IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

Improved Twitter Sentiment Analysis Using N Gram Feature Selection and Combinations

Payal B. Awachate[1], Prof. Vivek P. Kshirsagar[2]

Computer Science and Engineering Department, Government College of Engineering, Aurangabad, India[1, 2]

Abstract: Sentiment analysis is an important research area that identifies the people's sentiment underlying a text. Sentiment analysis widely studied in data mining. Sentiment analyses of tweets are widely studied. After reviewing and studying the current research on sentiment analysis, the goal of the proposed method is to get the more effective results of sentiment analysis on tweets. The aim of this paper is to improve the performance to classify the tweets with sentiment information. We use a feature combination scheme which uses the sentiment lexicons and extracted tweets n gram of high performance gain. We evaluate the performance of three popular machine learning classifiers among which Kern lab classifier achieves the highest accuracy rate.

Keywords: Data Mining; Sentiment Analysis; Twitter; Classification; Supervised learning Ngram; Feature Selection; Sentiment Lexicon.

I. INTRODUCTION

Now days, almost every business want to know what customers think about their products. Web 2.0 makes it easier. The rise of the Web 2.0 and its user generated information led to many changes of the internet and its usage. The user created information on Web 2.0 can contain a variety of important market research information and opinions, through which economic opportunities and risk can be identified at an early stage. With rapid increase of Web 2.0 applications, reviews, comments, recommendations, ratings, feedbacks are generated by the user. For a new product launch, it can give them instant feedback about the reception of the new product.

Using the data mining concept, opinions and sentiments of any product can be analysed. So data mining on content derived from Web 2.0 applications has been one of the most challenging research topics from the beginning. Social networking sites draw the attention of data analytics researcher. Sentiment analysis is a method for organizations to increase their users/customer awareness and to improve their relationship using such sites. Twitter is one of most popular social media platforms having more than 100 million tweets per day. Twitter offers an easy way to access great amount of real and user generated data through a rest API. In most of the social networking sites, comments are posted under a specific hash tag to state the relation of the post with a specific topic. Tweets are short in length (140 characters limit).Because of limited size, it can be easy to identify the sentiments. Limited size of tweets forces the author to use the slang words and non-standard expression (e.g. "gr8" instead of using great).Most of the machine learning algorithms are not tailored for processing short content. Most of the machines learning algorithms are taking time for analysing and classifying tweets.

This work uses a supervised machine learning model. The aim of this work is to improve the level of performance. The level of accuracy improves with low execution time. We implement the combinations of features which use the sentiment lexicon dictionary and extracted n gram tweets of high information gain. By using these features, we evaluate the performance of machine learning algorithms. Our work results show that Kernlab classifier achieves highest accuracy and takes less time in classification of large amount of tweets.

II. LITERATURE SURVEY

Twitter Sentiment analysis is most specialized problem in Sentiment Analysis. Two typical approaches to sentiment analysis are lexicon based and machine learning. Sentiment lexicons do not capture domain and context sensitivities of sentiment expressions. Because of these limitations, machine learning approaches such as Naive Bayes algorithm do not reply on lexicons to classify the content. It automatically learns the context of the text from the training data.

So feature selection is an important phase for training the algorithms. Based on the features, algorithms need to be trained. Feature selection seeks to select an optimal subset of features by eliminating features irrelevant or offer no additional information compared to features within the optimal subset. Forman (Forman 2003) expressed many available feature selection techniques can be used to reduce irrelevant feature while improving classifier performance for a wide range of text classification problems. Pang and Lee Pang and Lee 2004) successfully utilize the sentiment information such as "thumbs up" or "thumbs down" to accurately classify documents. Guyon and Elisseeff (Guyon & Elisseeff 2003) demonstrated that performance increases from feature selection are in part due to reduction of over fitting. Kouloumpis et al. (Kouloumpis, Wilson, & Moore 2011) developed using word polarity based on prior probabilities as additional features. Saif et al. (Saif, He, & Alani 2012) examined

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

sentiment-topic features and semantic features to be used in conjunction with unigrams to achieve higher accuracy than unigrams. Sentiment classification has been used to address real world problems such as election prediction (Wang et al. 2012), and product sales (Liu et al. 2007).

