International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 3 Issue 5, August 2019 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Lexicon Based Emotion Analysis on Twitter Data

Nang Noon Kham

Faculty of Information Science, University of Computer Studies, Lashio, Myanmar

How to cite this paper: Nang Noon Kham "Lexicon Based Emotion Analysis on Twitter Data" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-3 | Issue-5, August 2019, pp.1008-1012, **https://doi.org/10.31142/ijtsrd26566**

IJTSRD26566

ABSTRACT

This paper presents a system that extracts information from automatically annotated tweets using well known existing opinion lexicons and supervised machine learning approach. In this paper, the sentiment features are primarily extracted from novel high-coverage tweet-specific sentiment lexicons. These lexicons are automatically generated from tweets with sentiment-word hashtags and from tweets with emoticons. The sentence-level or tweet level classification is done based on these word-level sentiment features by using Sequential Minimal Optimization (SMO) classifier. SemEval-2013 Twitter sentiment dataset is applied in this work. The ablation experiments show that this system gains in F-Score of up to 6.8 absolute percentage points.

KEYWORDS: *Sequential Minimal Optimization; Twitter*

1. INTRODUCTION

Social media platforms, particularly micro blogging services such as Twitter, are increasingly being explored by people to access and publish information about a great variety of trends every day. The language used in Twitter provides substantial challenges for sentiment analysis. The words used in this platform contain many abbreviations, acronyms and misspelled words that are not observed in traditional media. Over the past decade, there has been substantial growth in the use of micro blogging services such as Twitter and access to mobile phones worldwide.

Thus, there is tremendous interest in sentiment analysis of short informal texts, such as tweets and SMS messages, across a variety of domains such as commerce, health, military intelligence, and disaster management. These short unstructured textual messages from Social Media bring in new challenges to sentiment analysis. They are limited in length, usually spanning one sentence or less. They tend to have many misspellings, slang terms, and shortened forms of words. They also have special markers such as hashtags, user mention that is used to facilitate search but can also indicate a topic or sentiment. This paper describes a sentiment analysis system addressing the classification of tweets into three categories such as positive, negative and neutral. The system is based on a supervised text classification technique leveraging a variety of lexicon-based sentiment features. Given only limited amounts of training data, sentiment analysis systems often benefit from the use of manually or automatically created sentiment lexicons. Sentiment lexicons are lists of words (and phrases) with prior associations to positive and negative sentiments. Some lexicons can additionally provide a sentiment score for a term to indicate its strength of evaluative intensity. Higher scores indicate greater intensity. For instance, an entry great (positive, 1.2) in lexicon states that the word great has positive polarity with the sentiment score of 1.2. An entry acceptable (positive, 0.1) specifies that the word acceptable has a positive polarity and its intensity is 0.1 that is lower than that of the word great. This sentiment analysis system applies four freely available, manually created, general-purpose sentiment lexicons. These are one for words in negated contexts (Negated Context Lexicon), one for words in affirmative (non-negated) contexts (Affirmative Context Lexicon), one for emotion words in NRC Emotion lexicon and one for subjective words in Multi-Perspective Question and Answering (MPQA) lexicon.

The paper is organized as follows. A brief description of related work is presented in Section 2. Next, the description of the methodology used in this paper. Section 4 presents the architecture of the proposed system and the detailed description of it, including the experimental setting of classifier models and the feature sets, and dataset used in this system are explained in Section 5. It also provides the results of the evaluation experiments of this system. Finally, the conclusion and future research directions described in Section 6.

2. Related Work

Over the last years, there has been an explosion of work retrieving various aspects of sentiment analysis: detecting positive and negative opinion of sentences; classifying sentences as positive, negative, or neutral detecting the person expressing the sentiment and the target of the sentiment; detecting emotions such as joy, fear, and anger; visualizing sentiment in text; and applying sentiment analysis in health, commerce, and disaster management. Pang and Lee (2008) and Liu and Zhang (2012) gave a summary of many of these approaches. Sentiment analysis systems have been applied to many different kinds of texts including product reviews, newspaper headlines, novels, emails, blogs, and tweets [2] [9] [10][11]. Sentiment analysis of tweets was also presented by some researchers [4] [5].

