Opinion Mining and Sentiment Analysis on a Twitter Data Stream

BalakrishnanGokulakrishnan*[1], Pavalanathan Priyanthan*[2], ThiruchittampalamRagavan*[3],
Nadarajah Prasath*[4], AShehan Perera*[5]

**Department of Computer Science and Engineering, University of Moratuwa,
Moratuwa, Sri Lanka*

[1]royalgok@gmail.com, [2]priyanthanuom@gmail.com, [3]sktragu@gmail.com, [4]n.prasath.002@gmail.com,
[5]shehan@uom.lk

Abstract — **Opinion mining and sentiment analysis is a fast growing topic with various world applications, from polls to advertisement placement. Traditionally individuals gather feedback from their friends or relatives beforepurchasing an item, but today the trend is to identify the opinions of a variety ofindividuals around the globe using microbloggingdata. This paper discusses an approach where a publicised stream of tweets from the Twitter microblogging site are preprocessed and classified based on their emotional content as positive, negative and irrelevant; and analyses the performance of various classifying algorithms based on their precision and recall in such cases. Further, the paper exemplifies the applications of this research and its limitations.**

Keywords— **Twitter, Classification algorithms, Data mining, Data preprocessing, Machine learning**

I. Introduction

Twitter has become a popular micro blogging service that has a large and rapidly growing user base where users create status messages called tweets. Users use these tweets not only as a means for updating what is on their mind, but also to express their opinion towards products, services, events and other Twitter usersthey are interested in. Pang et al. in [1] outline many cases where opinions expressed by Twitter users are useful in real world situations, such as product/service reviews on restaurants, electronics, hotels etc. Through performing a sentiment analysis on such tweets, marketers should be able to determine the public perception of their products and services; while consumers can know in advance what the other users think of the product/service they are interested in. Twitter, which is sometimes called a subscribe-and-publish social network, provides directed links among users; and hosts emotion-rich information across a wide set of users and topics. Thus, it is evident that mining user opinions and sentiments from Twitter will be very useful for many applications. The main objective of this research is to analyse the effectiveness of various popular classifiers and identify the more suitable classifier(s) for Twitter that could ease the process of classifying sentiments in tweets.

A sentiment can be defined as a personal positive or negative feeling. Opinion mining is the computational technique for extracting, classifying, understanding, and assessing the opinions expressed in various contents [2]. There has been a significant amount of research conducted in the area of sentiment classification, most of which has initially focused on classifying movie reviews [3] and blog posts [4]. Pang.et al in [3] have used different classifiers in order to classify movie reviews and compared the performance of those classifiers.

This paper focuses on identifying classifiers with an acceptable performance that could be used to classify tweets based on the expressed sentiment as *neutral*, *polar* and *irrelevant*; and then *polar* into either *positive* or *negative*. Even though there have been a number of researches which have attempted to classify tweets based on the sentiment stemming from various aspects, an accuracy of 80 per cent or more has never been achieved [5]. Therefore this area attracts many researches which focus on improving the accuracy and performance of classification techniques.

Sentiment analysis in Twitter is a different paradigm compared to other researches that attempt sentiment analysis through machine learning [6]. This is due to constraints which are present in identifying the sentiments expressed in tweets. Due the limitation of 140 characters, people frequently use shortened forms and abbreviations which could bring different interpretations in different contexts. Further, slang usage and sentences with dubious grammar increase the requirement for preprocessing exponentially. Due to such limitations in natural language processing, this issue becomes a bottleneck in increasing the accuracy of the results. Ambiguity is another factor that degrades accuracy. For example, people commonly use the word ‘Nike’ to refer to both the popular brand of apparels and accessories as well as the Greek goddess, but it depending on the context it may not be readily evident which Nike they are referring to. Therefore the two terms cannot be clearly differentiated in some cases leading to inaccurate results. In addition, the usage of emoticons for expressing mood has also triggered a number of researches in analysing sentiments using emotive symbols.

