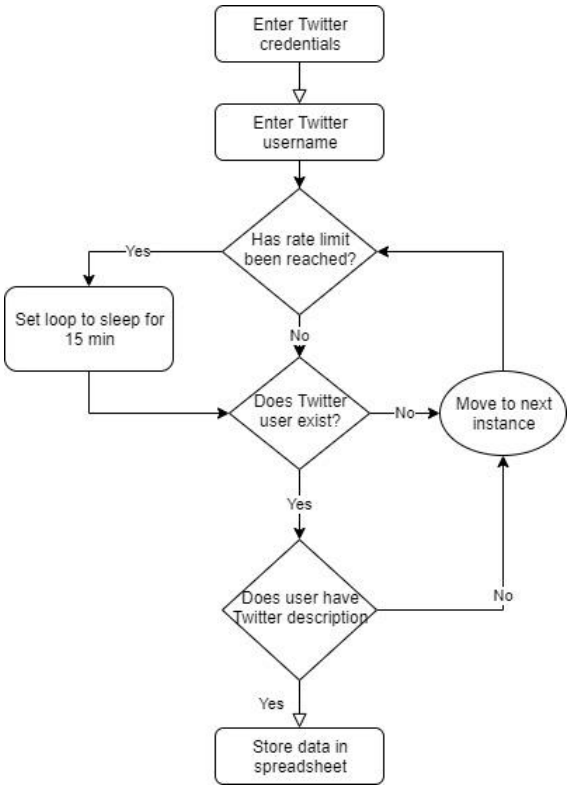


Figure 6: Profile Description Data Retrieval

The data retrieval for tweets followed similar procedures to that of profile descriptions and only varied in minor ways. First, the GET_USER_TIMELINE function was applied instead of the GET_USER function, which would return entries on the user's most recent activities. Second, the extracted was more voluminous when compared to the profile descriptions dataset, because the algorithm retrieved the five most recent tweets of each user.

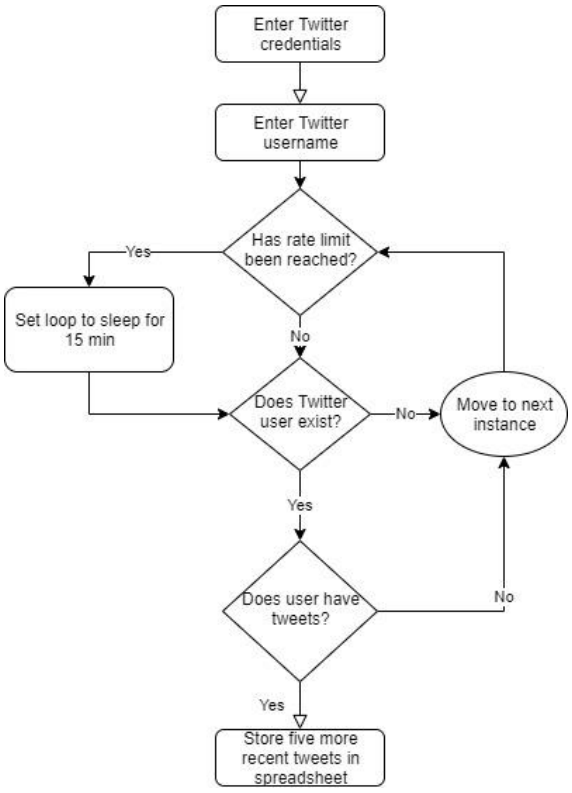


Figure 7: Tweets Data Retrieval

In order to run the classification tasks on E2, E3, and E4, a dataset using ordinary Twitter users had to be created, given that the goal for these experiments is to identify distinctive traits among the charitable donors. To reduce the lowest amount of bias possible, a random number generator was used to create numeric ids, which in turn would be used to call on to the Twitter API and retrieve both profile descriptions and most recent tweets on that particular id. This approach was not perfect as many of the generated ids that were passed on to Twitter API did not exist. Consequently, this process was repeated until an adequate amount of data was collected. Additionally, certain users did not pertain to English-speaking communities and were removed from the dataset.

Once the request call had been processed, Twitter API returned various amounts of data pertaining to the introduced username such as followers count, following users count, location, etc. The returned data came in the JSON format; however, for ease of manipulation and access purposes, it was transformed into a dictionary format. Dictionaries are data types

that store data in a key-value pair format. Users can access a particular data value by entering the respective key name [39]. The algorithm collected the user id, description, and tweets and stored this information in a dictionary of lists. Subsequently, it was transformed into a Pandas data frame because its two-dimensional structure allows the data to be saved as a CSV file.

3.4 Data Normalization

The results achieved by machine learning algorithms highly depend on the quality of the data used to train them. Data must first undergo a process of normalization and cleaning prior to processing it through the machine learning algorithms. The highly unstructured nature of text corpora requires different approaches when compared to regular numeric data. The major challenge is transforming unstructured data into a structured format.

The journey of data preprocessing begins with the data tokenization. As mentioned previously, the text data is segmented into a series of individual words, special characters, and punctuation that compose sentences. Experiments E1 and E2 underwent a word tokenization process. E3 and E4 were tokenized using both word and character n-grams using unigram, bigram, and trigrams.

Following the data segmentation, cleaning techniques were applied to standardize the data. Sarkar outlines several steps that were applied across these experiments [16]. First, the use of special characters and punctuation, such as exclamation and question marks, largely varies as they modify the message a speaker may convey. However, as individual tokens, they do not provide any significant meaning and consequently had to be removed from the data.

Second, computers are not able to distinguish between uppercase and lowercase text. For instance, the word “House” and the word “house” may hold the semantic meaning to a person. However, computers would process them differently due to syntactic differences. As a result, all words were lowercased to avoid “duplicating” word count.

Third, stopwords were removed from the text corpora. Stopwords are commonly used words that do not provide significant meaning such as articles and personal pronouns. For instance, the article “the” is highly present in any text document and due to its high frequency,

it would not only provide noise to the data, but it would also affect the robustness of techniques such as the bag of words vectorizer. This would attribute more importance to high-frequency words, which in actuality do not provide much value.

Finally, the concept of stemming was applied to the data. Words are composed of stems, the base form each word. Our vocabulary has been expanded by taking these stems and adding letters both prior and after the base stem. These are labeled prefixes and suffixes and their use had led to word inflections that creates nouns and different verb conjugations. Take the example of Figure 8, the word “consign” can be inflected by adding the suffixes “ed”, “ing”, “ment”, thus creating three new words. Nonetheless, they all share the same root form “consign” and have the same meaning. The Porter Stemmer algorithm was selected for all experiments as it can be applied to English-based data [16]. From a high-level perspective, it evaluates the combination of vowels and consonants and matches them against a predefined combination of letters that are removed to extract the stem word.