Often these systems have to cater to the specific needs of the text such as structured versus unstructured, length of utterances, etc. Sentiment analysis systems were implemented specifically for tweets [1] [3]. Several manually created sentiment resources have been successfully applied in sentiment analysis. The MPQA Subjectivity Lexicon, which draws from the General Inquirer and other sources, has sentiment labels for about 8,000 words [7]. The NRC Emotion Lexicon has sentiment and emotion labels for about 14,000 words. These labels were compiled through Mechanical Turk annotations. To promote research in sentiment analysis of short unstructured texts and to establish a common ground for comparison of various approaches, an international competition was organized by the Conference on Semantic Evaluation Exercises (SemEval-2013) (Wilson et al., 2013). This organization developed and provided tweets for training, development, and testing. They also provided a second test set consisting of SMS messages. The purpose of having this out-of-domain test set was to assess the ability of the systems trained on tweets to generalize to other types of short unstructured texts. Some research approaches sentiment analysis as a two layers classification. At first, a piece of text is classified as either objective or subjective, and then only the subjective text is assessed to determine whether it is positive, negative, or neutral [7]. Also, this paper focuses on sentiment analysis of tweets from Twitter and our model classifies a tweet as three labels such as positive, negative and neutral using SemEval-2013 dataset.

3. Methodology

This system is composed of four main parts. The first one is, preprocessing, the second part is feature extraction, the third one is feature selection and the final part is the classification using fast SVM as implemented in SMO (sequential minimal optimization. A comparative analysis is also presented using different features with different classifiers.

3.1. Preprocessing

In this study, we perform the pre-processing steps before the actual methods of sentiment analysis are applied. The typical pre-processing procedure includes the following steps:

- Tokenization: The incoming string is broken into tokens: comprising words and other elements, for example, URL links. The common separator for identifying individual words is white space; however other symbols can also be used. Tokenization of social-media data is more difficult than tokenization of the general text. This work also applied the ArkTweetNLP library which was developed by Carnegie Mellon University and was specially designed for working with twitter messages. Ark Tweet NLP recognizes specific to Twitter symbols, such as hashtags, at-mentions, retweets, emoticons, commonly used abbreviations, and treats them as separate tokens.
- Stemming: It is a procedure of replacing words with their stems, or roots. The dimensionality of the BOW will be reduced when different words, such as read, reader and reading are mapped into one-word read and are counted together. This work applies the Snowball stemmer for performing the stemming operation.
- Stop words removal: Stop words are words which carry a connecting function in the sentence, such as prepositions, articles, etc. There is no definite list of stop words, but some search machines, are using some of the most common, short function words, such as the, is, at, which, and on. These words are removed since they have a high frequency of occurrence in the text but do not affect the final sentiment of the sentence.
- Part-of-Speech Tagging (POS): The process of part-of-speech tagging allows to automatically tag each word of text in terms of which part of speech it belongs to noun, pronoun, adverb, adjective, verb, interjection, intensifier etc. The goal is to be able to extract patterns from analyzing frequency distributions of these part-of-speech tags and use it in the classification process as a feature.

3.2. Feature extraction

Feature extraction is concerned with transforming text messages into a simple numeric representation. In this step, texts from the preprocessing step are tokenized using ARK Tweet NLP [8]. Bigrams are collections of two neighboring words in a text and trigrams are collections of three neighboring words. In general the use of trigrams helped to produce better results than the use of unigrams and bigrams, however, while using trigrams in short text, the use of trigrams led to the decrease of classification performance. In this paper, hybrid unigram and bigram: Unigram and bigram are extracted for each word in the text without any stemming or stop-word removing, all terms with occurrence less than 3 and less than 3 characters except numerical characters are removed from the feature space. Lexicons can be used to compute the polarity of a message by aggregating the orientation values of the opinion words it contains. They have also proven to be useful when used to extract features in supervised classification schemes [8]. Opinion lexicons, which are lists of terms labeled by sentiment, are widely used resources to support automatic sentiment analysis of textual passages.