The analysis on Twitter opinions would be beneficial to a wide range of real world applications; from the placement of advertisements based on trends, to applications which collect feelings towards a particular subject. Users can identify the type of feedback for a particular product or service in order to make informed decision on their purchase.Survey applications and organisations would especially benefit from this research, since the opinions expressed on twitter reflect the perceptions of a very large set of mostly IT literate population. In addition, this approach could be made to fit areas of locality when tweets are collected for a certain geographical area, such as a country.

The rest of the paper is organized as follows: In section II, a selection of related work done previously in this area is discussed. Section III outlines the techniques that have been

978-1-4673-5530-8/12/$31.00 ©2012 IEEE

taken into account inpreprocessing the data; section IV discusses the approaches used for classification, and section V outlines the results obtained during the course of this research. Finally, section VI specifies possible future directions for this study.

II. Related Work

Microblogging is a rapidly growing new trend on the Internet. Traditionally blogs have been long posts which take many minutes to write. Microblogging enables users to post short musings or information within moments. Thus, microblogging platforms such as Twitter are a great way to discover what people think do and communicate. Thus, discovering Twitter trends and opinions has become a very popular field for research. A lot of research has been done in this field by researchers and scholars all around the world.

Sentiment analyses in Tweets are typically done in two phases: (a) identifying sentiment expressions and (b) determining the polarity of the sentiment expressed in tweets. There have been different approaches used by researchers to classify tweets and analyse sentiments and trends in Twitter. Most researchers use lexical resources and decide the sentimentality of Tweets by the presence of lexical items [7] [8]. Some other researchers combine additional features such as conjunction rules with lexical analysis to obtain more results with better accuracy [9].

One of the early researches in this field was done by Go et al. [10] where the authors try a novel approach to automatically classify sentiments in tweets. They have used distant learning methods where they classify tweets ending with positive emoticons such as :-) as positive; and tweets ending with negative emoticons such as :-(as negative. For feature space they try a unigram and bigram model. Their results indicate that the unigram model outperforms the bigram model. One drawback of this approach is that the data collected for this testing is through search queries in Twitter, which may be biased. The research also uses POS (Parts of speech) tags as features. The results of [10] have shown that using POS tags is not useful in classifying Tweets.

Barbosa et al [11] present another approach in sentiment classification in Twitter data where they label 1000 Tweets using polarity predictions from three different websites and another 1000 Tweets for testing. They explore some characteristics of tweets such as how they are written, as well as meta-information of the words used to compose them. In addition to polarity of words and POS of words they use syntax features of tweets such as re-tweets, hashtags, punctuations and exclamation marks. The accuracy of the test results obtained was very high due to (a) creating a more abstract representation of these messages instead of using raw word representation; and (b) labels of reasonable quality provided by data sources being combined.

Another popular research topic is to identify sentiments of popular brands such as Apple, Armani etc. This has a commercial value for the companies in question as they can identify the drawbacks of their products and improvements could be carried out as needed in order to be competitive. Jansen et al. [12] use a commercial sentimental analyser to analyse the sentiments of major brands. There have also been approaches to match opinion keywords [5]. For example, the hashtag #sarcasm can be used to identify sarcastic Tweets about a brand or an event.

III. Data

The proposed approach required the use of two disjoint datasets. Data collection from Twitter involved more effort than expected and required manual labelling of posts for sentiments in relation to a query. A publicly available training dataset for Positive-Negative classification has been used in this research. However, the collected data is imbalanced with approximately 8500 manually annotated negative tweets and 41,000 positive tweets, hence some sampling techniques has been applied to reduce the skewness of the data.

As there are only a few publicly available datasets for tweets with sufficient data needed for sentiment analysis, the Neutral-Polar-Irrelevant training dataset (with approximately 5300 tweets) was manually collected by querying the Twitter Application Programming Interface (API). These queries were arbitrarily chosen from different domains to ensure variety of data. No restrictions such as language, location was made during the collection process. Thus, the collection consists of tweets in foreign languages as well. Each tweet has been labelled *polar*, *neutral* or*irrelevant*by an adult male fluent in English. The "irrelevant" labelled tweets are mainly Non-English tweets. The imbalanced nature of the collected dataset has been rectified as explained in a later section.