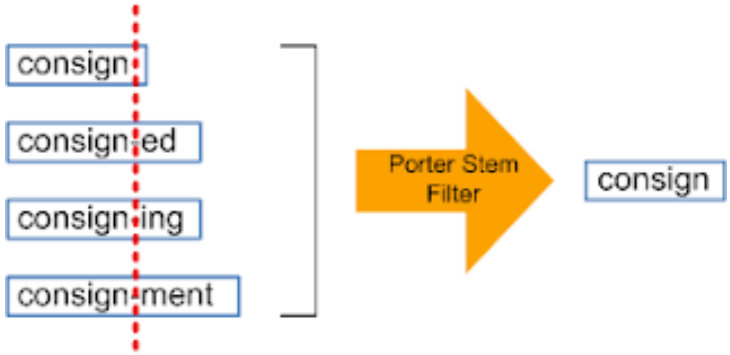


Figure 8: Stemming Example from [43]

3.4.1 Transforming Categorical Variables

The distinctive trait of supervised learning is the requirement of labeled instances for classification purposes. The labels are predefined by users based on existing information on the data or through data transformations. As a result, each experiment required the definition of user types.

Experiment 1 served as a proof of concept to understand the distinctive features of the Fundmetric dataset. The main goal was to evaluate potential differences within client donors within a particular charity. No feature describing lifetime donation was provided and as a result, distinguishing high-amount donors against low amount donors was not a feasible task. However, by leveraging the maximum donation variable, I was able to segment users as either donors or non-donors. As confirmed by domain experts from Fundmetric Inc. users with no maximum donation value are users who have not donated to charities in the past. However, these users were included in the client database, because they were considered as potential donors. The summary statistics were calculated using the Pandas library and it gave more insight into the overall data distribution. The maximum donation variable was heavily skewed towards the left side as approximately 47% of the instances have a value of zero. The presence of outliers may skew the data and as a result, they were removed from the dataset. In essence, any instance that found itself with a value of three standard deviations above the mean were removed.

Mean	Minimum	Maximum	Standard Deviation
29.46	0	49152	125.70

Table 2: Summary Statistics for Maximum Donation

Figures 9 and 10 show the overall data distribution. This confirms the skewness towards the left side, with the strong presence of outliers.

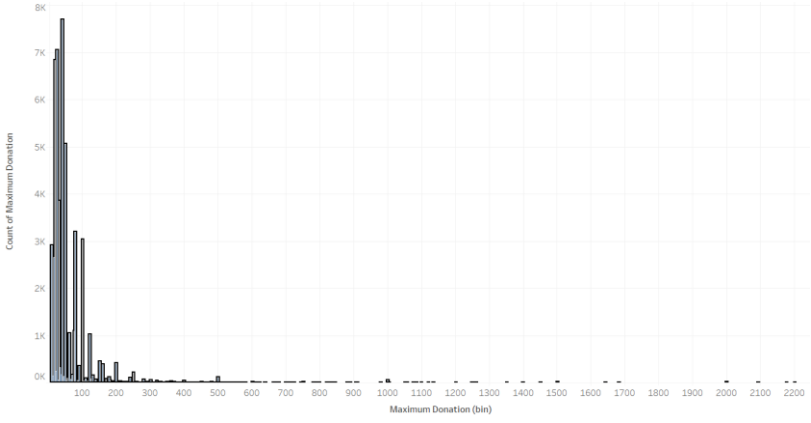


Figure 9: Distribution for Maximum Donation

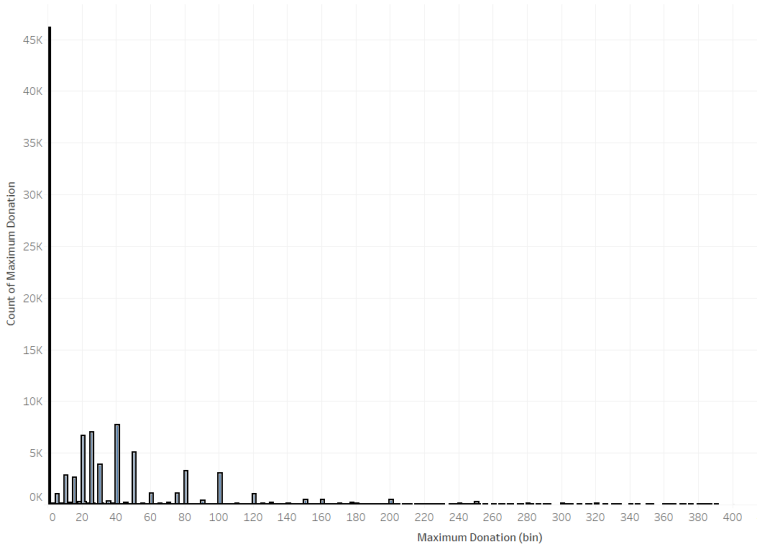


Figure 10: Distribution for Maximum Donation After Removing Outliers

After the removal of outliers, I proceeded to create two different populations. The first population was classified as “non-donors”. Instances under this group had a value of zero in their maximum donation attribute. The second population was classified as “donors”. This group was the inverse of the initial population. It was composed of instances that had a value greater than zero in the maximum donation attribute.

Experiment 2, 3 and 4 did not require existing variables in the dataset in order to define user type. Rather, user labeling was defined using the source of the data. Users pertaining to the Fundmetric dataset were labeled as “donors” as these individuals have shown past experiences of donating to charities or at least have demonstrated similar qualities to those of donors. Users, whose information was retrieved through the random id generator, were labeled as “random”.

Machine learning algorithms cannot process categorical variables, so these labels were transformed into a numeric format. This was achieved by applying integer encoding, where each categorical variable had a binary value. For instance, “donors” were labeled as 0 and “non-donors” as 1.

3.4.2 Final Datasets

A total of three datasets were created in order to test the proposed experiments. The first dataset was used to run E1 and pertained only to Fundmetric users' profile descriptions. To decrease the processing time and use of computational power, the dataset was reduced to a sample of 10,000 instances and includes only users' descriptions.

The second dataset pertained to user descriptions as well; however, the volume of instances was reduced to a much lower number to quicken processing time. To prevent any bias that may come from manual selection, a randomized sample of 2000 Fundmetric clients was extracted using the sample function in Python. This extract was combined subsequently with the random sample of users retrieved using the generated ids. Prior to this merge of data, both datasets were labeled respectively according to their data sources. The dataset had a total of 4000 instances.

