Prediction of Buzz in Social-Media Using Random Forest Algorithm

Mohammad Ali Haji Hasan Khonsari

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Eastern Mediterranean University
September 2018
Gazimağusa, North Cyprus

Approval of the Institute of Graduate Studies and Research

Assoc. Prof. Dr. Ali Hakan Ulusoy
Acting Director

I certify that this thesis satisfies the requirements as a thesis for the degree of Master of Science in Computer Engineering

Prof. Dr. H. Işık Aybay
Chair, Department of Computer Engineering

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Computer Engineering.

Assoc. Prof. Dr. Duygu Çelik Ertuğrul
Supervisor

Examining Committee

1. Assoc. Prof. Dr. Duygu Çelik Ertuğrul
2. Assoc. Prof. Dr. Önsen Toygar
3. Asst. Prof. Dr. Mehtap Köse Ulukök

ABSTRACT

Good management of the social media monitoring process contributes to effective planning in social networks. Knowing what potential customers are talking about a product brand, about sharing trends, and communicating with them is crucial in terms of marketing strategies. Buzz is actually about how a product brand is positioned in the eyes of its users and customers. Beside this, Buzz prediction on social media channels such as Twitter is a challenging task that has been generated from real data by defining different features to represent the Buzz case. These predictions are helpful in analyzing important brands' Buzz posts of their potential customers' considerations in social networks. In the majority of our related researches, Support Vector Machine (SVM) combined with Radial Basis Function (RBF) approach was observed and investigated. In addition to executing the prediction in the research studies, the data set used is classified. In this study, we used another method in order to cope with these predictions, named Random Forest (RF). This method has one more advantage than the mentioned ones which is rank ordering of the related data set. The findings on the same data set and the comparison between the mentioned three methods showed that the RF gives the overall better accuracy result with the value of 99% and fastest training time. It is also inferred that the Buzz is a dynamic event in which the basis of prediction could be modelled on the content as well as the forest. It can detect the most significant attributes in order to identify the created topic is either Buzz or not. Finally, the use of much faster and more reliable algorithms for Buzz prediction from products and brands comments in social media is crucial.

Keywords: Buzz prediction, Random Forest, Support Vector Machine, Twitter.

ÖZ

Sosyal medya takip sürecinin iyi yönetilmesi, sosyal ağlarda etkili planlar yapılmasına katkıda bulunur. Potansiyel müşterilerin, bir ürün markası hakkında neler konuştuğu, ilgili paylaşım eğilimlerini bilmek ve onlarla iletişime geçmek pazarlama stratejileri açısından son derece önemlidir. Buzz aslında, bir ürün markasının, kullanıcılarının ve müşterilerinin gözünde nasıl konumlandığı ile ilgilidir. İlaveten, Twitter gibi sosyal medya kanallarındaki, Buzz tahmini, müşteri yorumlarını analiz etmek için farklı özellikleri tanımlayarak, gerçek verilerden oluşturulan zorlu bir görevdir. Bu tahminler, önemli markaların potansiyel müşterilerinin sosyal ağlardaki düşüncelerini Buzz yayınlarını analiz etmede yardımcı oluyor. İlgili araştırmalarımızın çoğunda, Radyal Temel Fonksiyonu (RBF) yaklaşımı ile Destek Vektör Makinesi (SVM) gözlenmiş ve araştırılmıştır. Araştırma çalışmalarında tahmin etmenin yanı sıra, kullanılan veri kümesi sınıflandırılmıştır. Bu çalışmada araştırmacılar bu tahminlerle baş edebilmek için Rastgele Orman (RF) adlı başka bir yöntem kullanmışlardır. Bu yöntemin, diğerlerine göre avantajı ilgili veri kümesini sıralamasıdır. Aynı veri setindeki bulgular ve bahsi geçen üç yöntem arasındaki karşılaştırmalar sonucu %99 başarı değeri ve, en hızlı eğitim süresi ile, genel olarak daha iyi bir doğruluk sağladığı gözlemlenmiştir. Ayrıca Buzz'ın, öngörünün sadece içeriği değil, aynı zamanda ormanı da içeren modellere dayandığı dinamik bir fenomen olduğu sonucuna varılmıştır. Son olarak, sosyal medyada ürün ve marka yorumlarında Buzz tahmini için, çok daha hızlı ve güvenilir algoritmalara ihtiyaç vardır.

Anahtar Kelimeler: Buzz tahmini, Rastgele Orman, Destek Vektör Makinesi, Twitter.

In memory of my grand parents.

To my mother Shahla and father Valiollah.

To my sisters Shadi and Lamya.

Last but not least to all my new great friends that I met here in EMU, especially Dr.Pejman Bahramianfar who is the best for me.

ACKNOWLEDGEMENT

I would like to thank God for everything he offered to me. Special thanks to my supervisor Assoc. Prof. Dr. Duygu Çelik Ertuğrul for her amazing support, notes and advices that leaded me in writing my thesis.

Thanks to my mom's prayers and to my family support. Thanks to my dear father who supported me for my education. Thanks to my sister Lamya who supported me morally to reach this point. Thanks to my close friends Pejman, Sasan, Haman, Aryan and my Boss Mr.Efe Sidal for supporting me always in this step of my life.

I love you all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

All users of applications of social media networks can typically access to various kinds of social media channels through web-based technologies. As users are engaged with such services, a possibility is given in order to create some interactive platforms which are able to be shared by individuals, communities and organizations. This process can be followed by some discussions, co-creations or even modifications of user-generated and pre-made contents. These contents and discussions can be posted online as well. The introduction of some new topics of discussion is also possible and there might be some pervasive and expected changes in communication between individuals, organizations and communities. To sum up, the social media can make some changes in the communication system of individuals and in the large organizations. These changes tend to have more focus on the emerging fields of techno self-studies.

Twitter network was first established by J. Dorsey, N. Glass, B. Stone, and E. Williams that was fully improved in July 2006. This service gained the popularity of worldwide 6 years later, in 2012. Users of about 100 million sent tweets of about 340 million each day and 1.6 billion search queries about an average is done per day were handled by the service [1]. One of the online news and social networking services is Twitter. Through this service, users can send posts and have interactions by messages that are called "tweets”. Tweets have restrictions of approximately 140 characters. They were regarded as one of the largest known sources of news that are breaking by over 40 million election-related sent posts. The majority of events and topics have been discussed on Twitter that are about different fields such as current trends, Marketing strategies, Personal tweets, etc. The tweet can be either a Buzz or a valid information i.e. not a Buzz [2].

Trending topics are consisting words and phrases. Some of these topics are regarded as greater rates than other similar samples. They can be considered as popular by concerted effort of users through talking about some specific and helpful topics for Twitter. All users are involved in the understanding of what is going on in the world and what people think about. Trending topics are also remembered as the symbol of the result for some concerted efforts, the manipulations of instant teenager fans of celebrities or musicians like Lady Gaga, Justin Bieber, Rihanna, and the novel series Twilight and Harry Potter. It is a fact that the Twitter has already altered the trend of algorithm in the past in order to prevent some similar manipulations of this kind through the related limited success. It is also regarded as a kind of quite real time in nature. It is also a very robust source for getting the real time trends in the well as the news coverage and the greatly increased accessibility. By following some sources such as sites, housing printed publications, pure players, news agencies, blogs, articles, and the growth, it can be continued in order to feed such popular Internet facts. This is regarded as publications that originates from blogs. Synthesis categorized with Bouygues E-lab in order to deal with and plan the events of some levels of Buzz available on the websites in the way that it is being spread during some sites for media, video-sharing platforms and blogs. This study attempts to have a great look at the way through which the Buzz has been breaking down among the different sites as well as at other cycles for publishing that are the same as some certain events.

Predicting the behavior of users in social networks is an extremely challenging work. First, one of the most of existing approaches discusses primarily a global behavior that is predicting model with a goal of finding of a uniform model fit all users. It also ignores individuals' behaviors. In addition, although social impacts play important role in information diffusion, it has been largely ignored in conventional research. Hence, a system is needed to predict whether a discussion is a Buzz or not in the initial stage. This system should have highest accuracy too. There are some well-known methods used in this regard such as Support Vector Machines (SVM) and Radial Basis Function (RBF). One of the ways to interpret RBF is a simple way that is known as single-layer type of Artificial Neural Network (ANN). ANN is known as a radial basis function network that has the radial basis functions which play the activation role of the network SVM. In the process of machine learning, SVM is regarded as model which is a supervised one and it is associated with learning algorithms that have the role of analyzing the used data for classification and regression [3].

This study concentrates on a 'classification and prediction of the Buzz" a data set of the 'Twitter' and then, it analyzes possible relationships among users through Random Forest (RF) approach. [4][5] RF or Random Decision Forests (RDFs) are known as methods for ensemble learning used for some tasks such as classification or regression. These methods construct a multitude of decision trees during the training time. They also output the class that are regarded as mode of the classification or mean prediction (regression) of any single tree.

Different features represent the Buzz case in the data set and that leads to have a serious imbalance in the dataset. Hence; we applied RF methodology in order to define and to capture all inherent phenomena in the dataset. Regarding that the data set uses different features to define the Buzz case, we applied a popular form of feature ranking analysis to identify the most significant factor affecting the Buzz case. As far as we inferred, this study seems to be the initial interpretation that uses a feature ranking methodology to rank the factors affecting Buzz in the social media dataset.

This thesis is organized as follows: Chapter 2 presents a brief review of the literatures; Chapter 3 presents the used data, related descriptions and the used methodology; Chapter 4 presents the empirical findings. Finally, Chapter 5 states the inferred conclusion.

Chapter 2

LITERATURE REVIEWS

This Chapter tries to deal with the major existing contributions that are highly related to the subject of this study. Mayuri et al. used Radial Basis Function Network (RBFN) in the prediction of the Buzz in Twitter through using of attributes of a discussion in Twitter in 2017. RBFN are regarded as classifications and some functional approximations of neural network algorithm that have been working with non- linear values which enables complex data to be manipulated. Twitter has large amount of non-linear data; therefore, RBFN is known as a suitable function for the related analysis and also it is regarded as a feed forward network trained by supervised training algorithm and was established to do faster than back propagation networks [2].

Mayuri et al tried to use the Radial Basis Function Network for predicting the Buzz in Twitter through using attributes of a discussion in Twitter [2]. The radial basis function is a classification and functional approximation neural network that uses most common non- linear values. Using RBFN that uses non- linear set of data enabled the researcher to deal with complex data. Since Twitter has a large amount of non-linear RBFN, it perfectly suits for it and it feed-forwards the network. It was also trained through using supervised training algorithm that performs very faster than back propagation networks [6]. Accurate results were obtained by using this Radial Basis Function Network and it was shown that the RBFN can work with small sample sizes very efficiently.

2.1 Buzz Prediction System Architecture

Buzz prediction system has main components that are as follows: Random sampling, RF Training, RBF testing and Buzz Prediction.

In order to achieve unbiased results in a study one of the best ways is known as random sampling. It is a quite quick and easy way for obtaining unbiased results in a selected population that is going to be surveyed and also it is regarded as one of the ways for getting the most possible accurate information. In this sampling way there are three common methods.

Random number tables that have recently been regarded as random numbers generators, has been used as guide by researchers for the selection of subjects at intervals which are generated randomly. In this way some specific mathematical algorithm for pseudo-random number generators are also useful and may function effectively. There are some Physical randomization devices that may be simple like an electronic device that is called ERINE.

There are a lot of advantages of using random sampling in a survey, the biggest of which is the fact that subjects are clearly randomized, therefore; it is regarded as the best way to be ensured that the obtained results are unbiased, as another benefit being much faster and less expensive can be counted. Being able to provide valid results as well as enabling researchers to draw conclusions about large populations easily are also worth to be mentioned. The training process for this way is done by using the training data that are obtained by random sampling and it has three different phases.