The users' privacy issues were handled as follows. Most 'tweets' on twitter are set public, and can be viewed by any person regardless of membership to twitter. The tweets which require 'following' the author in order to be accessible are termed private, and they are not be reflected on the public data stream (called the *public timeline*). The corpus that was used in this research was condensed from the public timeline, and hence does not reflect tweets that have been made private. Further, only the content of the tweets are collected for this research and no information pointing to the users(represented by the handle) are kept so that no means were kept, which could link a user with his/her opinion.

IV. Data Preprocessing

Due to the varying and unpredictable nature of language used in tweets, it is likely that preprocessing techniques could be used to standardize certain tokens of tweets. It is highly likely that most tweets contain some form of grammatical or spelling mistakes, acronym, colloquialisms and slangs; incorporated into due to the 140 character limit imposed by Twitter on tweets.

The preprocessing process extracts the relevant content from the tweets while leaving out the irrelevant ones. The techniques applied in this paper are used commonly in information retrieval applications specifically in sentiment analysis in micro-blogging. The collected data is passed through a series of pre-processors that assist in the conversion of the message strings into the feature vector.

Some of the preprocessing steps that have been carried out are explained below. This is one of the more vital steps in the entire classification process as the quality of the features/attributes that are extracted from the training dataset using the said preprocessing technique directly affects the performance of the classifiers

A. Replacing Emoticons

In many micro-blogging posts, emoticons are used by the users as an easy way of expressing emotion in a concise manner. Thus, emoticons are an easy way to differentiate polar from non-polar messages and positive from negative messages. A range of about 30 emoticons, such as :) :(:D =] :] =) =[=(are replaced with either a SMILE or FROWN keyword.

B. *Uppercase Identification*

In micro- blogging space, it is common to express powerful emotions using all capital letters (e. g. REALLY). This practice is usually called e-shouting, and it could be considered a good indicator of the polarity of the message. By identifying a series of capitalized words and adding an ALL_CAPS keyword, this preprocessing step extracts this feature before removing casing.

C. *Lower Casing*

It is important to have the entire word in a consistent case when classifying texts in order to guarantee that all tokens map to the corresponding feature irrespective of casing. This is extremely important for this research work as it is very common to find irregular casing (such as "*TwITteRseNtIMeNtanAlYsiS*") in micro-blogs.

D. *URL Extraction*

Many tweets contain URLs in order to share more content than what can be given in the limited-character post. The content in the URL might provide supplementary knowledge regarding the emotion a user trying to express, however it would be far too expensive to crawl URLs for their content. In order to trim down the feature size during training, all URLs in the training tweets have been replaced with an equivalence class <URL>. This could considerably reduce feature size.

E. *Detection of Pointers (usernames and hashtags)*

In Twitter, posts can point to other users with the use of an□ @□ token in front of a username. And users tag tweets pertaining to a category in twitter, using #. Again, to avoid explosion of features, we abstract it to a constant symbol <USER> and <HASHTAG>. This replacement of usernames and hashtags reduce the feature size by a large margin.

F. *Identification of Punctuations*

In micro-blogging space, it is common to use excessive punctuation in order to stay away from proper grammar and to communicate emotion more easily. The punctuations can also give insight to the polarity of the message. For example, exclamation marks are used to express powerful emphasis which are usually polar messages [13]. In this step, irrelevant punctuations marks were removed by replacing<PUNCT> to avoid redundant feature in the training set.

G. *Removal of Stop Words*

In Information Retrieval, it is a common process to remove words that are extremely common (have a high IDF value) which do not add substantial value to the classification process. Common words like *a*, *an* and*the*are collectively called stop words. As the inclusion of these words in a tweet does not provide any useful information, they are removed.

H. *Removal of Query Term*

The required tweets are pulled from using a query term. Thus the query term itself should not be used to decide the sentiment of the post, hence every query term is replaced with a <QUERY> keyword. Although this makes it somewhat of a stop word, it can still be useful when not using a bag-of-words model and the location of the query in relation to other words becomes important [14].

I. *Compression of Words*

Twitter users tend to be very informal in their language and most of them elongate words to express strong emotions. For instance, the term *"happyyyyyyyyy"* coveys a higher degree expression than *"happy"*.During training and evaluation, for words containing more than 3 subsequent occurrence of the same repeated character/letter, we reduce it to an sequence of three characters. For example, we convert *"cooooooooool"* to *"coool"*. The sequence is not reduced to the more common two-character *"cool"* in order to differentiate between the regular usage and emphasized usage of the specific word [15].