The last dataset pertained to the same users found in the second dataset; however, instead of using profiles descriptions, the dataset included the five most recent tweets for each instance. As a result, the dataset expanded to a total of 20,000 instances.

A summary of the datasets used for each individual experiment is listed in Table 3

Dataset Summary				
Set Name	Text Type	Labels	Experiments	Instances
D1	Description	Donors, Non-Donors	E1	10,000
D2	Description	Fundmetric Users, Random	E2	4,000
D3	Tweets	Fundmetric Users, Random	E3, E4	20,000

Table 3: Dataset Summary

Chapter 4

Experimental Design

This chapter focuses on detailing the steps applied to establish the website prototype and the machine learning model that engines the site. Additionally, it will synthetize the logic of testing new instances out.

4.1 Machine Learning Model

After each dataset was normalized through the steps mentioned previously, vectorization was applied. Both Count Vectorizer and TF-IDF were applied to both experiments to compare overall performance between the two techniques. The concept of n-grams was not applied until experiments E3 and E4, because these were considered refinement techniques to the proof of concept validated in experiments E1 and E2. Both character and word n-grams techniques were tested using unigrams, bigrams, and trigrams.

Table 4 outlines the techniques used for each experiment, while Table 5 details the vectors dimensionality according to each n-gram type.

Vectorizing Summary			
Experiments	Techniques	N-gram Types	N-gram
E1	Countvectorizer, TF-IDF	Word	Unigrams
E2	Countvectorizer, TF-IDF	Word	Unigrams
E3	Countvectorizer, TF-IDF	Word, Character	Unigram - Trigram
E4	Countvectorizer, TF-IDF	Word, Character	Unigram - Trigram

Table 4: Experiment Vectorization Summary

Data Summary					
Text Type	Tokenization	Dataset	Users	Rows	Columns
Descriptions	Word Unigram	D1	10,000	10,000	20,023
Descriptions	Word Unigram	D2	4,000	4,000	8,729
Descriptions	Word Bigram	D2	4,000	4,000	22,616
Descriptions	Word Trigram	D2	4,000	4,000	24,222
Descriptions	Character Unigram	D2	4,000	4,000	226
Descriptions	Character Bigram	D2	4,000	4,000	3,445
Descriptions	Character Trigram	D2	4,000	4,000	15,978
Tweets	Word Unigram	D3	4,000	20,000	58,919
Tweets	Word Bigram	D3	4,000	20,000	190,455
Tweets	Word Trigram	D3	4,000	20,000	246,741
Tweets	Character Unigram	D3	4,000	20,000	1,770
Tweets	Character Bigram	D3	4,000	20,000	9,983
Tweets	Character Trigram	D3	4,000	20,000	73,542

Table 5: Data Vectors Summary

After applying vectorizing techniques, the dataset was split into two sets. The first set, labeled as the training set, was used by the machine learning algorithm to find distinctive patterns and traits between the provided labels. 60% of the data was reserved for training purposes, while the remaining portion being used as a testing set. The testing set is used to evaluate the performance and robustness of the model. The model first analyzes the testing set and tries to label them based on the identified patterns in the features used during the training phase. Once it has finished classifying all instances, the model matches them against their respective predefined labels, thus assessing the accuracy of the model. The model with the highest accuracy is to be selected as the engine behind the website prototype.

Depending on the magnitude of the training dataset, creating the machine learning model may require a substantial amount of time. Applications need to be user-centered and efficient, thus training a new machine learning model each time a user inputs a new instance would consume many resources. The library pickle was used to save the model as an SAV file, which would later be imported into the website prototype. This library allowed the model to be trained offline rather than in real-time.

4.2 Website Prototype

The website prototype is an amalgamation of different technologies. As mentioned previously, HTML 5 was utilized to structure the prototype, which is composed of two HTML documents. The first document labeled “index.html” is the main page, where users would first access the prototype. An input field is provided for users to submit a Twitter username. Once the machine learning model processes the text data pertaining to the new instance, the results are displayed on a second document labeled “output.html”. The visual aesthetics such as the website menu were defined using Bootstrap 4. None of the links are functional; however, they serve the purpose of showing a potential product layout by a company. The application is rendered on a Flask document labeled “home.py”.