Emotions are also used in sentiment classification. The tweets with ☹ emotions are treated as negative sentiment and the tweets with emotions are treated as positive sentiments. The algorithms of these are implemented by R. Bhayani, and L. Huang (R. Bhayani, and L. Huang 2009). Go, R. Bhayani, and L. Huang (Go, R. Bhayani, & L. Huang) examined twitter API to classify tweets and to integrate sentiment analysis classifier functionality into web applications.

Chamlertwatet al. (Chamlertwat et al. 2012) reported optimal performance for classification of tweets as subjective or objective was achieved by combining SVM with IG, they did not report the number of feature selected, or what other classifiers were tested. Narayanan et al. (Narayanan, Arora, & Bhatia 2013) conducted an experiment showing the benefit of applying feature selection in the related domain of movie review sentiment classification, but only tested single ranker, mutual information, using Na¨ıve Bayes.

Kouloumpis et al. (E. Kouloumpis, T. Wilson, and J. Moore, 2011) examine Twitter sentiment classification. With N-gram features, they include a sentiment lexicon, part of speech features as well as features that capture information about the informal and creative language used in micro blogging such as emoticons, abbreviations and the presence of intensifiers. Their findings show that Part of Speech features actually decrease the performance. Moreover, they claim that features from an existing sentiment lexicon were somewhat useful in conjunction with micro blogging features.

In this work, we use Sanders data set which is hand classified. We use prominent feature selection technique with N grams. We examine the classification algorithms performance by providing the different combinations of feature selection and sentiment lexicons.

III.METHEDOLOGY

A. Supervised Learning

Supervised learning process is used in our work. Here we first trained the classifier by selected features and labelled tweets. Then by using the trained classifier to predict the new tweets. The extracted feature is then combined with the labels to get the training set. The training set is represented as {(feature term1, label1),(feature term2, label2),(feature term3,label3)}.The whole training set is then used to build the classifer.Then use the train classifier to classify the testing data.

B. Classifiers

1. NaïveBayesClassifier

Naive Bayes classifier is based on Bayes therom. Naive Bayes classifiers are highly scalable. This classifier often does surprisingly well and is widely used. It often outperforms more sophisticated classification methods.

2. Decision Tree Classifier

Decision Tree algorithm is the most popular algorithm. Many real world problems can be solved by this algorithm. This algorithm is fast, accurate and more reliable algorithm

3. Kernlab Classifier

Kernlab classifier is a R package providing kernel based machine learning functionality. Kernel function used in training and classification .Kernlab includes Support Vector Machine. In the current work, this algorithm gives highest result as compared to other classifiers.

C. N gram features

N gram is a set of co-occurring words in a text. N gram is used for developing features for supervised machine learning model such as decision tree; naïve bayes.Ngram tokenization is generally used before removing the stop words. Go, R. Bhayani, and L. Huang (2009), extracted unigrams, bigrams and combination of unigrams and bigrams is this three different feature vectors applied to different classifiers. As bigram contains pair of word and first word can be a stop word and stop word normally contain more information. Like unigram “Good” is positive word. But bigram “Not good” is caring negative meaning. In most of the research word, unigram and bigram are highly recommended for tweeter sentiment analysis.

D. Sentiment Lexicons

In unsupervised method for sentiment analysis, sentiment lexicons are used as a feature and decided the polarity. In our work we tried to combine the concept of supervised and unsupervised methods. For sentiment lexicons we used the positive and negative lexicons by Bing Liu. In the current work you are combining the Ngram of high information gain feature extracted from the tweets and the sentiment lexicons to train the classifier and evaluated the performance of predicted result.