J. *Removing Skewness in Dataset*

When the training dataset is imbalanced, building useful classification models can be an especially challenging endeavour. Class imbalance presents a problem in the usage of traditional classification algorithms as they attempt to build models with the goal of maximizing overall classification accuracy [16]. Many techniques have been proposed to alleviate the problems associated with class imbalance such as data sampling and boosting.

Data sampling, attempts to improve classification performance by artificially balancing the class distributions of training datasets. There are two variations of sampling: over sampling and under sampling. Over sampling creates a more balanced dataset by increasing the number of instances in the minority class; and under sampling on the other hand reduces the number of instances belonging to the majority class. In this research both under sampling (using Stratified Sampling, Random Sampling) and oversampling (using SMOTE) have been used. Synthetic Minority Oversampling Technique (SMOTE) tries to counter the imbalance in the skewed dataset by synthetically generating the samples for minority class [14].

V. EVALUATION

A. *Experiment Methodology*

In many micro-blogging posts, emoticons are used by the Sentiment analysis for microblogging services could be carried out in many ways. One way, in which two or more classifiers are chained one after the other have been used by many such researches due to their high yield and better accuracy of mined data. In this research, an attempt is made to identify the most suitable classifiers out of a large pool of popular classifying techniques which can effectively be used for such a purpose. The first stage of such a process typically deals with classifying incoming per-processed data into three disjoint categories: neutral, polar and irrelevant. Then, during the second stage, the data classified under polar is fed to a second classifier in order to be further separated into positive and negative. It could be inferred that due to the removal of irrelevant and neutral data at an earlier stage, the accuracy of positive/negative classification is carried out in a higher accuracy than conventional methods where the whole data is classified into positive/negative/non-polar pools.

This process could be depicted in a diagrammatic form as follows:

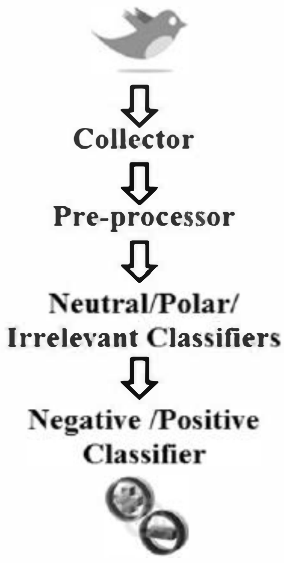

Fig.1 The classification process

B. *Building a Trained Data Model*

The sentiment analysis of the pre-processed tweet data has been carried out using several popular classifiers as discussed below, in order to identify the better performing classifiers and gauge their performance. The data mining and machine learning tool Weka, developed and maintained by the University of Waikato, New Zealand; has been used for this purpose as it has a number of classifiers in-built; with additional classifying mechanisms plugged in. A few of the classifying algorithms that are frequently used by several studies for this purpose have been described below:

a. Naïve Bayes

The Naïve Bayes classifier is a probabilistic model based on the Bayes' theorem, which calculates the probability of a tweet belonging to a specific class such as neutral, positive or negative. This assumes that all the features are conditionally independent [17]. Even though Naïve Bayes classifier has yielded better results in [18] it did not show superior results compared to some other classifiers as depicted in this section.

b. RandomForest

RandomForest consists of many classification trees known as tree classifiers, which are used to predict the class based on the categorical dependent variable [19]. Each tree gives a class for the input vector and the class with highest turns will be chosen. This classifier's error rate depends on the correlation between any two trees in the forest and the strength of the each individual tree in the forest. In order to minimize the error rate the trees should be strong and independent of each other [20].

c. Support Vector Machines (SVMs)

SVMs are the class of algorithms that are based on kernel substitution [21]. They can be defined as systems which use hypothesis space of linear functions in a high dimensional feature space. This will be trained with a learning algorithm that implements a learning bias derived from statistical learning theory. It is possible to construct highly nonlinear classification method using SVMs without getting stuck in local minima [21].