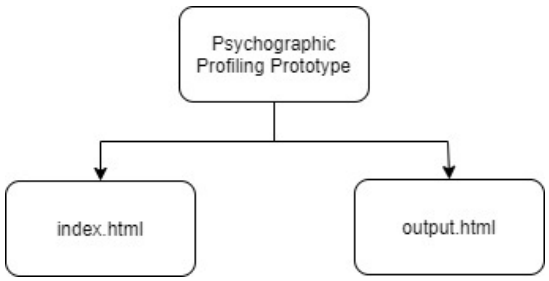


Figure 11: Prototype Diagram

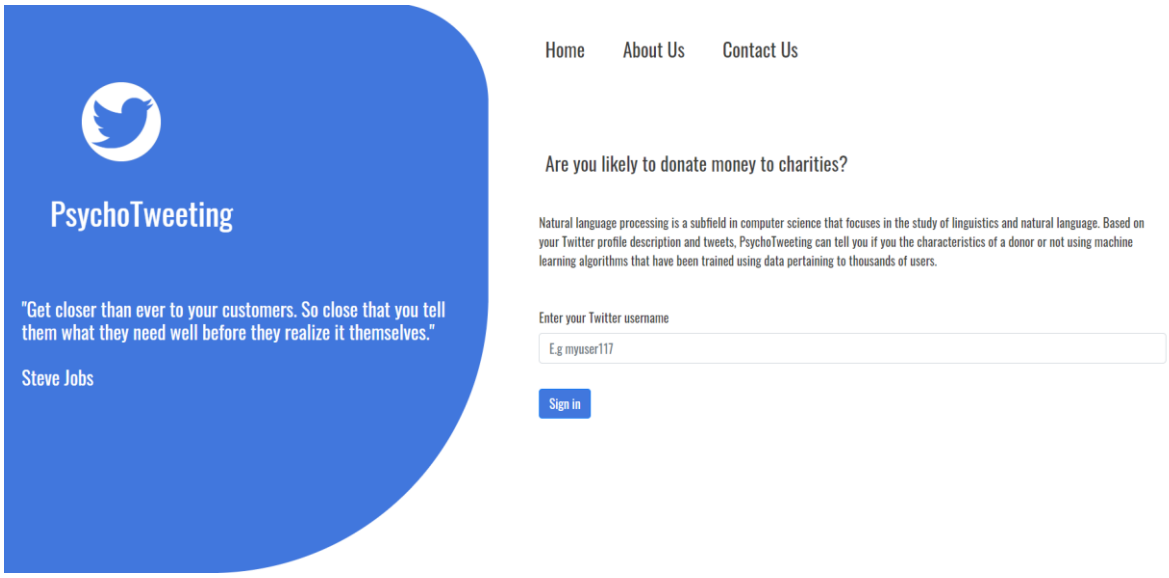


Figure 12: Index HTML Document

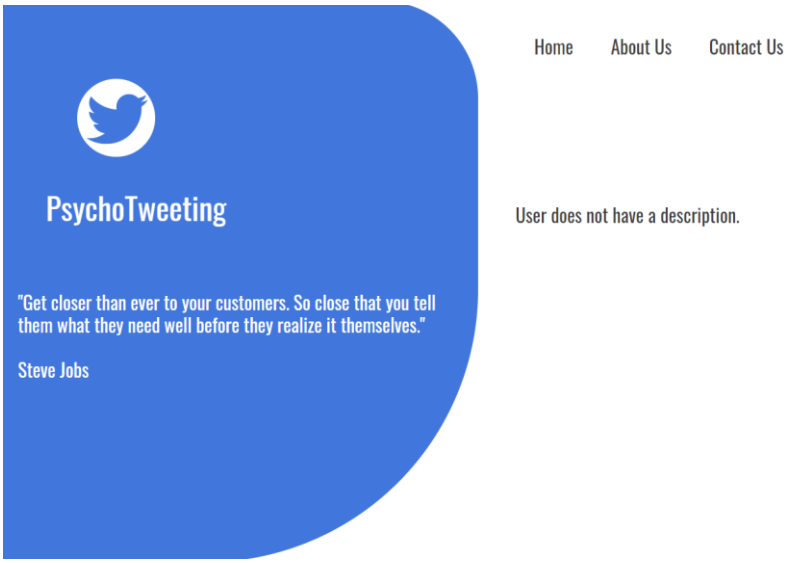


Figure 13: Output HTML Document - Invalid User

Figure 14: Output HTML Document - Classified User

4.3 Prototype Logic

The value of machine learning models is derived from users applying them to new instances. Once the user introduces a new instance to the model through the provided text field, several actions take place on the Flask document labeled “home.py”. First, the application uses personal Twitter credentials in order to make call requests to the Twitter API. Once the credentials have been validated, it verifies whether the provided username is valid. In the case of invalid data, the application notifies the user of this error. Contrarily, the application accesses the username’s Twitter profile to retrieve its profile description. If the profile data is not available, the user is notified of this and prompts them to test a new username. As mentioned, training a new model in real-time hinders user experience. Consequently, the model with the most optimal performance is loaded using the pickle library. Machine learning models are only able to test new instances that share the dimensionality as the data used to train and construct the model. Tweets and profile descriptions are dynamic as new words might appear that were not used previously to train the model. Fortunately, by saving the vocabulary used to feed the model through a serialized pickle file, the vocabulary structure can be loaded and applied to each new instance. The instance is structured according

to the vocabulary size and subsequently passed to the machine learning model. Based on the patterns identified by the model in the offline training phase, it classifies the introduced user as one class or another.

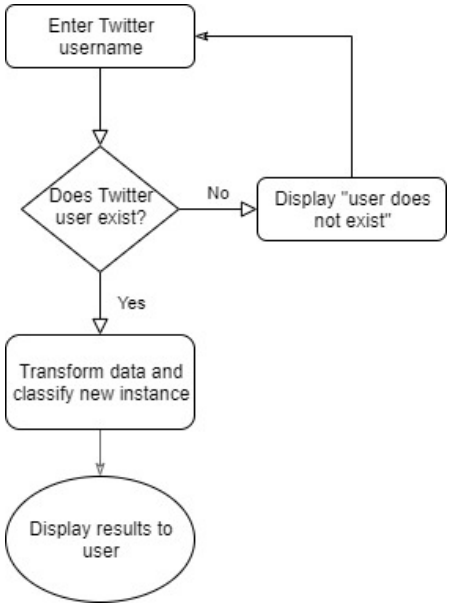


Figure 15: Application Logic Flow

Chapter 5

Results and Discussion

This chapter focuses on discussing and exploring the findings derived from the different experiments run. The robustness of each model was determined using the accuracy metric, the capability of a model to correctly classify an instance posterior to its training phase. Finally, the F1-score metric will be evaluated only on the highest accuracy performing algorithms. Additionally, wordclouds were applied in the data exploration phase. Wordclouds are a type of data visualization, where words are grouped together. The size of each word is determined by its frequency within a set of documents. Thus, larger words will have a larger size and vice versa. The results of these experiments validate our initial research question and demonstrate that users can indeed be classified based on their social media activity.

5.1 Donors and Non-Donors

E1 is the only experiment that focused on finding unique differences among the Fundmetric dataset. From Table 4, the use of TF-IDF surpassed the accuracy of the Count Vectorizer in all cases. The highest accuracy attained was a total of 57.1% using MNB with a TF-IDF vectorizer technique; however, the results are not robust. They do not differ substantially from that of random guessing as the error rates in all cases are close to 50%.

Experiment 1 Results				
Classifier	Accuracy	Vectorizer Technique	Data Type	N-Gram
SVM	54.4%	Count Vectorizer	Profile Description	Unigram
Multinomial NB	56.7%	Count Vectorizer	Profile Description	Unigram
Logistic Regression	54.8%	Count Vectorizer	Profile Description	Unigram
SVM	56.2%	TF-IDF	Profile Description	Unigram
Multinomial NB	57.1%	TF-IDF	Profile Description	Unigram
Logistic Regression	56.7%	TF-IDF	Profile Description	Unigram

Table 6: Experiment 1 Results

5.2 Fundmetric Clients and Random Users

In light of the underperformance of the models in E1, E2 explored a different dataset to evaluate differences between Fundmetric clients and random Twitter users. I observed a substantial improvement across the board. The average accuracy for E2 was 13.5% higher than that for E1. Similar to E1, the use of TF-IDF outperformed the count vectorizer technique. The highest accuracy achieved was using MNB with approximately 71%. These insights confirmed our research hypothesis and validated that donors indeed behave differently from regular users in social media channels.