The first phase determines the centroids that are regarded as representative x-values selected from the training data. An RBF network needs to have one centroid for every hidden node. The second phase of training determines widths that are regarded as values for describing the distance between the centroids. An RBF network also needs one width for every node. The third phase of training determines the RBF weights and bias values that are regarded as numeric constants. In case of having NI number of input nodes for an RBF network, NH number of hidden nodes will be. Testing of RBFN is done by using a new set of data that are called testing data. This dataset is used for the prediction of mean number of discussions that are active at a particular time.

2.2 Buzz Prediction

It can be understood from the output that has been predicted from the RBFN that whether it is Buzz or not. This is possible through analyzing the output that has been obtained from RBFN which is referred to as the mean number of the active negotiations/conversations. More mean number of the active conversations, the more valid discussion will be otherwise it is a Buzz.

Artificial Neural Network is another model of information processing which is modelled after biological nervous system, like the brain and its information processing phenomenon. One of the key elements of this model is the characteristic structure of the system where a large number of the processing elements, called neurons are highly interconnected the aim of solving problems introduced to the system. Learning process in biological systems involves the adjustment of some synaptic connections located between the neurons. Neural networks are highly applicable to real life problems and are used in many industries. Their configuration are highly dependent on the problem to be solved, hence, the designer needs to choose suitable input nodes, output nodes and hidden layer nodes using previously gained experience. The parameter for learning rate and momentum term were adjusted periodically to increase the rate of convergence, since the suitable architecture for each application is determined through trial and error method.

2.3 Radial Basis Function Neural Network

Radial Basis Function (RBF) is another unique type of a neural network that employs the radial basis function as its activation function. These networks are commonly used recently because of their function approximation, curve fitting, and prediction of time series [7]. One of the important factors in these networks is the choice of the amount of neurons in the hidden layer, where every neuron possess a specific activation function, because it has effects on the complexity of the network as well as the general capability. The most preferred function for activation is the Gaussian function that possess spread parameter for controlling the function's characteristics and operations.

Rastogi & Bist elaborated on the way through which different Machine-Learning techniques can classify features of time-windows of Twitter. Moreover, the researcher dealt with whether or not these times-windows are followed by Buzz events. Different machine learning techniques like Naïve Bayes and SVM were compared in order to find the accuracy of classification by regarding with or without applying dimensional reduction in the number of attributes with the help of Principal Component Analysis PCA algorithms. In 2014, a system was proposed by the study [4] that predicted Buzz events by a neural network algorithm which combines three features for predicting the number of retweets associated to a particular tweet on the Twitter. These features are based on expressivity, popularity, and singularity that are determined by extracting keywords that have been associated to a tweet that fits into a model which are estimated in thematic form from the Latent Dirichlet Allocation (LDA). The popularity of tweet is determined by analyzing RSS feeds statistically, probability of associating dominant themes is a saliency measure and uses unlikely associations of theme as a factor favored by the audience. Another indicator analyzes the similarity and associativity score of the tweet texts based on a sensitivity lexicon initially annotated. Aswani used a hybrid computing system inspired by biology in order to determine Buzz in Twitter [3]. ‘‘Buzz’’ is a potential outlier throughout the analysis and using Artificial Bee Colony (ABC) optimization gives a search algorithm hinged on a population where artificial bees search for sources of food. This function is based on how bees intelligently communicate with each other in a colony in order to detect and get to food sources. Bee colonies usually have 3 types of bees namely the onlookers, the scouts and the employers. These names are based on the way they search for food sources, pass on information about potential food sources and make the choice between alternative food sources. This way is regarded as a simple optimization method that employs parameters like size of colony and it segregates ‘‘Buzz’’ Twitter discussions successfully while avoiding getting stuck in local optimum solutions [3].

The idea of Buzz has become popular on social media and has led to innovative ventures in different sectors such as digital marketing and information management. There are many research that tried to investigate the drivers of discussions that become viral, that make Buzz on social media providers like twitter and has gained the attention of either individuals or organizations [8]. This study tries to mine, extract, differentiate and group outliers on social media texts by regarding ‘‘Buzz’’ as outliers. However, this study’s contributions are in two different aspects. The first aspect is the domain which uses a specific 11 attributes to detect and group Buzz. This is advantageous in many domains such as social media based marketing and social media information management where Buzz text are used to understanding user/community behavior as well as analysis of the resulting impact of such discussions on a population.

Karaboga & Basturk proposed hybrid method using k-nearest neighbor used alongside artificial bee colony optimization for identification of outliers in the dataset. The k-nearest neighbor method is one of the favored method by researchers because of its efficiency in detecting outliers [9]. Taking a look to the literature, it can be inferred that artificial bee colony optimization is also known for obtaining fairly accurate, with guarantee of reaching global optimum due to the criteria used in selection and neighborhood identification methods employed to converge to the solution [10][11][12]. By considering objective of obtaining a global optimum solution, the proposed method was found as useful for detecting outliers. Not any similar approach has been explored in a literature, while this study can be regarded as further study on the nearest neighbor approach of detecting outlier, carried out with the aim of proposing a composite mixture algorithm using artificial bee colony optimization plus k- nearest neighbors. Exploring such hybrid approaches in domains like Web 2.0 has vast amount of studies that uses classic approaches such as neural networks, particle swarm and genetic algorithms [12].

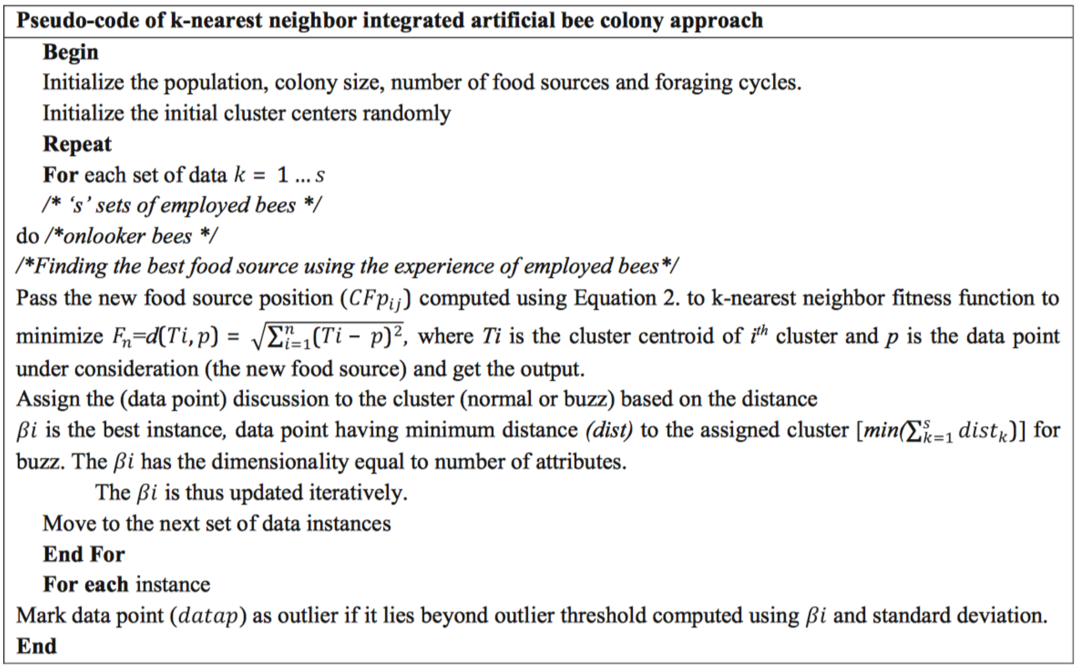
Karaboga & Basturk studied a mixed method of research which deals with bio-inspired computing as well as social media analysis was used because it was difficult to tackle the aim of the research using a sole interdisciplinary method. The main focus was to identify of Buzz on Twitter through the use of a hybrid methodology for the identification of outlier.

2.4 Hybrid Artificial Bee Colony (ABC) Approach

Artificial bee colony optimization can provide a method for population dependent search in which food sources are assessed by artificial bees [13]. This approach was modified and investigated through the years for different application domains [14][15]. It gives guaranteed results in different of domains [10][11]. ABC is based on how bees intelligently communicate with each other in a colony in order to detect and get to food sources. This population is composed of three types; namely the onlookers, the scouts and the employer bees. These names stem from the role of each bees in a colony in finding sources of food and choosing among food sources. It has been known as a simple method of optimization that uses parameters like size of colony, sources of food and exploration area as variables which control the algorithm. Its aim is the ability to randomly move through the forage are, done by Scouts, in search of food sources. Then determining food sources with the highest nectar amount, while updating their positions at all times. Other bees share the information of food sources and their nectar amount in deciding which sauce to go next. Therefore, selection of sources of food is primarily dependent on the category of bee; for employed bees both their own experience and the experience of their mates is used to decide food source. Onlookers have the responsibility of dancing to display the sources of food and their corresponding amount of nectar. All bees store the position of previous food source in the case where a candidate solution possess more nectar than the previous one, in which case the candidate become the current food source. ABC is mostly used in order to optimize the problems and to establish a possible solution by representing each solution as food sources [13][16]. The amount of nectar (NecAmt) on a food source indicates the fitness of the food source, which corresponds to the quality of the solution in this algorithm. For every food source, an employed bee is assigned, and these modifies the position in their memory to fit their present position, source of food. It depends upon the visual information that is locally available and is done by assessing the amount of nectar which corresponds to fitness of particular food source.

This proposed algorithm implements k-nearest neighbor used alongside ABC optimization as shown in Table 1. It is employed in order to explore and extract the outliers using 11 attributes and the related results are confirmed by calculating the mean of active texts that have been assumed using the same attributes. This proposed method can give 98.37 percent accuracy.

Table 1: Pseudo-Code for the Proposed Method [3]



```
Pseudo-code of k-nearest neighbor integrated artificial bee colony approach
  Begin
  Initialize the population, colony size, number of food sources and foraging cycles.
  Initialize the initial cluster centers randomly
  Repeat
  For each set of data $k = 1 \dots s$
  /* 's' sets of employed bees */
do /*onlooker bees */
/*Finding the best food source using the experience of employed bees*/
Pass the new food source position ($CFp_{ij}$) computed using Equation 2. to k-nearest neighbor fitness function to minimize $F_n = d(Ti, p) = \sqrt{\sum_{i=1}^{n}(Ti - p)^2}$, where $Ti$ is the cluster centroid of $i^{th}$ cluster and $p$ is the data point under consideration (the new food source) and get the output.
Assign the (data point) discussion to the cluster (normal or buzz) based on the distance
$\beta i$ is the best instance, data point having minimum distance (dist) to the assigned cluster [$min(\sum_{k=1}^{s} dist_k)$] for buzz. The $\beta i$ has the dimensionality equal to number of attributes.
      The $\beta i$ is thus updated iteratively.
  Move to the next set of data instances
  End For
  For each instance
Mark data point ($datap$) as outlier if it lies beyond outlier threshold computed using $\beta i$ and standard deviation.
End
```

Figure 1 shows the plots of outliers. Where the red patches denote the ‘‘Buzz’’ and normal discussions are noted by blue patches, depicting texts that did not generate enormous attention.

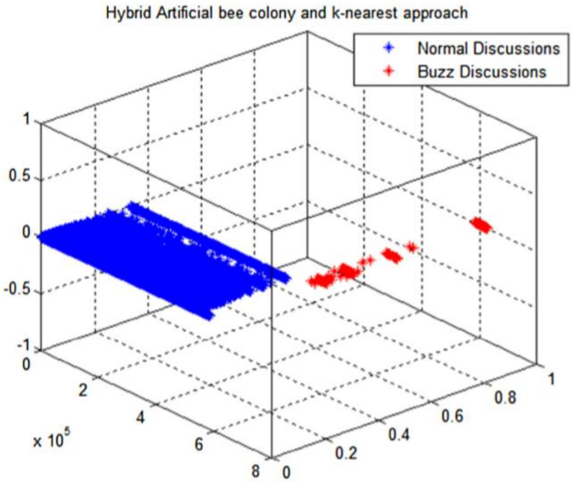


Figure 1: Outlier Plots of the Proposed Approach [13].

Further validation of the results is done with the aid of fivefold cross validation, which result in of 97.87 percent average accuracy score. The dataset is divided into 2 parts, 60 percent of the dataset forms training set that is selected randomly while and the remaining 40 percent is for testing.