d. SMO

Sequential mining optimization algorithm efficiently solves the optimisation problem when training support vector machines. SMO takes an iterative approach to solve the optimization problem where it breaks it into a series of smallest possible sub-problems and solve them analytically. Each small problem involves two Lagrange multipliers (αi) because of the linear equality constraint. For any two multipliers the constraint can be reduced to the following equations $0 \leq \alpha 1, \alpha 2 \leq C$, $y1\alpha 1 + y2\alpha 2 = k$ which can be solved analytically. The algorithm finds a multiplier that violates KKT conditions and picks a second multiplier and optimizes the pair. It repeats this process again and again until convergence [22].

e. J48 (pruned or un-pruned C4.5 decision tree)

C4.5 is an algorithm developed by Ross Quinlan to generate decision trees from a set of training data. J48 is an open source java implementation of C4.5 algorithm which is used in WEKA data mining tool. The training data set is already classified where each sample is a vector which represents the attributes of the samples. C4.5 splits the sample at each node by choosing a suitable attribute of the data based on information gain [23].

In this experiment, the classifiers discussed above were used for different sized datasets, ranging from 5000 instances to 17000 instances. The training models were constructed by loading the pre-processed manually annotated data into Weka, and then using a percentage split method as well as a cross validations method to train and test the classifier. Each test was carried out on a 2.67GHz Intel Core i7™- based machine with 6GB of RAM, running the 64-bit version of the Microsoft Windows™ 7 operating system and 64-bit Oracle Java™ version 6. The performance statistics for each run and for each classifier were captured. The time taken for training and testing of each run depended upon the nature of the classifier, the sample size and as the percentage split between training data and testing data. The time taken typically varied from a few minutes to more than five hours. The metrics used for analysis are average figures of several runs for specific classifiers, for a dataset of size 17000 instances for neutral/polar/irrelevant classification and 6000 for positive/negative classification.

VI. Results and Discussion

A. *Neutral/polar/irrelevant classification*

The performance figures observed for each classifier are given in Table I as follows. The figures Fig. 2 and Fig. 3 present a graphical representation of the data shown in Table I. Y-axes depicting the percentage of accuracy of classification as found during the tests, while the X-axes stand for the number of the sample under test.

TABLE I
Performance of Algorithms for a Neutral/Polar/Irrelevant Classification

Classifiers	Avg. Accuracy	Max. Accuracy	Avg. F
Naïve Bayes (NB)	69.82%	71.33%	0.688
Naïve Bayes Multinomial (NBM)	78.90%	80.14%	0.783

Complement Naïve Bayes(CNB)	80.63%	83.33%	0.804
DMNBtext	80.92%	84.58%	0.809
Bayesian Logistic Regression(BLR)	74.64%	76.62%	0.746
SMO	81.86%	84.82%	0.820
SVM	77.80%	76.62%	0.776
J48	73.25%	77.60%	0.728
Random Forest(RF)	79.79%	82.72%	0.798
Lazy - IBK	70.88%	75.52%	0.708

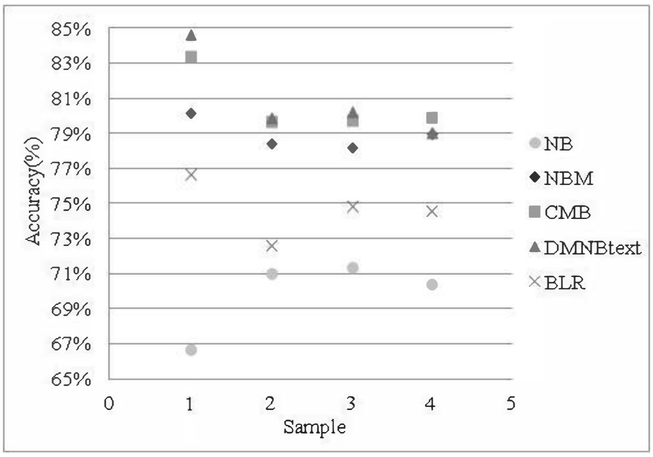

Fig.2 The variation of accuracy of Bayesian classifiers for different samples of fixed size, in neutral/polar/irrelevant classifications