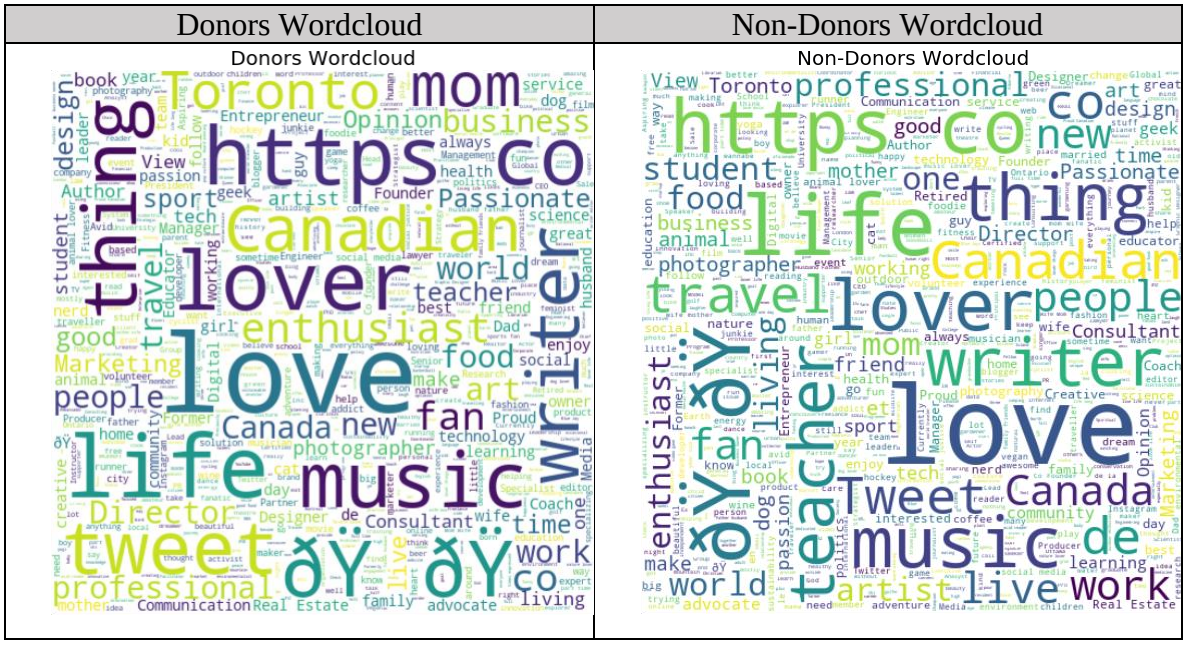
Experiment 2 Results				
Classifier	Accuracy	Vectorizer Technique	Data Type	N-Gram
SVM	66.9%	Count Vectorizer	Profile Description	Unigram
Multinomial NB	70.6%	Count Vectorizer	Profile Description	Unigram
Logistic Regression	68.7%	Count Vectorizer	Profile Description	Unigram
SVM	69.7%	TF-IDF	Profile Description	Unigram
Multinomial NB	71.0%	TF-IDF	Profile Description	Unigram
Logistic Regression	70.1%	TF-IDF	Profile Description	Unigram

5.3 Profile Description Wordclouds

Accuracy metrics are not able to provide a full scope of user behavior, thus wordclouds assisted us in the data exploration. The derived insights show that users utilize their profile descriptions to describe themselves. Some of the commonalities found in all user profile descriptions were the mention of professions. Words such as “designer”, “entrepreneur” and “founder” are examples of users including their profession in their profile. Additionally, they mention their hobbies. The word “love” is frequently used and is combined with the user’s personal interests such as “music” and “food”. Lastly, the presence of hypertext transfer protocol secure extensions is evidence of users including URL links in their profiles, which might be links to their individual websites.

Differences among each set of users can be challenging to discern. A major point to note is that Fundmetric users tend to mention their location more as evidenced by words such as “Canada” and “Toronto”. Additionally, the types of employments also vary by user types.

For instance, entrepreneurial and technology related vocabulary such as “founder”, “co-founder” and “tech” is more present in the random users dataset.



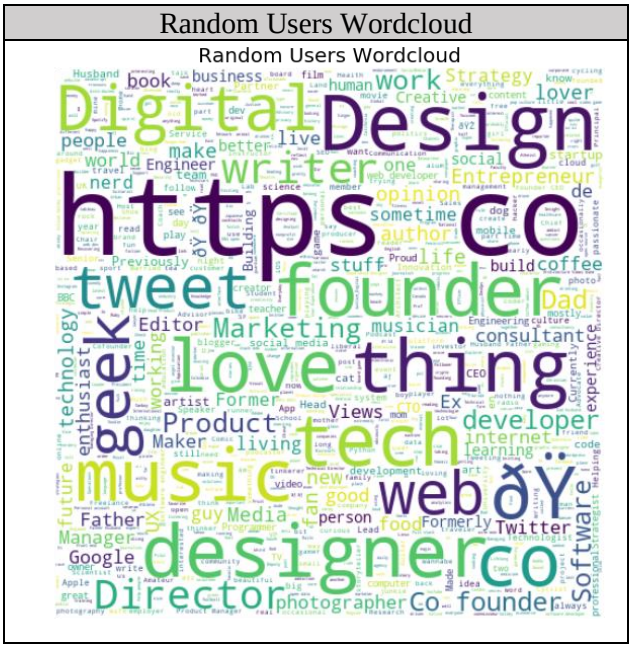


Figure 16: Profile Descriptions Wordclouds

5.4 Applying Word N-grams

The results attained in E2 validated our hypothesis and consequently furthered the need to test out different variations of the model. E3 and E4 were executed to evaluate potential model improvements. By introducing the n-grams as a new technique to vectorize the text data and testing a similar model using tweets, it is hypothesized that accuracy would increase in both cases. However, by comparing the results drawn from both E3 and E4, we are able to conclude that neither technique increased the robustness of the model. Both profile descriptions and tweets experienced a lower accuracy the higher the value of n was in n-grams. Notably, the drop in the accuracy score is more prevalent across profile descriptions, where accuracy decreases on average by approximately 14% from unigrams to trigrams. Unigrams had the highest accuracy for both count vectorizer and TF-IDF techniques. MNB yielded the highest accuracy for both profile descriptions and tweets. Similar to the results seen in E2, MNB using unigrams and TF-IDF on profile descriptions yielded the highest accuracy with an approximate value of 71%.