Digital age as well as the introduction of Web 2.0 have resulted in a rapid increase in use of social media as a preferred tool for communication around the world including among individuals as well as for customer service and other marketing related usage. The meaning of ‘‘Buzz’’ has been known as trending concept that significantly interest a group of people and this may spread even more. Almost all researches that have been done related to this issue tried to encompass the different factors surrounding the same issue. This study like the related ones is also trying to specifically show Buzz in social media by using a set of attributes discussed earlier. These attributions are comprise of generated text/discussions, increment in number of authors, level of gained attention, burstiness level, sparseness of contributions, interaction between authors, number of authors and calculated average length of discussions. These are employed to differentiate Buzz discussions from other topics on the platform. For the purpose of analysis, this research used 583,249 different topics from Twitter texts in total.

Considering the methodological aspect, this study attempted to propose a hybrid approach in order to detect outliers in the form of Buzz through integration of k-nearest neighbors plus artificial bee colony optimization. This outlined method is able to converge at globally optimum solution thereby avoiding being stuck in a local optima which is common phenomenon in traditional machine learning methods. Moreover, this method is also described as involving a lot of computation when employed for the purpose of dissecting high amount of data. Buzz texts are considered as outliers that deviates from normal interactions and it is able to successfully identify them with an accuracy of 98.37 percent. When compared with similar nearest neighbor dependent gray wolf optimizer for outlier identification, it was found to outperform them not only in accuracy but also the speed of convergence. These results could be of help in e-commerce, marketing and digital that is based on influences by using the approach to identify characteristics that may lead to Buzz and their effects on the consumers. The method will be scaled so that datasets with high volume, veracity and high number of varieties can integrate with any parallel programming framework.

The RBF networks training process have the optimization of spread metrics of every neuron. RBF network belongs to a category of feed forward neural network that comprises 3 layers, known as the input, hidden and output layers. The weights between the hidden layer and the output layer are to be selected to appropriately fit the system, this is done by trial and error. At the end, the bias values which are generated with each output are identified in the RBF network training process. Figure 2 shows a general block diagram of an RBF network.

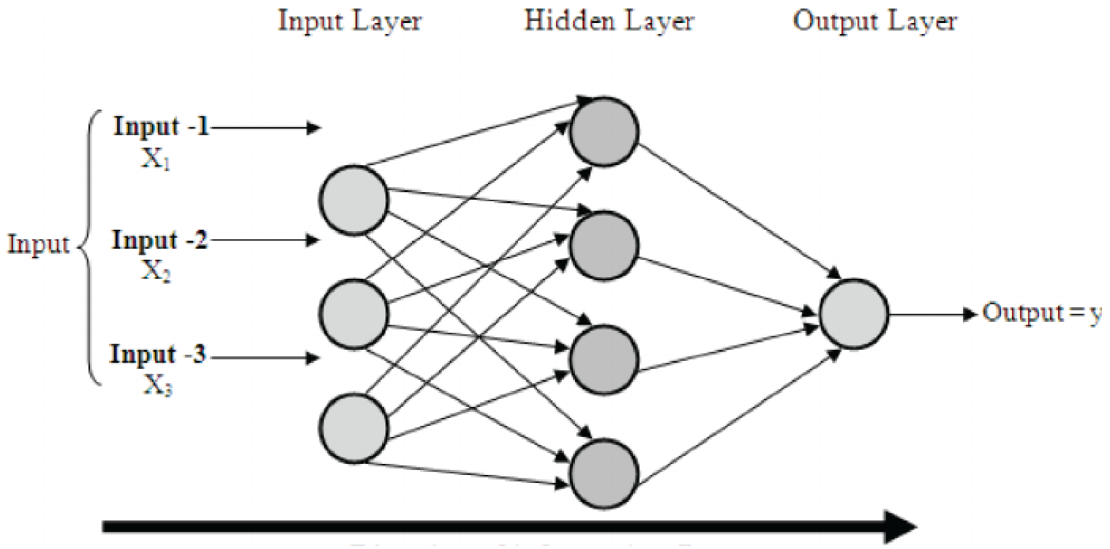


Figure 2: Radial Basis Function Network [2].

2.5 Multiple Regression

Multiple Regression is used to predict the dependent variable when the independent variables are known. The equation of Multiple Regression can be expressed as:

$$T = \alpha + aA + bB + cC + \cdots + zZ \; where \tag{1}$$

$T \; is \; Target \; Variable;$

$\alpha, a, b, c \; are \; Contants;$

$A, B, C \; are \; Independent \; Varibales.$

2.6 Comparison of Results with Multiple Regression

Graph in the Figure 3 shows the accuracy of results obtained by RBFN and Multiple Regression. Error is predicted by using “Mean Squared Error method”. The Comparison of obtained results of RBF and Multiple Regression implies that the results obtained by RBF are more “ACCURATE” than that of multiple regression.

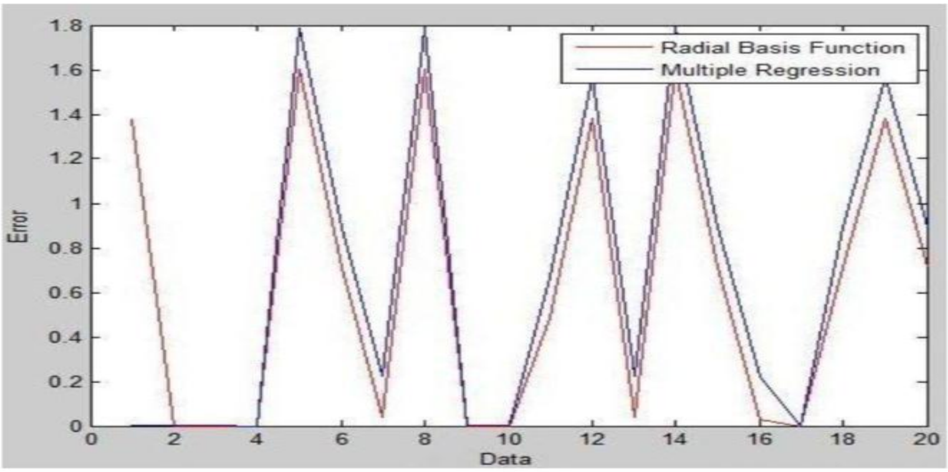


Figure 3: RBFN and Multiple Regression Accuracy Graph [2].

For preparing data in order to train our Naïve Bayes classifier and SVM algorithms, a scientific multidimensional array was generated from the.csv file and used the float data type. The same technique was used for SVM in order to save data in array form. There were 77 attributes that contain real type values entries with no missing values. Two label classes were provided and represented by 0 and 1. In this set the ‘0’ represented Non-Buzzed Event and 1 represented Buzzed Event.

2.7 Studied Algorithm

In order to observe what is achievable, Gaussian Naïve Bayes classification was used as an introductory. More sophisticated and advance techniques like SVM as well as to Principal Component Analysis (PCA) were also used to see if improvements that are possible to be made in the classification test. Each set of methods were experimented through the use of different features as well as different set of training and testing data.

$$P(y|x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n \mid y)}{P(x_1, \dots, x_n)} \tag{2}$$

By the use of the naive independence assumption for all the features that

$$P(x_i|y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i|y) \tag{3}$$

For all i, the relationship is further simplified to:

$$P(y|x_1,\dots,x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|\,y)}{P(x_1,\dots,x_n)} \qquad (4)$$

Since we know that P(x1...x n) is constant given the input, the following rule of classification can be employed:

$$P(y|x_1,\dots,x_n) \propto P(y)\prod_{i=1}^{n} P(x_i|\,y) \qquad (5)$$

$$\downarrow$$

$$\hat{y} = \arg max_y\, P(y)\prod_{i=1}^{n} P(x_i|\,y) \quad ,$$

Gaussian NB used the Gaussian Naive Bayes approach for classification. The likelihood of the features is taken to be Gaussian:

$$P(x_i|\,y) = \frac{1}{\sqrt{2\pi\sigma^2{}_y}} exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \qquad (6)$$

where, the parameters σ y and μy are predicted using maximum likelihood which is used as method of predicting the metrics of a statistical model for the given data.

2.8 SVM

SVM which stands for the Support Vector Machine was initially proposed by Vapnik [16][17]. It is known as one of the important supervised algorithms that has been regarded as the best one in its kind to offer optimal marginal classification. Based on the obtained results from recent studies SVM is highly effective in terms of the accuracy in classification with respect to the other algorithms [17]. It can separate the large chunk of the available data with a gap that can also separate the data points belonging to a different class. These data points that lie on these gaps are the Support Vector Points. They are based on the theory of decision planes that identify decision boundaries and they are able to separate a set of objects that belong to different classes.

Support Vector Algorithms work on various parameters that are effective in the result and the optimal time to achieve it.

Different parameters have been experimented in terms of better accuracy. There are a lot of parameters such as different kernel functions, the standard deviation of the Gaussian kernel and the number of training examples. Mathematical discussion of support vector algorithms is provided taking n features. Let us assume different data points as: $\{(x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots (x_n, y_n)\}$. And there are two classes for y $_n$ = 1 or -1. These data points can be visualized as by segregating hyper plane, that can be mathematically represented as:

$$w.x + b = 0 \tag{7}$$

where, b is scalar (similar to a bias feature in Regression analysis) and w is n-dimensional Vector. Factor b restricts solution by avoiding the hyper plane pass through origin all the time. We are focused to get high margin classification and there exists two classes y $_n$= -1 or 1. So, hyper plane which is parallel for both class share same features and scalar factor b, which are mathematically described as:

$$\begin{gathered} w.x + b = 1 \\ w.x + b = -1 \end{gathered} \tag{8}$$

If the training data can be separated with a single decision surface, hyper planes can be selected so that there are no points between them and thereafter try to maximize their distance. With the aid of geometry, we are able to find the distance between the hyper planes to be 2 /│w│. But we should minimize │w│. And to increase activities around data points, we need to ensure that for all i either w.xi – b≥1 or w.xi –b≤-1.

This is given as

$$y_i(w.x_i- b) \geq 1, 1 \leq i \leq n \quad (9)$$

Data Points that reside along the hyper planes or decision boundary are known as Support Vectors (SVs). A hyper plane that separates using biggest margin represented by $M = 2/|w|$ that is specifies support vectors refers to training data points closest to it.

$$y_j [w^{T.} x_j + b] = 1, i =1 \quad (10)$$

Different kernel (parameter) will be dealt with that have influence testing result and the accuracy. These kernels are: Linear kernel: $K (x_i, x_j) = x_i^{T}\cdot x_j$. Polynomial kernel: $K (x_i, x_j) = (\gamma x_i^T x_j + r)^{d,}$ $\gamma > 0$. Radial Basis Function (RBF) kernel: $K (x_i, x_j) = \exp (-\gamma \|x_i - x_j\|^2)$, $\gamma > 0$. Here, γ, r and d are kernel parameters. In these popular kernel functions.

SVM has been used in the majority of real world problems used in different engineering application like image recognition. The output result of SVM is very responsive to how the cost metrics and kernel metrics are set. Therefore, the user must carry out a rigorous cross validation in order to arrive at the appropriate optimal metric settings for a particular study.