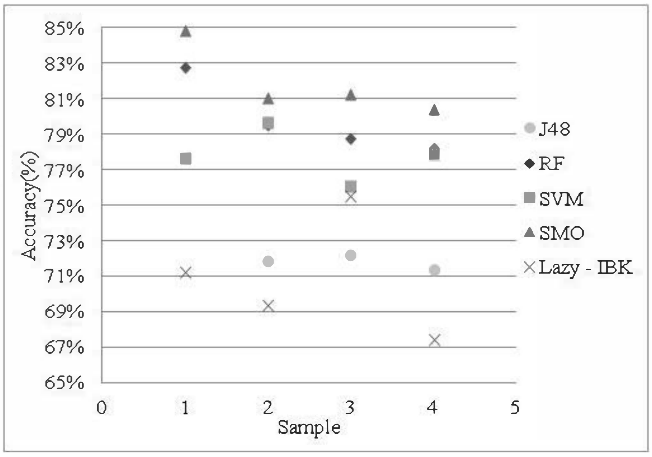

Fig.3The variation of accuracy of other classifiers for different samples of fixed size, for neutral/polar/irrelevant classifications

It is surprising to see some of the widely used algorithms fail to achieve satisfactory performance figures for this case. Most notably, the Naïve Bayesian and the Lazy - IBK classifiers achieve an average accuracy of about 70%. However, other variants of Bayesian classifiers, SMO and RandomForest gave accuracy figures of more than 80% in some cases. The SMO classifier gives the best overall average accuracy while the tree-based J48 classifying algorithm fails to perform consistently.

An issue that was encountered during training and classification of this dataset is that high skewness was present in the training set which may have affected the accuracy of the tested classifiers. Thus, the training dataset used for neutral/polar/irrelevant classification has been sampled using the SMOTE technique in order to reduce the skewness, and it is noted that the use of SMOTE to the preprocessed dataset has improved the accuracy by an extent. For example, the average accuracy of SMO has increased from 77.2% to 81.9% due to the application of SMOTE.

B. Positive/negative classification

Similar to the previous case, the following table shows the accuracy figures obtained during positive/negative classification of the data. Fig. 4 shows the variation of accuracy of Bayesian classifiers while Fig. 5 shows the variation of accuracy of non-Bayesian classifiers under study.

TABLE II
PERFORMANCE OF ALGORITHMS FOR A POSITIVE/NEGATIVECLASSIFICATION

Classifiers	**Avg. Accuracy**	**Max. Accuracy**	**Avg. F**
Naïve Bayes (NB)	64.66%	65.13%	0.646
Naïve Bayes Multinomial (NBM)	74.99%	76.42%	0.750
Complement Naïve Bayes(CNB)	74.96%	76.05%	0.749
DMNBtext	74.66%	75.96%	0.685
Bayesian Logistic Regression(BLR)	75.03%	76.62%	0.750
SMO	72.29%	74.90%	0.718
Random Forest(RF)	72.80%	72.80%	0.725
J48	65.44%	65.44%	0.667
Filtered Classifier(FC)	66.63%	67.85%	0.665
SVM	72.70%	72.70%	0.727

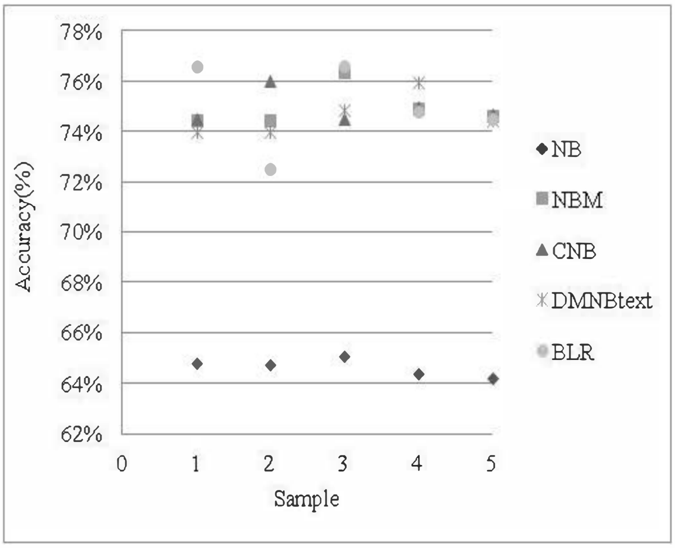

Fig.4The variation of accuracy of Bayesian classifiers for different samples of fixed size, in positive/negative classifications