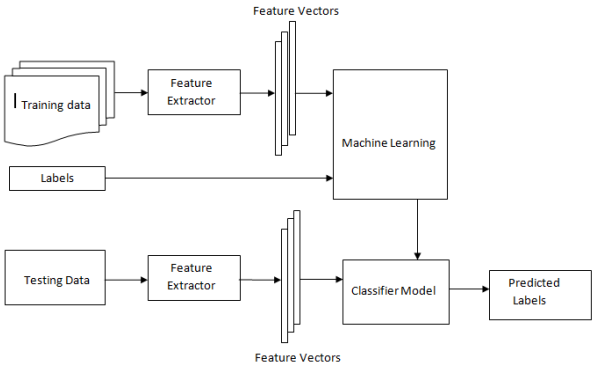

Fig. 1The Supervised learning process used in our work

IV.EXPERIMENT

A. Data Description

In this work; we used Sanders dataset of product reviews. There are around 2000 tweets out of which we considered 958 tweets which are having labels positive and negative.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

Out of which 632 tweets are negative and 326 tweets are positive tweets. We divided the data as train data and test data. So 50% of tweets are used for training the classifier and remaining 50% are used for testing purpose. In both train and test data, 316 are negative tweets and 163 are positive tweets.

B. Pre processing

Data pre-processing is important process to reduce the number of features which are unwanted and to make the features in manageable size. All unnecessary white spaces are removed. Removed punctuations, numbers. Stemming is performed on each word of the tweets. In Ngram feature selection.

Stop words are not removed. But in my current data set, by removing stop words, it gives higher accuracy as compared to accuracy obtained without removing stop words. The corpus is then converted into the lowercase to have consistent messages.

C. Feature Selection

After pre-processing, we have a useful corpus now. From this corpus, using Ngramtokenizer, we tokenized each corpus into the Ngrams by setting the minimum and maximum gram. In our work we set unigram as minimum and trigram as maximum gram. Form these Ngram, we removed the sparse terms. We set the sparse value 0.98.So the terms that occurs at most in less than 0.02 corpuses is removed.

Then we created BOW (Bag of words) vector by finding the frequent terms observed in selected terms .This vector is then used to create the Ngramtokanizer of testing dataset. By using this BOW vectors, we created three categorized of features as follows.

1) Features selected using Top Chi-Square: Here we used chi-square to use information gain for each ngrams. Our algorithm to select the top ngrams of high weights works as follows:

- We calculate the chi square weights for the training data terms. Regenerate training data set after chi-square feature selection.
- We pre-processed the testing data. Created the document term matrix with NGramTokenizer by passing the frequent terms observed in the training data set. So it will consider only that test corpus in which these terms are available and created testing dataset.
- Regenerate the testing dataset by passing the chi-square feature selection.
- Then train our classifier by using the training dataset. And evaluate the result for testing dataset.

2) Features which appear both in the corpuses as well as in the sentiment word list: Here we use sentiment lexicons provided by Liu.In our work we identified the positive and negative sentiment words which are observed in our corpuses. Algorithm works as follows:

- We created a word dictionary which is combination of sentiment words observed in our corpuses and bag of frequent terms observed in training dataset.
- We pre-processed the testing data. Created the document term matrix with N Gram Tokenizer by passing the word dictionary created above.
- Regenerate the training and testing data by considering the sentiment word observed in our corpuses.
- Then train our classifier by using the training dataset. And evaluate the result for testing dataset.

3) Combined list of features which consist of features from first category and from second category. Algorithm works as follows:

- We created a word dictionary which is combination of sentiment words observed in our corpuses and bag of frequent terms observed in training dataset.
- We pre-processed the testing data. Created the document term matrix with N Gram Tokenizer by passing the word dictionary created above.
- Regenerate the training and testing data by considering the combination of sentiment word observed in our corpuses and the chi square features extracted from bag of frequent terms.
- Then train our classifier by using the training dataset. And evaluate the result for testing dataset.

V. EXPERIMENTAL RESULT

We evaluated the performance of these three categories of feature selection by using three different classifiers. We evaluate the performance with respect to the aspect of accuracy, false positive, recall rate and F-measure. For all three categories, the kernlab support vector machine classifier and decision tree classifier gives better performance.

Kernlab support vector machine gives the highest performance for all three feature selection categories. Category 3 gives the highest performance result.

As shown in the Tab1e 1. in first category, Kernlab support vector machine (KSVM) and Decision tree classifier gives the good result. Out of which support vector gives the result larger than 80%.In second category also, kernlab support vector machine (KSVM) and Decision tree classifier gives the good result.