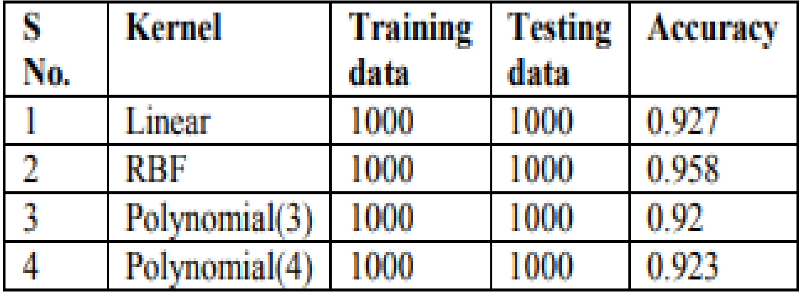
2.9 Principal Component Analysis

Karl Pearson created Principal Component Analysis (PCA) in 1901[3] and it was known as a correlative of the principal axis theorem in mechanics18]. Harold Hoteling later named the algorithm to be PCA [18]. PCA is known as a classical statistical method of turning features of dataset into a new set of features that are not related called Principal Components (PCs). The amount of principal components might be smaller or the same as the amount of original variables. PCA can be used to decrease the dimensionality of a data set, while still keeping high percentage of the variability of the dataset. Data with high dimensions can be a problem for machine learning because predictive models based on such data run the risk of over fitting [18]. These features may decrease the possibility of getting more accurate results from the testing data sets. Moreover, a good number of the features may be repetitions or even regarded as closely related to each other, which may result in a low accuracy. Therefore, for having higher accuracy, it is necessary to consider more important features that have effects just on the region of the classifier for various classes.

2.10 Experiments

The classification experiments were conducted on Buzz in social media. Data Set could be taken from https://archive.ics.uci.edu/ml/datasets/Buzz+in+social+media+. Python language and its tools were used for experimentation. Both methods on a different data set with and without dimensional reduction were employed. Using Naïve Bayes in order to get baseline accuracy was followed by SVM with different kernel linear, polynomial, and RBF. The result of different machine learning techniques has been studied and the result is framed on a table with pictorial representations of Machine Learning Techniques with respect to Accuracy at given algorithms as shown in Table 2 and Figure 4.

Table 2: Comparing Accuracy of Different Kernels [5]



S No.	Kernel	Training data	Testing data	Accuracy
1	Linear	1000	1000	0.927
2	RBF	1000	1000	0.958
3	Polynomial(3)	1000	1000	0.92
4	Polynomial(4)	1000	1000	0.923

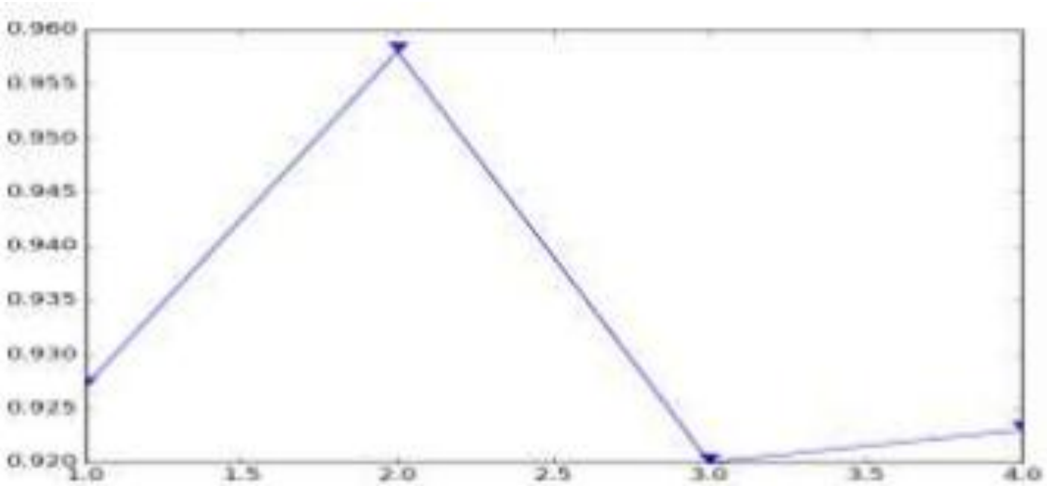
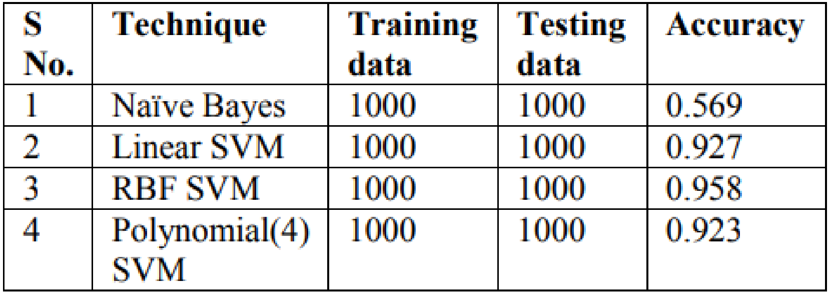


Figure 4: Comparing Accuracy of Different Kernels [5]

Results were obtained after applying the machine learning algorithm to train the classifier through using all the 77 attributes which the 77 dimensions of the multidimensional array of data sets are. Among these three methods, Naïve Bayes performed the worst and SVM performed the best which differs by 38.9% approximately. All the methods have roughly the same performance on our data set, excluding the Naïve Bayes. This is probably because there was one feature that was not strongly associated with buzz event.

It can be assumed from the Naïve Bayes model that almost all features are independent and also features that are independent are not necessarily a bad assumption for our problem. Table 3 illustrates a summary of the achievable accuracy, using Naïve Bayes and SVM. In SVM. RBF kernel (non-linear) outperforms the Linear SVM and gives better result but we cannot question the optimal result on the basis of large dataset.

Table 3: Machine Learning Classifiers without Dimensional Reduction [5]



S No.	Technique	Training data	Testing data	Accuracy
1	Naïve Bayes	1000	1000	0.569
2	Linear SVM	1000	1000	0.927
3	RBF SVM	1000	1000	0.958
4	Polynomial(4) SVM	1000	1000	0.923

It can be inferred therefore that after applying dimensional reduction in datasets, an increase in accuracy of the classifier was seen. It shows that even though many features were given in data, most of the features were found not useful in classification.

The obtained results of the mentioned practical work emphasis that although Naïve Bayes classifier is not able to give higher accuracy, it showed vast improvement after the features are transformed through PCA algorithms. The system proposed is composed of 3 major steps; in the first step, keywords unique with a tweet are extracted. This step is based on a theme-based model predicted from the Latent Dirichlet Allocation (LDA) algorithm. Secondly, the descriptors for popularity, singularity as well as expressivity are extracted from a text and its theme model representation. Lastly, a neuronal network is employed to identify amount of retweets for every tweet with the aid of the initially formed descriptors.

Tweet t is represented with a feature vector W^t. Estimation of a LDA model on a large enormous body of texts D of documents to produce a topic space T_{spc}. Projection of

W Extraction of a subset S^W representing the tweet key into T_{spc} to select a subset of topics words from S^Z regarding W^t $S^{\ z} \subset T_{spc}$ representing the tweet. Removal of an index vector from S^W having coefficients depicting the score of popularity, expressivity and singularity. The steps are further expatiated on below.

2.10.1 Keywords Extraction

Twitter limits the size of each messages to 140 characters until recently where 280 character text has been introduced. Based on this limitation, using a particular vocabulary that is often uncommon, including fabricated words, misspelled and/or even truncated words is obtained [19]. But using the tweet words alone is insufficient [4].

For compensating these particularities, two approaches have been compared in order to raise the first tweet lexicon from an additional body of text documents: a classicistic word representation with the TF-IDF-RP method [20] and a topic space representation with the LDA approach [21].

2.10.2 Keywords Extraction using TF-IDF-RP

D represents a body of n_d documents d and n_W is the vocabulary size. Every tweet t can be inferred as a location of IR^{n_W} by the vector W_i^t of size n_W where the i^{th} feature (i = 1, 2,..., n_W) put together; the Term Frequency (TF), Relative Position (RP) and the Inverse Document Frequency (IDF) [20] of a word w_i of t:

$W_i^t = tf_i.idf_i.rp_i$. This method allows for easy identification and removal of the n most representative words in $W^{\ t}$ of a particular tweet.

2.10.3 Latent Topics Combination

Latent Dirichlet Allocation (LDA) is an unconventional method which checks a document model (known as a *bag of words*) as a combination of rate of occurrence of latent topics [20]. Latent topics are identified by a distribution of word probabilities which are linked to them. After the LDA analysis, a set of topics is obtained for each, a set of words and chances of emission.

LDA is applied on a body of text D composed of a vocabulary of m_W words. Firstly, a topic model is developed using a feature vector V_i^z linked with every topic z of the semantic space T^{spc}. Each i^{th} feature (i = 1,2,...,m_W) of V_i^z represents the chance of the word w_i while being aware of the topic z.

2.10.4 Buzz Ability Descriptors

It was proposed to investigate on the contribution of 3 indicators to the *Buzz* events, first indicator and the most important one is the "popularity" of words based on RSS feeds' statistical analysis. Second indicator is dependent on the chances of linking dominant themes of the *tweet* that is regarded as a measure of importance and uses unlikely theme linking as a factor for enticing the targeted population. While the last indicator examines the expressivity of the *tweet* text from a *sensitivity lexicon* stored somewhere annotated.

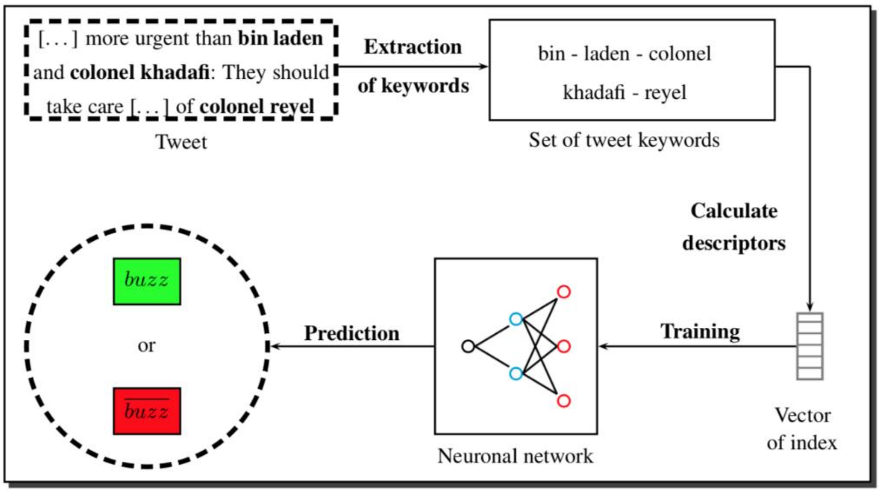


Figure 5: Architecture of the *Buzz* Prediction System [4].

The proposed method aims at examining and correctly reporting *Buzz* that is bursty events on the Twitter. 3 descriptors were evaluated individually and alone, and then combined. From the obtained results complementarity was shown. The most promising system achieved a 72 percent F-score. It is obvious that *Buzz* is a dynamic event where the prediction can be done based on models that include not only the content but also the information speed spreads. Incorporate the dynamic and/or structural area of the diffusion system could significantly worked on to improve the quality of the prediction.

Chapter 3

DATA AND METHODOLOGY

3.1 Data

3.1.1 Data Description

The used data set in this study is provided by the UC Irvine Machine Learning Repository website under the topic Buzz Prediction in Social Media (Twitter Data set) where binary classification of Buzz that is Buzz / no Buzz and the domain is Twitter is discussed. The total used data is the sample of 14706 observations over 77 attributes. Appendix A shows the description and details of used data set. Each instance covers seven days of observation for a specific topic (e.g. overclocking). Considering the fortnight following this initial observation; if there are at least 14706 additional active discussions by day, then the predicted attribute Buzz is true. Observations are Independent and identically distributed. There are 77 primary features in each instance, which are listed in Table 4.

Time representation is as follows; every instance is described by 77 features; those describe the evolution of 77 `primary features' through time. Hence every feature name is post fixed with the relative time of observation. For instance, the value of the feature `Nb_Active Discussion' at time t is given in 'Nb_Active_Discussion_t'.