By using the unigram and bigram features set, we applied lexicons to extract the emotion and sentiment related features. Lexicon-based is also called a dictionary. It contains a dictionary of words with pre-calculated polarity or sentiment scores. These features can be used as an independent method since the quality of classification in the lexicon-based approach depends solely on the quality of the lexicon. Sometimes, these features are considered to be part of the Machine Learning Unsupervised approach. However, in this paper, lexicon-based features are used for combination with other features to applied Supervised Machine Learning. To extract lexical features, we applied two lexicons such as Sentiment140 and the Multi-Perspective Question Answering (MPQA) Opinion corpus which is publicly available and consists of 4,850 words, which were manually labeled as positive or negative and whether they have strong or weak subjectivity.

3.3. Feature Selection

Feature selection is to select a subset of relevant features for building effective prediction models. By removing irrelevant and redundant features, feature selection can improve the performance of prediction models by alleviating the effect of the curse of dimensionality, enhancing the generalization performance, speeding up the learning process, and improving the model interpretability. Feature selection has found applications in many domains, especially for the problems involved in high dimensional data. Especially in a text mining application, feature selection is the best choice to improve classification accuracy and save time to build the model. In this work, we applied feature selection using the gain ratio method. We chose the best 1000 features set and 2000 features set for classification.

3.4. Classification

In this step, we used supervised machine learning techniques. These techniques require a labeled raining dataset on which the classifier will be trained. Each example instance in the training dataset consists of an input object and a label or a class (also called supervised signal). The supervised algorithm analyses labeled data extracts features that model the differences between different classes and infers a function, which can be used for classifying new instances. In the simplified form, the text classification task can be described as follows:

If our training dataset of labeled data is T = f{(t1, l1) ; ….; (tn, ln)}, where each text ti belongs to a dataset T and the label li = li(di) is a predefined class within the group of classes L ={l1, l2, …,ln}, the goal is to build a learning model that will receive as an input the training set T and will generate a classifier that will accurately classify unlabeled tweets. For this purpose, we applied two supervised machine learning classifiers such as Naïve Bayes and Sequential Minimal Optimization (SMO) for tweets classification in our sentiment analysis. We used the WEKA package to perform classification and performance analysis of feature extraction methods and learning models. Given a set of features extracted from the dataset, the classifiers trained statistical models. These trained models are then employed in the classification of unknown tweets and, for each tweet, they assign the probability of belonging to a class: Positive, Negative, and Neutral.

4. The architecture of the proposed system

The architecture of the proposed system is described in Fig. 1. The system has first loaded the tweets datasets. It removes the hashtags, URLs, user mention and RT (retweet) symbols from the tweets. And then it also eliminates the stop words during the preprocessing step. After that, word unigram and bigram features are extracted during the feature generation process. By using the gain ratio based feature selection method, this system chose the most relevant features from the generated features set.

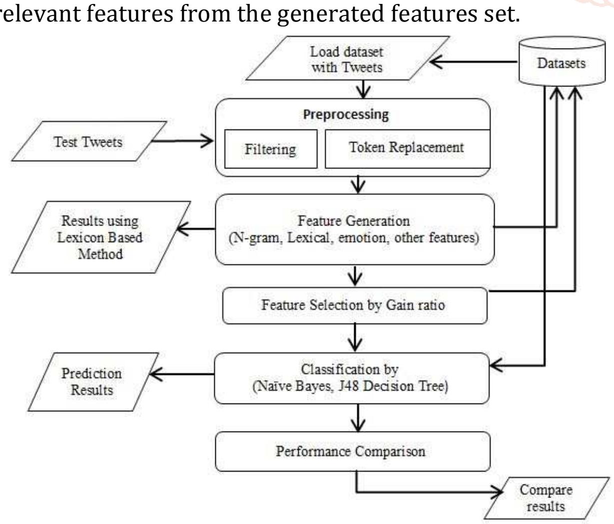

Fig.1. Proposed System Architecture

This system selected the top one thousand features for further classification. The selected feature vectors are applied to learn the two machine learning algorithms such as Naïve Bayes and J48 decision tree. The learned models are tested using 10 fold cross-validation method. The classification results are compared for the feature extraction methods as well as the classifiers.