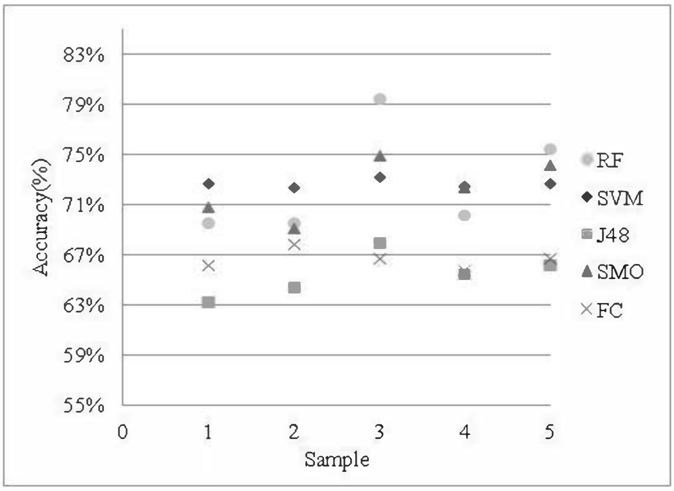

Fig.5The variation of accuracy of other classifiers for different samples of fixed size, in positive/negative classifications

In classifying data into positive/negative pools, again the Naïve Bayesian classifier fails to show sufficient accuracy. Other Bayesian variants perform much better, while the tree-based method RandomForest as well as SMO show better results for the third set. The variations of Bayesian classifiers gave more or less the same accuracy. Although the accuracy of many classifiers varies for eachsample, SVM gives consistent performance throughout the training set. Also, we can observe that FilteredClassifier offers marginally better performance compared to J48. However the resulting accuracy is less than that of neutral/polar/irrelevant classification.

Also, skewness of data problems present in the neutral/polar/irrelevant training set was less evident in this training set.

VII. Conclusions

In this study, we have analysed the problem of sentiment analysis and opinion classification of Twitter micro blog data, which, as discussed, is significantly different from other sentiment classification problem on structured and detailed messages.

We have analysed preprocessing of raw twitter messages (tweets) in detail and have provided guidelines for creating viable sets for training, based on research literature.

The reason for having multiple samplesis that if we arrive at a conclusion based solely on a single sample, the results could be misleading as they reflect characteristics that are specific to that sample only. The impact of sample-specific characteristics on the final conclusion (i. e. percentage accuracy of classification) were minimised by running the classification process on several samples. This is evident from the variations of accuracy of different sample as shown in figures Fig. 2, Fig. 3, Fig. 4 and Fig. 5, as each classifier shows a considerable variation between samples.

The skewness of some of the datasets was handled by introducing the SMOTE oversampling technique which helped improve the accuracy of some of the classifiers.

In the study of performance, Naïve Bayesian classifiers, most notably DMNBText provide consistently accurate results for both neutral/polar/irrelevant and positive/negative training sets, while the basic Naïve Bayesian classifier failed to perform at both cases. Out of other tested classifiers, SMO, SVM and RandomForest also displayed acceptable performance figures.

VIII. Future Work

We have tried several classification models and different classification algorithms for various sizes of datasets to address the topic. The biggest challenge for the neutral/polar/irrelevant classifier is the skewness in the dataset that impacts recall. In an attempt to counter this effect we tried sampling techniques such as oversampling as well asundersampling. Boosting techniques such as AdaptiveBoost etc. could be used in conjunction with classifiers to eliminate irrelevant entries in the training set, which may provide better recall rates, but may only be suitable for certain kinds of data.

Some of the more difficult challenges of natural language processing could also be used as further extensions of this study, such as sarcasm detection, comparison handling, context switches etc. Classification of international expressions and foreign words could also be looked into in more detail in the future.