Word N-gram Results					
Description Accuracy	Tweet Accuracy	Percent Difference	Classifier	N-Gram	Text Representation
66.9%	64.9%	2.1%	SVM	1	Count Vectorizer
70.6%	67.7%	2.9%	MultinomialNB	1	Count Vectorizer
68.7%	66.2%	2.5%	LogisticRegression	1	Count Vectorizer
69.7%	66.8%	2.9%	SVM	1	TF-IDF
71.0%	66.6%	4.4%	MultinomialNB	1	TF-IDF
70.1%	66.0%	4.0%	LogisticRegression	1	TF-IDF
62.0%	60.6%	1.5%	SVM	2	Count Vectorizer
63.4%	60.8%	2.5%	MultinomialNB	2	Count Vectorizer
63.7%	61.4%	2.4%	LogisticRegression	2	Count Vectorizer
63.4%	62.3%	1.2%	SVM	2	TF-IDF
65.0%	62.5%	2.5%	MultinomialNB	2	TF-IDF
62.0%	62.4%	-0.4%	LogisticRegression	2	TF-IDF
51.0%	61.0%	-10.1%	SVM	3	Count Vectorizer
49.4%	60.7%	-11.4%	MultinomialNB	3	Count Vectorizer
58.4%	61.5%	-3.1%	LogisticRegression	3	Count Vectorizer
57.0%	61.1%	-4.1%	SVM	3	TF-IDF
58.2%	61.3%	-3.1%	MultinomialNB	3	TF-IDF
55.2%	61.4%	-6.2%	LogisticRegression	3	TF-IDF

Table 7: Word N-gram Results

Word N-Gram Average Accuracy				
Description Accuracy	Tweet Accuracy	Percent Difference	N-Gram	Text Representation
68.7%	66.2%	2.5%	1	Count Vectorizer
70.2%	66.4%	3.8%	1	TF-IDF
63.0%	60.9%	2.1%	2	Count Vectorizer
63.5%	62.4%	1.1%	2	TF-IDF
52.9%	61.1%	-8.2%	3	Count Vectorizer
56.8%	61.3%	-4.5%	3	TF-IDF

Table 8: Word N-gram Average Accuracy Results

5.5 Applying Character N-grams

The use of character n-grams as opposed to word n-grams yielded lower results across the board. Even though character unigrams show high accuracy in several cases, we cannot use these results to conclude the robustness of the model, given that classification cannot be done based on one single character. Average accuracy scores behave inversely depending on the data type used. Accuracy decreases when the n value decreases on TF-IDF; however, the opposite takes place when count vectorizer is used. The magnitude of these changes is more evidently seen in profile descriptions.

Character N-gram Results					
Description Accuracy	Tweet Accuracy	Percent Difference	Classifier	N-Gram	Text Representation
64.6%	59.9%	4.8%	SVM	1	Count Vectorizer
62.6%	61.8%	0.9%	MultinomialNB	1	Count Vectorizer
64.3%	66.4%	-2.1%	LogisticRegression	1	Count Vectorizer
69.7%	67.1%	2.6%	SVM	1	TF-IDF
71.0%	66.1%	4.9%	MultinomialNB	1	TF-IDF
70.1%	62.8%	7.3%	LogisticRegression	1	TF-IDF
63.4%	61.4%	2.1%	SVM	2	Count Vectorizer
65.9%	63.1%	2.8%	MultinomialNB	2	Count Vectorizer
64.9%	63.1%	1.8%	LogisticRegression	2	Count Vectorizer
63.4%	63.7%	-0.3%	SVM	2	TF-IDF
65.0%	63.0%	2.0%	MultinomialNB	2	TF-IDF
62.0%	63.4%	-1.4%	LogisticRegression	2	TF-IDF
64.4%	66.3%	-1.9%	SVM	3	Count Vectorizer
68.4%	64.8%	3.6%	MultinomialNB	3	Count Vectorizer
66.2%	62.0%	4.2%	LogisticRegression	3	Count Vectorizer
57.0%	62.3%	-5.3%	SVM	3	TF-IDF
58.2%	62.0%	-3.8%	MultinomialNB	3	TF-IDF
55.2%	63.7%	-8.5%	LogisticRegression	3	TF-IDF

Table 9: Character N-gram Results

Character N-Gram Average Accuracy				
Description Accuracy	Tweet Accuracy	Percent Difference	N-Gram	Text Representation
63.9%	62.7%	1.2%	1	Count Vectorizer
70.2%	65.3%	4.9%	1	TF-IDF
64.8%	62.5%	2.2%	2	Count Vectorizer
63.5%	63.4%	0.1%	2	TF-IDF
66.3%	64.3%	2.0%	3	Count Vectorizer
56.8%	62.7%	-5.9%	3	TF-IDF

Table 10: Character N-gram Average Accuracy Results

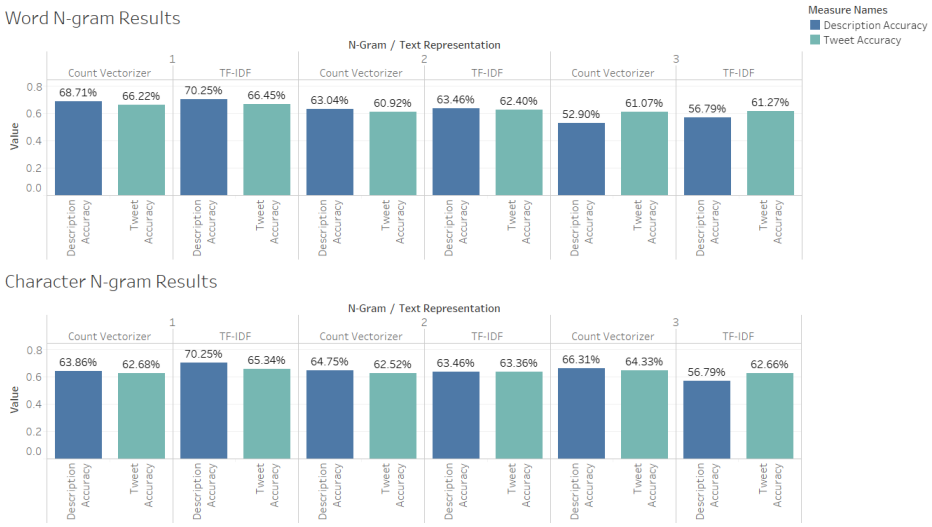


Figure 17: N-Gram Accuracy Results Summary

5.6 Tweet Wordclouds

Wordclouds were applied to tweets in order to visually understand the type of words used in these text documents. Similarities can be found in both groups. First, the character “RT”, which is used when users “retweet” a message from another user, can be seen among the most common words. Second, the presence of hypertext transfer protocol secure extensions is evidence of users sharing content via external URL links to other websites. Identifying differences among both groups again proves to be challenging and one can only speculate that the content on their tweets varies in light of the difference in accuracy results.