5. Experimental Setting

This system performs a set of development experiments to evaluate the effectiveness of features extraction, learning models and lexicons on the performance of the proposed approach. A final test is done under the best development settings in order to evaluate the model with the best features set. This section presents experiments and results for the classification of two datasets based on two learning models. For each of the classification models, this system applies four different combinations of features set:

1. Unigram and lexicon features
2. Unigram, Bigram and lexicon features
3. Unigram, Bigram, Trigram and other features and
4. Unigram features only.
5. Bigram features only
6. Trigram features only
7. Unigram and Bigram features
8. Unigram, Bigram and Trigram features

The number of extracted features using unigram model is 6897, hybrid unigram and bigram are 24836 and hybrid unigram, bigram and trigram are 43481features. The number of lexical features extracted from two lexicons is 24 features. These are eight from sentiment140 unigram lexicon, eight from sentiment140 bigrams lexicon and eight from MPQA lexicon.

Before classification, the above-mentioned feature sets selected using Gain Ratio and we created the two different feature sets for each of the featured models. One contains 1000 features and the other contains 2000 features. The experiments are conducted using 14 features set of seven feature models by three well-known classifiers.

5.1. Dataset Description

This work uses the data provided for the SemEval-2013 competition (Wilson et al., 2013). In this dataset, tweets were collected through the public streaming Twitter API during a period of one year: from January 2012 to January 2013. There are total 8,258 tweets with 4,004 neutral tweets, 1,209 negative tweets and 3,045 positive tweets in this SemEval-2013 dataset. The tweets are comprised of regular English-language words as well as Twitter-specific terms, such as emoticons, URLs, and creative spellings. This system performed 10 fold cross-validation to test the efficiency of the features extraction and the model built during the training and testing phase. The results along with the experimentation of different datasets are described based on the accuracy of classifier models. The classification results of the different features sets with three classifier models are described in the following tables.

Table1. The classification Results of different Feature Models by using Naïve Bayes Classifier

Features Models	P	R	F	Acc	Time (seconds)
Unigram features(1000)	0.602	0.602	0.589	0.602	0.32
Unigram features(2000)	0.601	0.601	0.588	0.601	0.73
Bigram Only(1000)	0.465	0.465	0.465	0.465	0.24
Bigram Only(2000)	0.467	0.467	0.467	0.467	0.52
Trigram Only(1000)	0.449	0.449	0.449	0.449	0.4
Trigram Only(2000)	0.450	0.450	0.450	0.450	0.45
Unigram and Bigram features (1000)	0.600	0.599	0.586	0.599	
Unigram and Bigram features (2000)	0.600	0.600	0.587	0.600	0.57
Unigram, Bigram, Trigram Only(1000)	0.600	0.599	0.587	0.599	0.51
Unigram, Bigram, Trigram Only(2000)	0.599	0.599	0.586	0.599	0.56
Unigram and Lexicon (1000)	0.603	0.594	0.597	0.594	0.53
Unigram and Lexicon (2000)	0.605	0.597	0.599	0.597	0.98
Unigram, Bigram and lexicon Features (1000)	0.606	0.597	0.600	0.597	0.4
Unigram, Bigram and lexicon Features (2000)	0.606	0.597	0.600	0.597	0.82
Unigram, Bigram, Trigram and lexicon Features (1000)	0.607	0.598	**0.601**	0.598	0.39
Unigram, Bigram, Trigram and lexicon Features (1000)	0.607	0.598	**0.601**	0.598	0.39