Table 4: Data Description [4]

#	Categories	Explanation	Features observed
1	**Number Discussions created (NCD)**	This gives the amount of discussions created at time step t and involving a particular topic.	Columns [0,6] in Table X: NCD_0, NCD_1, NCD_2, NCD_3, NCD_4, NCD_5, NCD_6
2	**Authors interacting (AI)**	Measure of the number of new authors interacting on the instance's topic at time t (popularity)	Columns [7,13] in Table X: AI_0, AI_1, AI_2, AI_3, AI_4, AI_5, AI_6
3	**Attention measures AS(NA)**	The attention gained by a topic on a social media.	Columns [14,20] in Table X: AS(NA)_0, AS(NA)_1, AS(NA)_2, AS(NA)_3, AS(NA)_4, AS(NA)_5, AS(NA)_6
4	**Burstiness Level (BL)**	Burstiness* level of a topic.	Columns [21,27]) in Table X: BL_0, BL_1, BL_2, BL_3, BL_4, BL_5, BL_6
5	**Number of Atomic Containers (NAC)**	The total number of atomic containers generated via the whole social media on the instance's topic.	Columns [28,34] in Table X: NAC_0, NAC_1, NAC_2, NAC_3, NAC_4, NAC_5, NAC_6
6	**Attention Level (measured with number of contributions) AS(NAC)**	Measure of the attention gained by an instance's topic on a social media.	Columns [35,41] in Table X: AS(NAC)_0, AS(NAC)_1, AS(NAC)_2, AS(NAC)_3, AS(NAC)_4, AS(NAC)_5, AS(NAC)_6
7	**Contribution Sparseness measures (CS)**	The spread of contributions about discussion for the instance's topic	Columns [42,48] in Table X: CS_0, CS_1, CS_2, CS_3, CS_4, CS_5, CS_6
8	**Author Interaction measures (AT)**	Amount of authors interacting on the instance's topic within a discussion	Columns [49,55] in Table X: AT_0, AT_1, AT_2, AT_3, AT_4, AT_5, AT_6
9	**Number of Authors measures(NA)**	The number of authors interacting on the instance’s topic	Columns [56,62] in Table X: NA_0, NA_1, NA_2, NA_3, NA_4, NA_5, NA_6
10	**Average Discussions Length(ADL)**	Average Discussions Length directly measures the average length of a discussion belonging to the instance’s topic	Columns [63,69] in Table X: ADL_0, ADL_1, ADL_2, ADL_3, ADL_4, ADL_5, ADL_6
11	**Average Discussions Length(NAD)**	The number of discussions involving the instance’s topic	Columns [70,76] in Table X: NAD_0, NAD_1, NAD_2, NAD_3, NAD_4, NAD_5, NAD_6.

* In statistics, burstiness is the intermittent increases and decreases in activity or frequency of an event.

3.1.2 Case Study

In this section, a case study is considered via a scenario to explain various critical categories while buzz prediction. According to the study of [22, 23], celebrities’ death nowadays grab people’s attention for different reasons, but they are quickly forgotten as people move onto news. A as scenario for buzz prediction, Michael Jackson’s death was one of the most marked in recent history. As somebody shuts down Twitter, millions of comments flooded the internet, and one that generated and amount of media attention. This topic covers seven days of observation for this topic and is described by the evolution of 11 primary features through the time:

- The first feature shows the number of discussions created with the average of 22899 over the sample.
- The 2nd feature shows the number of new authors interacting with the average of 110.877 over the sample.
- The 3rd feature shows the measure of the high attention paid over the sample.
- The 4th feature shows the high burstiness level over the sample.
- The 5th feature shows the total number of atomic containers generated through the whole twitter with the average of 200.500 over the sample.
- The 6th feature shows the high attention paid over the sample.
- The 7th feature shows the high measure of spread of contributions over the discussion sample.
- The 8th feature shows the average amount of authors interacting with the average of 1.012 over the sample.
- The 9th feature shows the number of authors interacting with the average of 154.592over the sample.
- The 10th feature shows the average length of a discussion belonging with the average of 1.113 over the sample.
- The 11th feature shows the amount of discussions involving the topic with the average of 216.765over the sample.

3.2 Methodology

In this study, binary case (Buzz /Non-Buzz) classification of Social Media benchmark data (Twitter) by neural network is implemented. This data set has serious imbalance property. Some solutions to the class imbalance problem was proposed in the past for both at the data level and at the level of algorithm. At the data level, solutions comprise of many unique forms of resampling such as random oversampling with replacement, random under sampling and so on. At the algorithmic level, solutions comprise of adjusting the costs of the various classes, adjusting the probabilistic prediction at the tree leaf, adjusting the decision threshold and so on.

In this thesis, three different machine learning networks were applied namely, SVM, RBF and RF which are very sensitive to imbalanced data in order to perform this classification task and compared their performances. RF is implemented introducing variable rank ordering, which is an efficient strategy to detect the most significant attributes to identify the created topic is Buzz or not.

RFs are composed of tree predictors combined such that each tree depends on the values of a random vector sampled differently from others but with the same distribution for all trees in a forest. The generalization error for forests converges to a limit as the number of trees in the forest becomes large. The generalization error of a forest of tree classifiers is dependent upon the strength of each tree in the forest and the relationship between these trees. By employing a random selection of features process to break each node into two generates rates of error that are more effective because of the ability to disregard noise. Internal estimates monitor error, strength, and correlation are used to indicate the response to increasing number of features used in the process of splitting, and they also employed to measure the importance of the variable.

RF models are based on decision trees that can be employed for the purpose of classification of a discrete variable or regression of a continuous variable. Classification and Regression Tree (CART) is often used to describe these decision trees. Briefly, the RF algorithm involves randomly subsetting samples from your dataset and builds a decision tree based on these samples. At every node in the tree m_{try} (a set parameter), number of features is selected from the set of all features. The feature that provides the best split (given any preceding nodes) is chosen and then the procedure is repeated. This algorithm is run on a large number of trees, based on different sample subsets, which means that this method is less prone to over fitting than other CART methods.

Since every decision tree in the forest is only based on a subset of samples, each tree's performance can be evaluated on the left-out samples. When this validation is performed on all samples and trees in a RF, the resulting metric is called the out-of-bag error. The advantage of using this metric is that it removes the need for a test set to rate the performance of your model. Also, the out-of-bag error can be used to calculate the variable importance of all the features in the model. Variable importance is usually calculated by re-running the RF with one feature's values scrambled across all samples. This difference in accuracy between this model with the scrambled feature and the original model is one measure of variable importance.

Sometimes RF is run to perform feature selection on a dataset. This can be useful when there are thousands of features and you'd like to reduce the number to a less complex subset. However, it is important to realize that you need to validate selected features on independent data. Breiman designed the RFs [18], that adds an additional layer of randomness to bagging. RF's have changed the way classification and / or regression trees are made, and each tree, in addition to creating the data using a different bootstrap instance. In standard trees, every node is partitioned using the best partitioning among all variables when an RF is used, and every node is best partitioned using a randomly selected subset of tokens in that node. This unexpected strategy performs very well when compared to a large number of other classifiers, including discriminant analysis, support vector machines and neural networks. It is also efficient against over fitting. In addition to its user friendliness because it has only two parameters, that is number of variables in the random subset at every node and the amount of trees in the forest, and is usually not responsive to the values on each of them. The RF package has an interface for R and the FORTRAN programs developed by Breiman and Cutler [24] [25] [26].

3.2.1 Random Forest Algorithm

The RF approach for both classification and regression:

Step 1. Using the initial data, construct $\mathrm{n}_{\mathrm{tree}}$ bootstrap samples.

Step 2. Develop a raw classification or regression tree for every bootstrap sample, but apply modifications as follows: at every node, randomly sample $\mathrm{m}_{\mathrm{try}}$ of the predictors and choose the best split from among those variables instead of choosing the best split among all predictors,

Step 3. Evaluate new data by grouping the predictions of the $\mathrm{n}_{\mathrm{tree}}$using majority votes for classification, and average for regression.

Error rate can be estimated, by the following:

Step 1. At every bootstrap cycle, predict the data not in the bootstrap sample called “out-of-bag”, or OOB, data) using the tree developed with the bootstrap sample.

Step 2. Group the OOB predictions. Calculate the error rate and call it the OOB estimate of error rate. It was found that the OOB estimation of rate of error largely correct, provided that enough trees have been grown (otherwise the OOB estimate can bias [27].

3.2.2 Variable Importance Measures

The Random Forest package generates 2 more pieces of information optionally: these are; a measure of the significance of the variables of prediction, and internal structure measure which could include the closeness of different data points to others). Importance of variable is a difficult mechanism to define given that the significance of a variable may be due to the interaction it has with other variables.

There are 2 very useful other products of RF: out-of-bag estimates of generalization error [18], [27] and variable importance measures [25] [28]. Liaw and Wiener worked on 2 methodologies for calculating variable importance measures in the random Forest R package, which differ in some ways from the four heuristics originally suggested for variable importance measures [24].

The first heuristic is based on the Gini criterion. To be specific, at each split the decrease in the Gini node impurity is recorded for the variable xj that was used to form the split. The average of all decreases in the Gini impurity in the forest where xj forms the split yields the Gini variable importance measure Δxj.

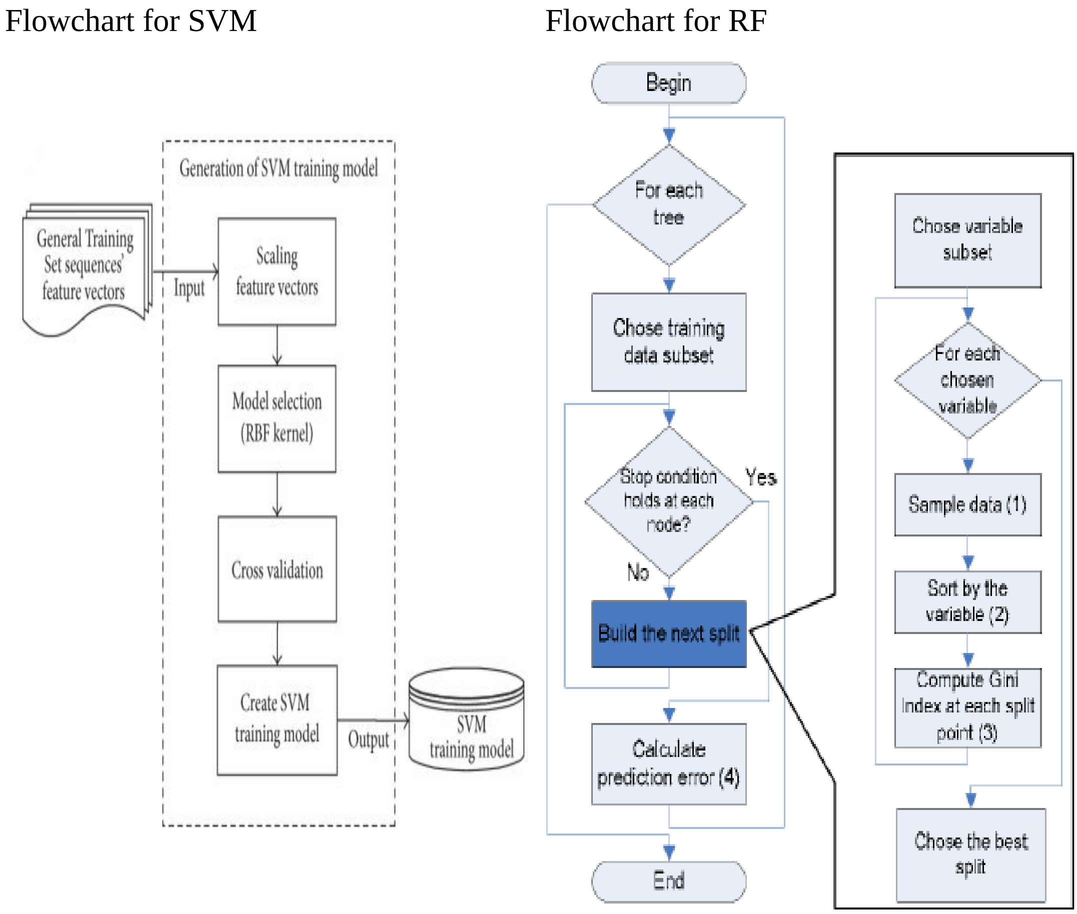
The random forest algorithm estimates the importance of a variable by looking at degree to which the guessing error goes up whenever OOB data for that variable is altered while all others are remain untouched. Computation required are carried out tree after tree as the random forest is generated. There are actually 4 different measures of variable importance which are modelled in the classification code. Refer to [29] for more definitions.