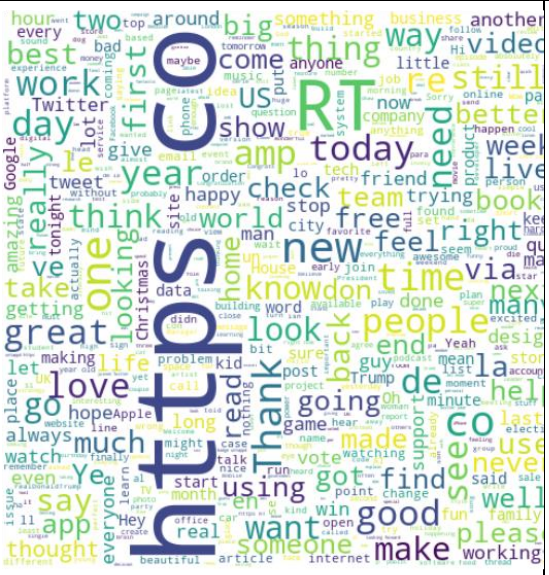
Random Tweets Wordcloud	Fundmetric Tweets Wordcloud
	

Figure 18: Tweets Wordcloud

5.7 Preliminary Model

After following several experiments with different combinations of datasets, algorithms and preprocessing techniques only one algorithm can be implemented as the engine behind the website prototype. Given the various amount of test trials run, it would be hard to agree as to what constitutes the "best" model given that in many cases accuracy may be relatively high, but not on other evaluation metrics, including f1-score, precision, and recall. One may argue that accuracy is not the only means to determine robustness in a machine learning model. The previous experiments show that the use of profile descriptions using TF-IDF and word unigrams yields the highest average accuracy scores. Consequently, f1-scores will be calculated to reassess the robustness of these models. Even though logistic regression has a higher f1-score, the accuracy score still is higher for Multinomial Naïve Bayes. As a result, Multinomial Naïve Bayes is the algorithm implemented as the engine for the website prototype.

Profile Description Word Unigrams using TF-IDF				
Algorithm	Precision	Recall	F1-Score	Accuracy
MNB	0.71	0.7	0.69	71.0%
SVM	0.69	0.69	0.69	69.7%
Logistic Regression	0.7	0.7	0.7	70.1%

Table 11: Final Model Results

5.8 Applications

The findings of this research can assist Fundmetric Inc. in creating more user-centric experiences. It can apply this classification model to identify users who behave similarly to charity donors and target them to expand its customer base. This strategy would lead to a higher client conversion rate and reduce costs in marketing to users that might not even be interested in charitable donations. In addition to expanding its client base, Fundmetric Inc. can use this proof of concept to further segment their clients. For instance, customers could be segmented according to the donation amount. Depending on the amount and frequency of each customer segment, Fundmetric Inc. could evaluate potential time intervals and methods to reach out to clients to ensure they are engaged according to their unique profile. Personalized customer experiences can aid companies in optimizing resources.

Chapter 6

Conclusion and Future Work

This project began by exploring Twitter profile descriptions from Fundmetric users, who have demonstrated previous behavior of donating to charities. Machine learning techniques did not find substantial differences among the users and thus classifying donors from non-donors was not possible, until the initial dataset was expanded by adding random users. After tuning the model, we can conclude that charity donors exhibit distinctive behavior on social media channels. Charities hoping to optimize resources can apply similar content-based approaches to better understand user behavior, ensure the right users are being followed up on and identify prospective donors who have not donated to charities in the past.

Even though this proof of concept has successfully demonstrated the ability of machine learning techniques to be extrapolated to the domain of charitable donations, further improvement can be done to optimize the work done in this project. In the following sections, I will explore potential optimizations that could be applied.

6.1 Data Extraction Automation

All datasets were retrieved by running scraping scripts several times on Twitter API until a sufficient volume of data was retrieved. Specific machines may be assigned and dedicated to the automated extraction of data. This would produce a more robust model because the machine learning model would have more data to analyze and learn from. Naturally, content shared among social media channels might differ across time and be subject to different trends. Thus, the existing datasets used to train the model might become obsolete in the future.

6.2 Application of Deep Learning Models

With the expansion of training datasets, model performance might deteriorate, and more computational power might be required to ensure optimality. To match this increase of data, deep learning models might prove to be an adequate alternative. Marcus explains that deep learning technologies are “data-hungry”, thus matching the expansion of datasets [44]. As mentioned in research by Heidarysafa et al., deep learning may in many cases surpass that of standard machine learning techniques [45]. The application of this subbranch of machine learning might improve robustness in the model.

6.3 Content-based Approach using Followers Information

Statistical-based approaches using data on followers and numbers of users being followed have been explored in previous research. However, it is to the best of my knowledge that no domain has explored the use of follower’s activity to identify users. Subsequent steps would include scraping Twitter API for several followers data on Fundmetric and random users and testing whether this approach yields improved results to the model.

Bibliography

[1] M. Uddin, M. Imran and H. Sajjad, “Understanding Types of Users on Twitter”, Jun. 2014. [Online] Available: https://arxiv.org/abs/1406.1335

[2] Twitter.com, 2020. [Online]. Available: https://twitter.com/ [Accessed: 07-Feb-2020].

[3] Dreamgrow.com, “Top 10 Social Networking Sites by Market Share Statistics [2020]”, 2020. Available: https://www.dreamgrow.com/top-10-social-networking-sites-market-share-of-visits/. [Accessed: 07-Feb-2020]

[4] Entrevestor, “Fundmetric Signs Major Customers”, 2018. Available: http://entrevestor.com/ac/blog/fundmetric-signs-major-customers [Accessed: 07-Feb-2020]

[5] Y. Liang, “A Machine Learning Approach to Fundraising Success in Higher Education”, Master’s thesis, University of Victoria, 2017. [Online]. Available: https://dspace.library.uvic.ca//handle/1828/8028

[6] L. Farrokhvar, A. Ansari and B. Kamali. “Predictive models for charitable giving using machine learning techniques”, 2018. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6169901/

[7] J. Han and M. Kamber, “Data Mining Concepts and Techniques”, pp. 279 – 303, 2001.