Acknowledgment

The authors wish to thank Mr Nisansa de Silva, Department of Computer Science and Engineering for his assistance in carrying out this research.

We also wish to acknowledge the assistance of Mr Mark Hall(http://markahall.blogspot.com/) in securing the corpus for the training set and the testing process.

References

[1] B. Pang and L. Lee, —Opinion Mining and Sentiment Analysis. *In Foundations and Trends in Information Retrieval*, vol. 2, pp. 1–135, 2008.

[2] S. Quantification, —AI and Opinion Mining, IEEE Computer Society, pp. 72–79, 2010.

[3] B. Pang, L. Lee, and S. Vaithyanathan, —Thumps up? Sentiment Classification Using Machine Learning Techniques, in *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002.

[4] G. Mishne, —Experiments with mood classification in blog posts, in *Proc. 1st Workshop on Stylistic Analysis of Text For Information Access*, 2005.

[5] D. Davidov, O. Tsur, and A. Rappoport, —Enhanced sentiment learning using twitter hashtags and smileys, in *Proc. 23rd International Conference on Computational Linguistics: Posters*, 2010, pp. 241–249.

[6] R. Parikh and M. Movassate, —Sentiment Analysis of User-Generated Twitter Updates using Various Classification Techniques. 2009.

[7] J. M. Wiebe, —Learning subjective adjectives from corpora, in *Proc. National Conference on Artificial Intelligence*, 2000, pp. 735–741.

[8] E. Riloff and J. Wiebe, —Learning extraction patterns for subjective expressions, in *Proc. Conference on Empirical methods in Natural Language Processing*, 2003, pp. 105–112.

[9] H. Kanayama and T. Nasukawa, —Fully automatic lexicon expansion for domain-oriented sentiment analysis, in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 2006, pp. 355–363.

[10] A. Go, R. Bhayani, and L. Huang, —Twitter Sentiment Classification using Distant Supervision, Stanford, Technical, 2009.

[11] L. Barbosa and J. Feng, —Robust sentiment detection on twitter from biased and noisy data, in *Proc. 23rd International Conference on Computational Linguistics: Posters*, 2010, pp. 36–44.

[12] B. J. Jansen, M. Zhang, K. Sobel, and A. Chowdury, —Twitter power: Tweets as electronic word of mouth, *Journal of the American Society for Information Science and Technology*, vol. 60, no. 11, pp. 2169–2188, Nov. 2009.

[13] V. Pandey and C. V. K. Iyer, "Sentiment analysis of Microblogs," unpublished.

[14] S. Prasad, —Micro-blogging Sentiment Analysis Using Bayesian Classification Methods, Technical Report, 2010.

[15] A. Agarwal, B. Xie, I. Vovsha, O. Rambow and R. Passonneau, ―Sentiment Analysis of Twitter Data, *ACL HLT 2011*, p. 30, 2011.

[16] C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano, ―Building useful models from imbalanced data with sampling and boosting, in *Proc. 21st Annual FLAIRS Conference,* 2008, pp. 306–311.

[17] K.P.Murphy. (2006). Naive Bayes classifiers. [Online]. Available:http://www.cs.ubc.ca/~murphyk/Teaching/CS340-Fall06/reading/NB.pdf.

[18] A. Pak and P.Paroubek, "Twitter as a Corpus for Sentiment Analysis and Opinion Mining", in *Proc. 7thconference on International Language Resources and Evaluation LREC'10* ,May 2010.

[19] L.Breiman."Random Forests.",*Machine Learning*, vol. 45, pp.5-32, Jan. 2001.

[20] L.Breiman and A. Cutler. "RandomForests." Internet: www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm.[Apr.1, 2012].

[21] K.P.Bennet and C.Campbell. "Support Vector Machines: Hype or Hallelujah?" in *Proc.SIGKDD Explorations, 2000*, vol. 2, no. 2, pp 1-13.

[22] John. C. Platt. "Sequential Minimal Optmization: A fast algorithm for training support vector machines", Microsoft research technical report, Apr 21, 1998.

[23] J.Ross Quinlan. Programs for machines learning. Edward Brothers, 1993, pp.17-32