3.2.3 Proximity Measure

The (i, j) components of the closeness matrix generated by RF is the decimal part of trees in which elements i and j coming in the same ending node. The premise is that “similar” observations should be located in the same terminal nodes at most occurrences than those different from each. The proximity matrix may be employed to detect structure in the dataset (see [29]) and/or for random forests using unsupervised learning [25].

3.2.4 Usage in R

The user interfaces to RF in accord with that of other classification functions like the NNET [30] [31] and SVM (in the e1071 package) [32]. There is a formula interface, and predictors can be specified as a matrix or data frame via the x argument, with responses as a vector via the y argument. RF carries out classification process if the response is a factor, that is, the response is not continuous; if the response not a factor (that is, not a factor), RF carries out regression process. RF carries out unsupervised learning whenever the response is not specified. At the moment, RF does not handle statistically categorical responses. Note that categorical predictor variables must also be specified as factors so that they are not wrongly treated as continuous). The RF function returns an object of class "Random Forest". Explanation about the elements of such object are given in documentation available online. Methods given for the class comprise of predict and print. As we mentioned before in order to predict the Buzz events on Twitter, we utilized two different classes of advanced valuation techniques namely SVM and RF which the pseudo code in shown in Figure 6.


Flowchart for SVM

Flowchart for RF

Generation of SVM training model
General Training Set sequences' feature vectors
Input
Scaling feature vectors
Model selection (RBF kernel)
Cross validation
Create SVM training model
Output
SVM training model
Begin
For each tree
Chose training data subset
Stop condition holds at each node?
Yes
No
Build the next split
Calculate prediction error (4)
End
Chose variable subset
For each chosen variable
Sample data (1)
Sort by the variable (2)
Compute Gini index at each split point (3)
Chose the best split

Figure 6: SVM and RF Flowcharts [5]

Chapter 4

EMPIRICAL FINDINGS

In this section, firstly the results of some other research studies are given that are discussed with details in literature section. As we mentioned in literature section, the studies [4][5] used only 2000 samples of the focused dataset. Machine learning are studied using SVM with different kernels namely, linear, polynomial with degree 3, polynomial with degree 4, and RBF are used and their results shown in Table 5.

Table 5: Estimation Results

Kernel	References	Accuracy	Training set	Testing set
Linear	[5]	0.927	1000	1000
RBF	[4]	0.958	1000	1000
Polynomial 3	[5]	0.92	1000	1000
Polynomial 4	[5]	0.923	1000	1000

On the other hand, this study applied three different machine learning networks which are: RBF, RF and SVM with three different kernels. The kernel types considered are Polynomial, Radial-Linear and Sigmoid. In addition, we used same dataset with entire samples to improve accuracy. Therefore, the test and train sets used in this study contain all 14706 observations over 77 attributes. Table 6 represents our results with the error, accuracy, the numbers of the test and training sets used while applying the three machine learning networks.

Table 6: Error Accuracy

Type	Error	Accuracy	Training set	Testing set
SVM-RBF	0.341	0.66	9804	4902
SVM-Linear	0.361	0.64	9804	4902
SVM-Polynomial	0.762	0.14	9804	4902
RBF	0.061	0.94	9804	4902
RF	0.001	0.99	9804	4902

The error parameter is calculated as 1.00 minus the corresponding accuracy in Table 6. The related obtained error and accuracy are also shown in Figures 8 and 9 as follow:

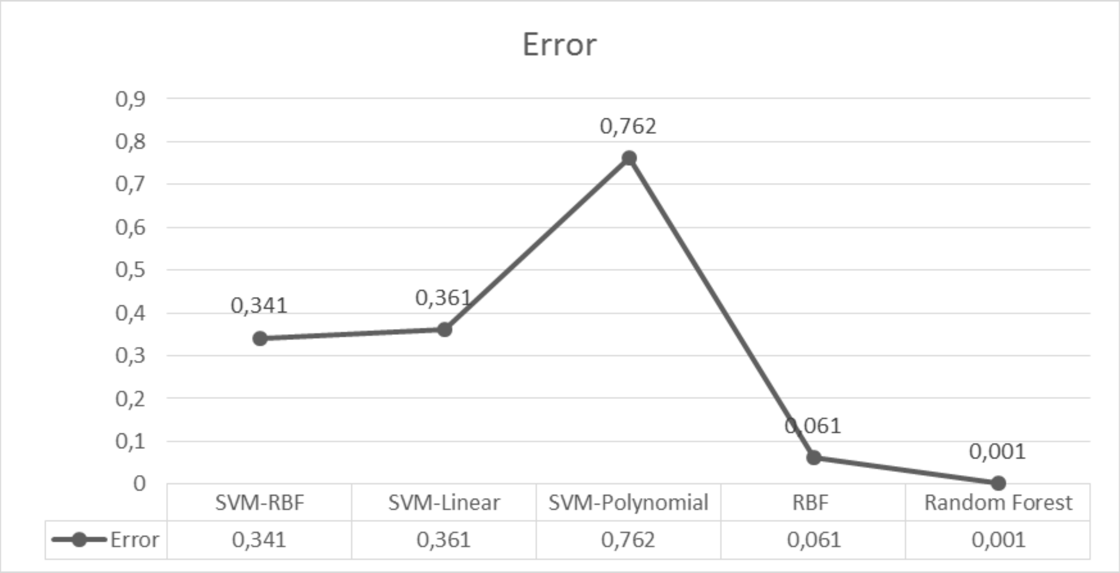
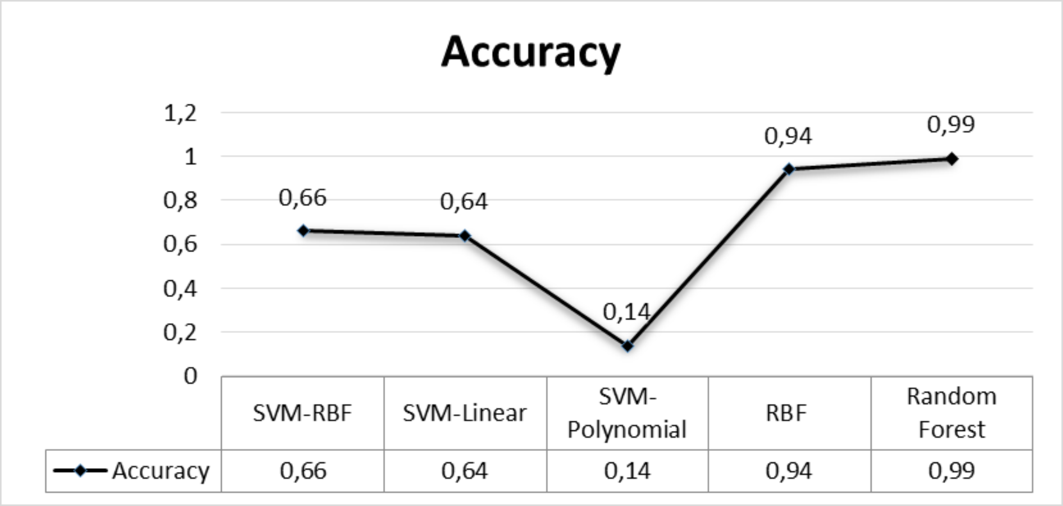


Figure 7: RMSES Error



	SVM-RBF	SVM-Linear	SVM-Polynomial	RBF	Random Forest
Accuracy	0,66	0,64	0,14	0,94	0,99

Figure 8: Accuracy

Conclusively, Figure 10 infer that using RF helps also to detect the most significant attributes in order to identify the created Buzz and to illustrate if a discussion is Buzz or not.

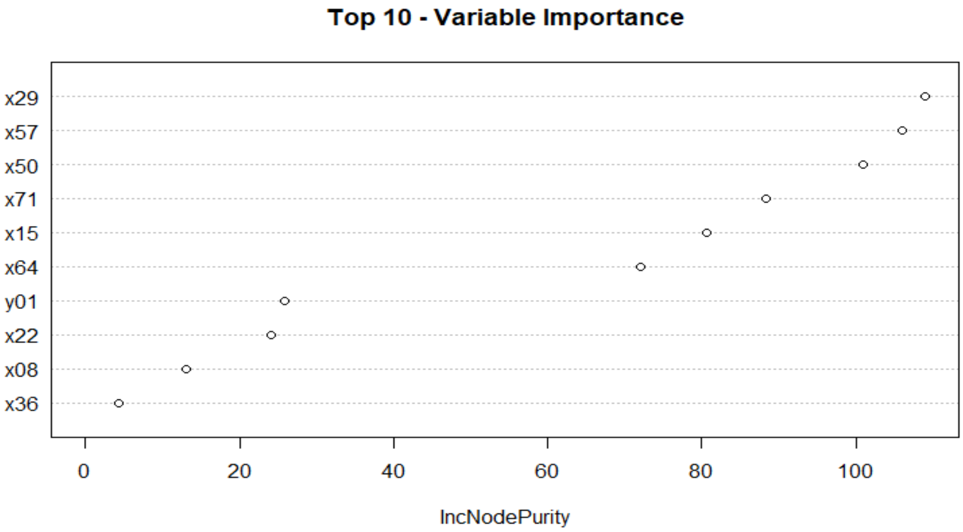


Figure 9: Variable Importance

All in all, as shown in the Figure 10, RF outperformed other machine learning classes as it has lower RMSE relative to the other approaches. Moreover, the variable rank ordering for Buzz prediction (Figure 10) shows that the top variables have more significant effect on the accurate prediction out of the total R.H.S variables. Appendix A shows the description of used data set.

Chapter 5

CONCLUSION

Good management of the social media monitoring process contributes to effective plans in social networks. Knowing what potential customers are talking about a product brand, about sharing trends, and communicating with them is crucial in terms of marketing strategies. Considering product users' comments on the social media always gives positive results for the potential customer. With the power of social media, you can be successful about a product or service, including talking to a group by starting a conversation about a sector. This will be beneficial in raising the brand perception of customers.

It is also important to keep track of the results of campaigns and other advertisements that a brand has made over social media and to notice about the adverse effects of that campaign on the positive side. This information is actually a measure of how your brand is positioned in the eyes of users and customers. Therefore, as it is known that the recent social media takes place in a wide variety of contexts and sizes. The vast majority of messages on social media do not lead to debate. Some of these messages trigger trends and some become viruses. Thus, early detection of Buzz in a short period of time can help alleviate or prevent the negative consequences of social media outbreaks against companies or individuals. It could give them a chance to react early.

In this thesis, Buzz prediction on Twitter, a social media platform, is considered through the use of Random Forest (RF) algorithm. The performance of this method was evaluated and compared to the performances of two other similar algorithms which are Support Vector Machine (SVM) in three different kernels, and Radial Basis Function (RBF). Results from the analysis showed that RF has the overall best results in terms of accuracy and fastest training time among the other studied methods.

Additionally, RF was implemented with variable ranking feature that identified features that are more important than others. This new feature is particularly unique and will help to describe Buzz activities more accurately and further research in this area of research.

The performance of the algorithms was evaluated by using the same dataset that were divided into two groups: training data and testing data. They are used respectively in terms of training the system and subsequent testing for the accuracy of the Buzz prediction in the implementation phase of this study.

According to evaluations, RF achieved 99% accuracy, thus it can be concluded that it is the best of the three algorithms namely, Support Vector Machine (SVM), Radial Basis Function (RBF) and the Random Forest (RF). It is also much faster and more reliable for Buzz prediction on social-media. The experiments also proved that more precise or more accurate result can be obtained with increased training data and testing data. Future works of this thesis is expected to a look at the Buzz activities in other social media platforms other than Twitter. In addition, Tuning Random Forest algorithm can be used to retrieve better performance. The potential economic, social and political benefits of recent researches in this field can also be considered.

REFERENCES

[1] Kwak, H., Lee, C., Park, H., & Moon, S. (2010, April). What is Twitter, a social network or a news media? In Proceedings of the 19th international conference on World wide web (pp. 591-600). ACM.