[8] S. Xu, “Bayesian Naïve Bayes classifiers to text classification”, in *Journal of Information Science*, vol. 44, iss. 1, 2018. [Online] Available: https://journals.sagepub.com/doi/full/10.1177/0165551516677946

[9] Frank E., Bouckaert R.R. “Naive Bayes for Text Classification with Unbalanced Classes”, in *Knowledge Discovery in Databases: PKDD 2006,* pp. 503-510, 2006. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6169901/

[10] Sigmoid Activation Function from Towardsdatascience.com. 2018. [Online]. Available: https://towardsdatascience.com/logistic-regression-detailed-overview-46c4da4303bc. [Accessed: 07-Feb-2020]

[11] Logistic Regression Formula from Towardsdatascience.com, 2019. Available: https://towardsdatascience.com/understanding-logistic-regression-step-by-step-704a78be7e0a

[12] D. Ramayanti and U. Salamah. “Complaint Classification Using Support Vector Machine for Indonesian Text Dataset”, in *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 3, iss. 7, 2018. [Online]. Available: http://ijsrcseit.com/CSEIT183723

[13] J. R. Brzezinski and G. J. Knafl, "Logistic regression modeling for context-based classification", in *Proceedings. Tenth International Workshop on Database and Expert Systems Applications*. DEXA 99, 1999, pp. 755-759.

[14] Antje Kirchner and Curtis S. Signorino. “Using Support Vector Machines for Survey Research”, in *Survey Practice*, vol. 11, iss. 1, 2018. [Online]. Available: https://doi.org/10.29115/SP-2018-0001

[15] Support Vector Machine Example from Medium.com, 2017. [Online]. Available: https://medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72

[16] D. Sarkar., “Text Analytics with Python: A Practical Real-World Approach to Gaining Actionable Insights from your Data”, 1st ed, pp. 167-217, Apress. 2016

[17] C. Conrad and V. Keselj, "Predicting Political Donations Using Twitter Hashtags and Character N-Grams," in *2016 IEEE 18th Conference on Business Informatics (CBI)*, 2016, pp. 1-7.

[18] A. Tripathy, A. Agrawal and S. KumarRath. “Classification of sentiment reviews using n-gram machine learning approach” in *Expert Systems with Applications*, vol 57, pp. 117-126, 2017

[19] J. Wieting, M. Bansal, K. Gimpel and K. Livescu. “Charagram: Embedding Words and Sentences via Character n-grams”, Jul. 2016. [Online]. Available: https://arxiv.org/abs/1607.02789

[20] TF-IDF Formula from Towardsdatascience.com., 2019. Available: https://towardsdatascience.com/natural-language-processing-feature-engineering-using-tf-idf-e8b9d00e7e76. [Accessed: 07-Feb-2020]

[21] B. Das and S. Chakraborty, “An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation”, 2018. [Online]. Available: https://arxiv.org/abs/1806.06407

[22] K. Daouadi, R. Rebaï and I. Amous, “Organization vs. Individual: Twitter User Classification”, in *Proceedings of the 2nd Language Processing and Knowledge Management international conference*, 2019. [Online]. Available: https://arxiv.org/abs/1806.06407

[23] D. Rao, D. Yarowsky, A. Shreevats, and M. Gupta. “Classifying latent user attributes in Twitter”, in *SMUC '10: Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pp. 37–44, 2010. [Online]. Available: https://doi.org/10.1145/1871985.1871993

[24] M. Pennacchiotti and A. Popescu, “A Machine Learning Approach to Twitter User Classification”, in *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[25] A. Kim, T. Miano, R. Chew, M. Egger and J. Nonnemaker. “Classification of Twitter Users Who Tweet About E-Cigarettes”, in *JMIR Public Health Surveill*, vol. 3, Jul. 2017.

[26] A. Zubiaga, A. Voss, R. Procter, M. Liakata, B. Wang and A. Tsakalidis. “Towards Real-Time, Country-Level Location Classification of Worldwide Tweets”, in *Transactions on Knowledge and Data Engineering*, vol. 29, no. 9, pp. 2053-2066, Sept. 2017.

[27] D. Preotiuc-Pietro, Y. Liu, D. J. Hopkins, L. Ungar. “Beyond Binary Labels: Political Ideology Prediction of Twitter Users”, in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 729–740, 2017. [Online]. Available: https://www.aclweb.org/anthology/P17-1068

[28] K. Stowe, J. Anderson, M. Palmer, L. Palen and K. Anderson. “Improving Classification of Twitter Behavior During Hurricane Events”, in *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pp. 67-75, Jul. 2018. [Online]. Available: https://www.aclweb.org/anthology/W18-3512/

[29] Paypal.com, 2020 [Online]. Available: https://www.paypal.com

[30] UnderArmour.com, 2020. [Online]. Available: https://www.underarmour.com

[31] Mazda, 2020. [Online]. Available: https://www.mazda.com

[32] Matthew B. Hoy, “HTML5: A new standard for the web”, 2011

[33] H. Wium Lie, “Cascading Style Sheets”, Doctor of philosophy thesis, University of Oslo, Norway, 2005.

[34] M. Otto and J. Thornton, “Bootstrap”. [Online]. Available: https://getbootstrap.com/

[35] G. van Rossum. “Python”. [Online]. Available: https://www.python.org/

[36] M. Grinberg. “Flask Web Development: Developing Web Applications with Python”, 2nd ed., O’Reilly Media, Inc., 2014.

[37] S. van der Walt, S. Colbert and G. Varoquaux, “The NumPy array: a structure for efficient numerical computation”, in *Computing in Science & Engineering*, vol. 13, iss. 2, 2011.

[38] W. McKinney, Pandas: a foundational Python library for data analysis and statistics. In: Python for High Performance and Scientific Computing, vol. 14 (2011)

[39] F. Pedregosa. G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M.Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, “Scikit-learn: Machine Learning in Python” , in *Journal of Machine Learning Research*, vol. 12 pp. 2825-2830 (2011)

[40] E. Loper and S. Bird. “NLTK: The Natural Language Toolkit”, Jul. 2002

[41] “Pickle Library”. [Online]. Available: https://docs.python.org/3/library/pickle.html

[42] J. Roesslein and A. Hill, “Tweepy”. [Online]. Available: http://docs.tweepy.org/en/v3.5.0/

[43] Stemming Example. [Online] Available: https://www.thinkinfi.com/2018/09/difference-between-stemming-and.html. [Accessed: 07-Feb-2020]

[44] Marcus, G. “Deep Learning: A Critical Appraisal”, Jan. 2018. [Online]. Available: https://arxiv.org/abs/1801.00631

[45] M. Heidarysafa, K. Kowsari, D. Brown, K. Meimandi and L. Barnes. “An Improvement of Data Classification Using Random Multimodel Deep Learning (RMDL)”, Aug. 2018. [Online]. Available: https://arxiv.org/abs/1808.08121