The accuracy rate of SVM (79.74%) good compare to others. In third category also kernlab support vector machine (KSVM) and Decision tree classifier gives the good result. The accuracy of KSVM is highest for this category as well as compare to other category as well.

The accuracy of KSVM for third category is 86.22% which is 4.6% and 6.48% higher than its counterparts in first and second category. The Kern lab support vector machine gives the best accuracy in our work.

DOI 10.17148/IJARCCE.2016.5935

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 9, September 2016

Table 1. Experiment result of three categories of n gram.

Classifiers	Category 1	Category 2	Category 3
Decision Tree	74.73 % (A) , 34.96 % (Rec), 79.16 % (P) , 48.51% (FM)	76.40% (A) , 55.78% (Rec), 76.04% (P) , 56.37% (FM)	74.73 % (A) , 34.76 % (Rec), 79.16 % (P) , 48.51 % (FM)
Naïve Bayes	34.44 % (A) , 100 % (Rec), 34.17 % (P) , 50.93 % (FM)	37.99 % (A) , 98.15 % (Rec), 35.24 % (P) , 51.86 % (FM)	34.02 % (A) , 100 % (Rec), 34.02 % (P) , 50.77 % (FM)
Kernlab Support Vector machine	81.62 % (A) , 56.44 % (Rec), 84.40 % (P) , 67.64 % (FM)	79.74 % (A) , 54.60% (Rec), 79.46 % (P) , 64.72 % (FM)	86.22 % (A) , 65.64 % (Rec), 91.45 % (P) , 76.42 % (FM)

VI.CONCLUSION AND FUTURE WORK

In the context of this work an improved feature selection with N gram features for sentiment analysis is provided. We evaluated the performance issues in terms of classification accuracy. We experimented 3 different feature combination techniques. Out of which the combination of sentiment lexicons and ngrams of high performance gain gives highest result compare to others. We experimented Sanders product data set. We evaluate the performance of three Classifiers. Out of which Kernlab classifier gives best result.

Regarding future work, we plan to do more experiment on cross validations. We will try to evaluate the result for different domain dataset. We will try to evaluate it using more classifiers.

ACKNOWLEDGMENT

I am heartily thankful to our guide, **Prof. Vivek. P. Kshirsagar**, Head of the department, Government college of Engineering for their valuable guidance and support for understanding the topic and actual development of the research work. We are also like to thank the institute for providing the required facilities.

REFERENCES

[1] Forman, G. 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res.3:1289–1305.
[2] Guyon, I., and Elisseeff, A. 2003. An introduction to variable and feature selection. J. Mach. Learn. Res. 3:1157–1182.
[3] Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in Proceedings of the 42nd annual meeting on Association for Computational Linguistics. Association for Computational Linguistics, 2004, p. 271.
[4] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," CS224N Project Report, Stanford, vol. 1, p. 12, 2009.
[5] A. Go, R. Bhayani, and L. Huang. (2015) Sentiment140 api. [Online].Available: http://help.sentiment140.com/api
[6] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the omg!" Icwsm, vol. 11, pp.538–541, 2011.
[7] Asiaee T., A.; Tepper, M.; Banerjee, A.; and Sapiro, G.2012. If you are happy and you know it... tweet. In Proceedings of the 21stACM International Conference on Information and Knowledge Management, CIKM '12, 1602–1606.
[8] Chamlertwat, W.; Bhattarakosol, P.; Rungkasiri, T.; and Haruechaiyasak, C. 2012. Discovering consumer insight from twitter viasentiment analysis. J. UCS 18(8):973–992.
[9] Witten, I. H., and Frank, E. 2011. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, 3rdedition.
[10] Wang, H.; Khoshgoftaar, T.; and Van Hulse, J. 2010. A comparative study of threshold-based feature selection techniques. InGranular Computing (GrC), 2010 IEEE International Conference on, 499–504.
[11] Narayanan, V.; Arora, I.; and Bhatia, A. 2013. Fast and accurate sentiment classification using an enhanced naive bayes model. In Intelligent Data Engineering and Automated Learning–IDEAL 2013. Springer. 194–201.
[12] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau, "Sentiment analysis of twitter data," in Proceeding of the Workshop on Languages in Social Media. Association for Computational Linguistics, 2011, pp. 30–38.