[2] Mayuri, M. Sneha, M. L., Kamatchi Priya, international Conference on Interdisciplinary Engineering and Sustainable Management Sciences 2015, Prediction of Buzz in Social-media using Radial Basis Function Neural Networks.

[3] Aswani, R., Ghrera, S. P., Kar, A. K., & Chandra, S. (2017). Identifying Buzz in social media: a hybrid approach using artificial bee colony and k-nearest neighbors for outlier detection. Social Network Analysis and Mining, 7(1), 38.

[4] Morchid, M., Linares, G., & Dufour, R. (2014, May). Characterizing and Predicting Bursty Events: The Buzz Case Study on Twitter. In LREC (pp. 2766-2771).

[5] Rastogi, M., & Bist, A. S. (2016). Analysis of Twitter Data with Machine Learning Techniques. International Journal of Engineering Sciences & Research Technology

[6] Breiman, L., & Wald Lecture, I. I. (2002). Looking inside the black box. Wald Lecture II, Department of Statistics, California University.

[7] Hausmann, A. (2012). Creating ‘buzz’: opportunities and limitations of social media for arts institutions and their viral marketing. *International Journal of Nonprofit and Voluntary Sector Marketing*, 17(3), pp.173-182.

[8] Batra, R., Ramaswamy, V., Alden, D., Steenkamp, J. And Ramachander, S. (2000). Effects of Brand Local and Nonlocal Origin on Consumer Attitudes in Developing Countries. *Journal of Consumer Psychology*, 9(2), pp.83-95.

[9] Karaboga, D. and Basturk, B. (2008). On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing*, 8(1), pp.687-697.

[10] Karaboga, D. and Ozturk, C. (2011). A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Applied Soft Computing*, 11(1), pp.652-657.

[11] Kar, A. (2016). Bio inspired computing – A review of algorithms and scope of applications. *Expert Systems with Applications*, 59, pp.20-32.

[12] Karaboga, D. and B. Basturk, 2007a. A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm. J. Global Optim., 39: 459-471

[13] Karaboga, D. and B. Akay, 2009. A comparative study of artificial bee colony algorithm. Applied Math. Comput., 214: 108-132

[14] Karaboga, D. and Gorkemli, B. (2014). A quick artificial bee colony (qABC) algorithm and its performance on optimization problems. *Applied Soft*

Computing, 23, pp.227-238.

[15] Karaboga, D. and B. Basturk, 2007b. Artificial Bee Colony (ABC) optimization algorithm for solving constrained optimization problem. Proceedings of the 12th International Fuzzy Systems Association World Congress, June 18-21, 2007, Cancun, Mexico, pp: 789-798.

[16] Vapnik, V., Golowich, S. and Smola, A. Support vector method for function approximation, regression estimation, and signal processing. *Advances in Neural Information Processing Systems*, 9:281–287, 1996.

[17] Chen, X., Chen, C. and Jin, L. (2011). Principal Component Analyses in Anthropological Genetics. *Advances in Anthropology*, 01(02), pp.9-14.

[18] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[19] De Choudhury, M., Sundaram, H., John, A., & Seligmann, D. D. (2009, August). Social synchrony: Predicting mimicry of user actions in online social media. In *Computational Science and Engineering, 2009. CSE'09. International Conference on*(Vol. 4, pp. 151-158). IEEE.

[20] Salton, G. (1989). ABSTRACTS (Chosen by G. Salton from recent issues of journals in the retrieval area.). *ACM SIGIR Forum*, 23(3-4), pp.123-138.

[21] D. Blei, A. Ng. and M. Jordan. Latent dirichlet allocation. Journal of Machine Learning Research, 3:993-1022. 2003.

[22] Austin, B. J. (2014). Celebrities, drinks, and drugs: a critical discourse analysis of celebrity substance abuse as portrayed in the New York times.

[23] Synthesio. (2011, April) Predicting Online Buzz and Audience In The Next Step in New Market Research

[24] Breiman, L., & Cutler, A. (2003). Manual for Setting Up. Using, and Understanding Random Forest, 4.

[25] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

[26] Chang, Y., Yamada, M., Ortega, A., & Liu, Y. (2014, December). Ups and downs in Buzz es: Life cycle

[27] Bylander, T. (2002). Estimating generalization error on two-class datasets using out-of-bag estimates. Machine Learning, 48(1-3), 287-297.

[28] Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random forest: a classification and regression tool for compound classification and QSAR modeling. Journal of chemical information and computer sciences, 43(6), 1947-1958

[29] Breiman, L., & Wald Lecture, I. I. (2002). Looking inside the black box. Wald Lecture II, Department of Statistics, California University.

[30] Venables, W. N., & Ripley, B. D. (2013). Modern applied statistics with S-PLUS. Springer Science & Business Media.

[31] Ripley, B., Venables, W., & Ripley, M. B. (2016). Package ‘nnet’. R package version, 7-3

[32] Dimitriadou, E., Hornik, K., Leisch, F., Meyer, D., & Weingessel, A. The e1071 package, 2005. Software available at< http://cran. r-project. org/src/contrib/Descriptions/e1071. html.

[33] Jiang, P., Wu, H., Wang, W., Ma, W., Sun, X., & Lu, Z. (2007). MiPred: classification of real and pseudo microRNA precursors using random forest prediction model with combined features. Nucleic acids research, 35(suppl_2), W339-W344.

[34] Zumel, N., Mount, J., & Porzak, J. (2014). Practical data science with R (pp. 101-104). Manning.

[35] Papadimitriou, A., Symeonidis, P., & Manolopoulos, Y. (2012). Fast and accurate link prediction in social networking systems. Journal of Systems and Software, 85(9), 2119-2132.

[36] Chang, Y., Yamada, M., Ortega, A., & Liu, Y. (2016). Lifecycle Modeling for Buzz Temporal Pattern Discovery. ACM Transactions on Knowledge Discovery from Data (TKDD), 11(2), 20.

[37] Biau, G., Scornet, E., & Welbl, J. (2016). Neural random forests. arXiv preprint arXiv:1604.07143.

[38] Strobl, C., Boulesteix, A. L., Zeileis, A., & Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, *8*(1), 25.

[39] Archer, K. J., & Kimes, R. V. (2008). Empirical characterization of random forest variable importance measures. *Computational Statistics & Data Analysis*, *52*(4), 2249-2260.

[40] Mccord, M., & Chuah, M. (2011, September). Spam detection on twitter using traditional classifiers. In *international conference on Autonomic and trusted computing* (pp. 175-186). Springer, Berlin, Heidelberg.

[41] Wee, L. J., Simarmata, D., Kam, Y. W., Ng, L. F., & Tong, J. C. (2010, December). SVM-based prediction of linear B-cell epitopes using Bayes Feature Extraction. In *BMC genomics* (Vol. 11, No. 4, p. S21). BioMed Central.

APPENDICES

Appendix A: The Description of Used Data Set

```
1. Title of Database: Buzz prediction on Twitter - Relative
Labeling - Threshold Sigma equals 1000


2. Sources:
   -- Creators :
        FranÃƒÂ§ois Kawala (1,2) and
        Ahlame Douzal (1) and
        Eric Gaussier (1) and
        Eustache Diemert (2)

   -- Institutions :
        (1) UniversitÃƒÂ© Joseph Fourier (Grenoble I)
            Laboratoire d'informatique de Grenoble (LIG)
        (2) BestofMedia Group

   -- Donor: BestofMedia (ediemert@bestofmedia.com)
   -- Date: May, 2013


3. Past Usage:
   -- References :
        Predicting Buzz Magnitude in Social Media (in
submission (ECML-PKDD 13))

   -- Predicted attribute :
        Buzz. This attribute is boolean: 1 meaning `buzz
observed', 0 meaning
        `no buzz observed'. It is stored is the rightmost
column.

   -- Study results :
        The results achieved are acceptable, nevertheless the
unbalanced nature
        of this dataset leaves some room for improvement.
Using random forest
        yields a F-1 score of around 0.65 for the Buzz class,
when the data is
        scaled and normalized. First order discrete difference
over features may also
        be considered as additional features.

4. Relevant Information Paragraph:
   -- Observations :
        Each instance covers seven days of observation for a
specific topic (eg.
        overclocking...). Considering the couple day following
this initial
        observation; If there is at least 500 additional
active discussions by
        day (on average, with respect to the initial
observation) then, the
        predicted attribute Buzz is True.
        Observations are Independent and identically
distributed.


5. Number of Instances
   -- Total number of instances : 140 707


6. Number of Attributes
   -- Total number of attributes : 77.

   -- Time representation :
        Each instance is described by 77 features, those
describe the evolution
        of 11 `primary features' through time. Hence each
feature name is
        postfixed with the relative time of observation. For
instance, the value
        of the feature `Nb_Active_Discussion' at time t is
given in
        'Nb_Active_Discussion_t'.


7. Attributes

    -- Number of Created Discussions (NCD) (columns [0,6])

       -- Type : Numeric, integers only
       -- Description : This feature measures the number of
discussions created
          at time step t and involving the instance's topic.
       -- Columns : From column 0 (NCD at relative time 0) to
column 6 (NCD at
          relative time 6)
       -- Abbreviations : NCD_0, NCD_1, NCD_2, NCD_3, NCD_4,
NCD_5, NCD_6
       -- Statistics :
          +---------+-----+-------+---------+---------+
          | feature | min | max   | mean    | std     |
          +---------+-----+-------+---------+---------+
          | NCD_0   | 0   | 24210 | 172.267 | 509.768 |
          +---------+-----+-------+---------+---------+
          | NCD_1   | 0   | 22899 | 155.135 | 471.615 |
          +---------+-----+-------+---------+---------+
          | NCD_2   | 0   | 20495 | 165.459 | 495.287 |
          +---------+-----+-------+---------+---------+
          | NCD_3   | 0   | 27007 | 176.811 | 528.350 |
          +---------+-----+-------+---------+---------+
          | NCD_4   | 0   | 30957 | 186.929 | 560.329 |
          +---------+-----+-------+---------+---------+
          | NCD_5   | 0   | 28603 | 216.197 | 632.107 |
          +---------+-----+-------+---------+---------+
          | NCD_6   | 0   | 37505 | 243.856 | 707.354 |
          +---------+-----+-------+---------+---------+
    -- Author Increase (AI) (columns [7,13])

       -- Type : Numeric, integers only
       -- Description : This featurethe number of new authors
interacting on
          the instance's topic at time t (i.e. its popularity)
       -- Columns : From column 7 (AI at relative time 0) to
column 13 (AI at
          relative time 6)
       -- Abbreviations : AI_0, AI_1, AI_2, AI_3, AI_4, AI_5,
AI_6
       -- Statistics :
          +---------+-----+-------+---------+---------+
          | feature | min | max   | mean    | std     |
          +---------+-----+-------+---------+---------+
          | AI_0    | 0   | 15105 | 87.050  | 234.733 |
          +---------+-----+-------+---------+---------+
          | AI_1    | 0   | 15730 | 78.639  | 218.438 |
          +---------+-----+-------+---------+---------+
          | AI_2    | 0   | 16389 | 84.270  | 233.560 |
          +---------+-----+-------+---------+---------+
          | AI_3    | 0   | 17445 | 90.534  | 249.850 |
          +---------+-----+-------+---------+---------+
          | AI_4    | 0   | 18654 | 95.750  | 262.838 |
          +---------+-----+-------+---------+---------+
          | AI_5    | 0   | 22035 | 110.877 | 295.251 |
          +---------+-----+-------+---------+---------+
          | AI_6    | 0   | 29402 | 127.184 | 342.008 |
          +---------+-----+-------+---------+---------+

    -- Attention Level (measured with number of authors)
(AS(NA))
       (columns [14,20])

       -- Type : Numeric, real in [0,1]
       -- Description : This feature is a measure of the
attention payed to a
          the instance's topic on a social media.
       -- Columns : From column 14 (AS(NA) at relative time 0)
to column 20 (AS(NA)
          at relative time 6)
       -- Abbreviations : AS(NA)_0, AS(NA)_1, AS(NA)_2,
AS(NA)_3, AS(NA)_4,
          AS(NA)_5, AS(NA)_6
       -- Statistics :
          +----------+-----+-------+-------+-------+
          | feature  | min | max   | mean  | std   |
          +----------+-----+-------+-------+-------+
          | AS(NA)_0 | 0   | 0.025 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+
          | AS(NA)_1 | 0   | 0.022 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+
          | AS(NA)_2 | 0   | 0.024 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+
          | AS(NA)_3 | 0   | 0.025 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+
          | AS(NA)_4 | 0   | 0.027 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+
          | AS(NA)_5 | 0   | 0.029 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+
          | AS(NA)_6 | 0   | 0.040 | 0.000 | 0.001 |
          +----------+-----+-------+-------+-------+



    -- Burstiness Level (BL) (columns [21,27])

       -- Type : Numeric, defined on [0,1]
       -- Description : The burstiness level for a topic z at
a time t is
          defined as the ratio of ncd and nad
       -- Columns : From column 21 (BL at relative time 0) to
column 27 (BL at
          relative time 6)
       -- Abbreviations : BL_0, BL_1, BL_2, BL_3, BL_4, BL_5,
BL_6
       -- Statistics :
          +---------+-----+-----+-------+-------+
          | feature | min | max | mean  | std   |
          +---------+-----+-----+-------+-------+
          | BL_0    | 0   | 1   | 0.901 | 0.292 |
          +---------+-----+-----+-------+-------+
          | BL_1    | 0   | 1   | 0.909 | 0.281 |
          +---------+-----+-----+-------+-------+
          | BL_2    | 0   | 1   | 0.872 | 0.329 |
          +---------+-----+-----+-------+-------+
          | BL_3    | 0   | 1   | 0.885 | 0.314 |
          +---------+-----+-----+-------+-------+
          | BL_4    | 0   | 1   | 0.890 | 0.308 |
          +---------+-----+-----+-------+-------+
          | BL_5    | 0   | 1   | 0.929 | 0.250 |
          +---------+-----+-----+-------+-------+
          | BL_6    | 0   | 1   | 0.955 | 0.199 |
          +---------+-----+-----+-------+-------+



    -- Number of Atomic Containers (NAC) (columns [28,34])

       -- Type : Numeric, integer
       -- Description : This feature measures the total number
of atomic
          containers generated through the whole social media
on the instance's topic until time t.
       -- Columns : From column 28 (NAC at relative time 0) to
column 34 (NAC at
          relative time 6)
       -- Abbreviations : NAC_0, NAC_1, NAC_2, NAC_3, NAC_4,
NAC_5, NAC_6
       -- Statistics :
          +---------+-----+-------+---------+---------+
          | feature | min | max   | mean    | std     |
          +---------+-----+-------+---------+---------+
          | NAC_0   | 0   | 26644 | 184.746 | 536.961 |
          +---------+-----+-------+---------+---------+
          | NAC_1   | 0   | 25228 | 166.159 | 494.900 |
          +---------+-----+-------+---------+---------+
          | NAC_2   | 0   | 22065 | 177.286 | 520.721 |
          +---------+-----+-------+---------+---------+
          | NAC_3   | 0   | 30592 | 189.778 | 556.903 |
          +---------+-----+-------+---------+---------+
          | NAC_4   | 0   | 35089 | 200.500 | 589.702 |
          +---------+-----+-------+---------+---------+
          | NAC_5   | 0   | 32289 | 232.445 | 664.037 |
          +---------+-----+-------+---------+---------+
          | NAC_6   | 0   | 37505 | 262.269 | 740.397 |
          +---------+-----+-------+---------+---------+


    -- Attention Level (measured with number of contributions)
(AS(NAC))
       (columns [35,41])

       -- Type : Numeric, real in [0,1]
       -- Description : This feature is a measure of the
attention payed to a
          the instance's topic on a social media.
       -- Columns : From column 35 (AS(NA) at relative time 0)
to column 42
          (AS(NAC) at relative time 6)
       -- Abbreviations : AS(NAC)_0, AS(NAC)_1, AS(NAC)_2,
AS(NAC)_3, AS(NAC)_4,
          AS(NAC)_5, AS(NAC)_6
       -- Statistics :
          +-----------+-----+-------+-------+-------+
          | feature   | min | max   | mean  | std   |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_0 | 0   | 0.021 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_1 | 0   | 0.022 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_2 | 0   | 0.017 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_3 | 0   | 0.015 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_4 | 0   | 0.017 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_5 | 0   | 0.022 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+
          | AS(NAC)_6 | 0   | 0.022 | 0.000 | 0.000 |
          +-----------+-----+-------+-------+-------+


    -- Contribution Sparseness (CS) (columns [42,48])

       -- Type : Numeric, real in [0,1]
       -- Description : This feature is a measure of spreading
of contributions
```

```
          over discussion for the instance's topic at time t.
       -- Columns : From column 42 (CS at relative time 0) to
column 48
          (CS at relative time 6)
       -- Abbreviations : CS_0, CS_1, CS_2, CS_3, CS_4, CS_5,
CS_6
       -- Statistics :
          +---------+-----+-----+-------+-------+
          | feature | min | max | mean  | std   |
          +---------+-----+-----+-------+-------+
          | CS_0    | 0   | 1   | 0.907 | 0.291 |
          +---------+-----+-----+-------+-------+
          | CS_1    | 0   | 1   | 0.914 | 0.280 |
          +---------+-----+-----+-------+-------+
          | CS_2    | 0   | 1   | 0.876 | 0.329 |
          +---------+-----+-----+-------+-------+
          | CS_3    | 0   | 1   | 0.890 | 0.313 |
          +---------+-----+-----+-------+-------+
          | CS_4    | 0   | 1   | 0.894 | 0.307 |
          +---------+-----+-----+-------+-------+
          | CS_5    | 0   | 1   | 0.934 | 0.249 |
          +---------+-----+-----+-------+-------+
          | CS_6    | 0   | 1   | 0.960 | 0.196 |
          +---------+-----+-----+-------+-------+


    -- Author Interaction (AT) (columns [49,55])

       -- Type : Numeric, integer.
       -- Description : This feature measures the average
number of authors
          interacting on the instance's topic within a
discussion.
       -- Columns : From column 49 (AT at relative time 0) to
column 55
          (AT at relative time 6)
       -- Abbreviations : AT_0, AT_1, AT_2, AT_3, AT_4, AT_5,
AT_6
       -- Statistics :
          +---------+-----+-----+-------+-------+
          | feature | min | max | mean  | std   |
          +---------+-----+-----+-------+-------+
          | AT_0    | 0   | 175 | 1.013 | 1.124 |
          +---------+-----+-----+-------+-------+
          | AT_1    | 0   | 177 | 1.012 | 1.308 |
          +---------+-----+-----+-------+-------+
          | AT_2    | 0   | 177 | 0.973 | 1.253 |
          +---------+-----+-----+-------+-------+
          | AT_3    | 0   | 178 | 0.989 | 1.124 |
          +---------+-----+-----+-------+-------+
          | AT_4    | 0   | 282 | 0.997 | 1.421 |
          +---------+-----+-----+-------+-------+
          | AT_5    | 0   | 176 | 1.052 | 1.243 |
          +---------+-----+-----+-------+-------+
          | AT_6    | 0   | 283 | 1.113 | 1.648 |
          +---------+-----+-----+-------+-------+
    -- Number of Authors (NA) (columns [56,62])

       -- Type : Numeric, integer.
       -- Description : This feature measures the number of
authors interacting
          on the instance's topic at time t.
       -- Columns : From column 49 (NA at relative time 0) to
column 55 (NA at
          relative time 6)
       -- Abbreviations : NA_0, NA_1, NA_2, NA_3, NA_4, NA_5,
NA_6
       -- Statistics :
          +---------+-----+-------+---------+---------+
          | feature | min | max   | mean    | std     |
          +---------+-----+-------+---------+---------+
          | NA_0    | 0   | 21723 | 150.690 | 417.139 |
          +---------+-----+-------+---------+---------+
          | NA_1    | 0   | 20594 | 135.635 | 383.109 |
          +---------+-----+-------+---------+---------+
          | NA_2    | 0   | 18800 | 144.479 | 407.611 |
          +---------+-----+-------+---------+---------+
          | NA_3    | 0   | 24156 | 154.592 | 436.318 |
          +---------+-----+-------+---------+---------+
          | NA_4    | 0   | 28133 | 163.159 | 457.828 |
          +---------+-----+-------+---------+---------+
          | NA_5    | 0   | 26705 | 188.250 | 512.333 |
          +---------+-----+-------+---------+---------+
          | NA_6    | 0   | 34085 | 211.736 | 571.083 |
          +---------+-----+-------+---------+---------+


    -- Average Discussions Length (ADL) (columns [63,69])

       -- Type : Numeric, real.
       -- Description : This feature directly measures the
average length of a
          discussion belonging to the instance's topic.
       -- Columns : From column 63 (ADL at relative time 0) to
column 69 (ADL at
          relative time 6)
       -- Abbreviations : ADL_0, ADL_1, ADL_2, ADL_3, ADL_4,
ADL_5, ADL_6
       -- Statistics :
          +---------+-----+---------+-------+-------+
          | feature | min | max     | mean  | std   |
          +---------+-----+---------+-------+-------+
          | ADL_0   | 0   | 180     | 1.058 | 1.235 |
          +---------+-----+---------+-------+-------+
          | ADL_1   | 0   | 182     | 1.051 | 1.404 |
          +---------+-----+---------+-------+-------+
          | ADL_2   | 0   | 182     | 1.017 | 1.344 |
          +---------+-----+---------+-------+-------+
          | ADL_3   | 0   | 183     | 1.036 | 1.226 |
          +---------+-----+---------+-------+-------+
```

ADL_4	0	294	1.045	1.520
ADL_5	0	185.667	1.113	1.374
ADL_6	0	295	1.196	1.826

-- Average Discussions Length (NAD) (columns [70,76])

-- Type : Numeric, integer.

-- Description : This features measures the number of discussions involving the instance's topic until time t.

-- Columns : From column 70 (NAD at relative time 0) to column 76 (NAD at relative time 6)

-- Abbreviations : NAD_0, NAD_1, NAD_2, NAD_3, NAD_4, NAD_5, NAD_6

-- Statistics :

feature	min	max	mean	std
NAD_0	0	24301	172.827	510.902
NAD_1	0	22980	155.616	472.512
NAD_2	0	20495	165.932	496.151
NAD_3	0	27071	177.304	529.269
NAD_4	0	31028	187.453	561.277
NAD_5	0	28697	216.765	633.118
NAD_6	0	37505	244.467	708.367

-- Annotation (column 77)

-- Type : Numeric, integer: 0 or 1

-- Description : See 3. and 4.

-- Columns : 77

-- Buzz = 1
Non Buzz = 0

8. Missing Attribute Values:

-- There is not any missing values.

9. Class Distribution:

-- Positives instances (ie. Buzz) : 1177 (0.83 %)

-- Negative instances (ie. Non Buzz) : 139530 (99.16 %)

10. CLASSIFICATION TASK

In the classification task you will be provided with time-windows showing an upward trend. The objective of this task is to determine whether or not these time-windows are followed by buzz events. In this task:

Each example matches an upward window. Such an example is a multivariate time-series ranging from t to t+Î².

The labeling (ie. buzz; non-buzz) of an example, as well as the upward detection, are performed considering an univariate time-series. This time series (Y, the target feature, presented bellow) is meant to reflect the popularity of a topic.

There is two ways to label examples: Absolute labeling and Relative labeling. the second one is based on the increment of popularity level before and after Î²

For both of these labeling methods, the threshold value Ïƒ varies in order to qualify buzz of distinct magnitude. Concretely Ïƒ = 500 implies that an example is labeled as a buzz if:

(Relative labeling) the difference between (a) the Yâ€™s mean value between t+Î²+1 and t+Î²+Î´ and (b) the Yâ€™s mean value between between t to t+Î² is greater than 500