Table2. The classification Results of Three Features Model by using J48 Classifier

Features Models	P	R	F	Access	Time (seconds)
Unigram features(1000)	0.573	0.574	0.546	0.574	5.22
Unigram features(2000)	0.565	0.562	0.531	0.562	10.93
Bigram Only(1000)	0.447	0.447	0.447	0.447	0.24
Bigram Only(2000)	0.447	0.447	0.447	0.447	0.51
Trigram Only(1000)	0.447	0.447	0.447	0.447	0.38
Trigram Only(2000)	0.447	0.447	0.447	0.447	0.46
Unigram and Bigram features (1000)	0.566	0.570	0.540	0.570	4.13
Unigram and Bigram features (2000)	0.564	0.569	0.539	0.569	4.40
Unigram, Bigram, Trigram Only(1000)	0.566	0.570	0.540	0.570	4.58
Unigram, Bigram, Trigram Only(2000)	0.566	0.569	0.539	0.569	7.53
Unigram and Lexicon (1000)	0.544	0.554	0.548	0.554	5.69
Unigram and Lexicon (2000)	0.531	0.538	0.533	0.533	14.26
Unigram, Bigram and lexicon Features (1000)	0.552	0.559	**0.554**	0.559	5.6
Unigram, Bigram and lexicon Features (2000)	0.549	0.554	0.551	0.554	11.64
Unigram, Bigram, Trigram and lexicon Features (1000)	0.550	0.559	0.553	0.553	5.14
Unigram, Bigram, Trigram and lexicon Features (2000)	0.551	0.559	0.553	0.553	10.78

Table3. The classification Results of Three Features Model by using SMO Classifier

Features Models	P	R	F	Acc	Time (seconds)
Unigram features(1000)	0.648	0.643	0.631	0.643	1.86
Unigram features(2000)	0.616	0.618	0.608	0.618	2.94
Bigram Only(1000)	0.635	0.512	0.420	0.512	0.16
Bigram Only(2000)	0.605	0.517	0.439	0.517	0.42
Trigram Only(1000)	0.719	0.465	0.315	0.465	0.38
Trigram Only(2000)	0.675	0.468	0.325	0.468	0.14
Unigram and Bigram features (1000)	0.695	0.669	0.655	0.669	2.07
Unigram and Bigram features (2000)	0.681	0.663	0.651	0.663	1.26
Unigram, Bigram, Trigram Only(1000)	0.693	0.665	0.650	0.665	1.47
Unigram, Bigram, Trigram Only(2000)	0.687	0.662	0.647	0.662	1.24
Unigram and Lexicon (1000)	0.663	0.662	0.657	0.662	2.01
Unigram and Lexicon (2000)	0.624	0.628	0.623	0.628	2.75
Unigram, Bigram and lexicon Features (1000)	0.687	0.682	0.677	0.682	1.77
Unigram, Bigram and lexicon Features (2000)	0.688	0.675	0.666	0.675	1.38
Unigram, Bigram, Trigram and lexicon Features (1000)	0.693	0.681	**0.673**	0.681	1.44
Unigram, Bigram, Trigram and lexicon Features (2000)	0.690	0.680	**0.673**	0.680	1.80

5.2. Evaluation of Feature Extraction and Classifier Models

In this section, the performance of the proposed system is evaluated on six different feature models with two different numbers of feature sets such as 1000 and 2000 feature sets with the best configuration obtained in different cross-validation tuning by SMO, Naïve Bayes and J48 classifiers. Table 1 presents the results of Naïve Bayes classifier using different feature models. According to the result, this classifier achieves up to 60% accuracy using unigram, bigram, trigram, and lexicon feature model. Table 2 presents the results of J48 classifier using different feature models. According to the result, this classifier achieves up to 55.4% accuracy using unigram, bigram and lexicon feature model. Table 3 presents the results of SMO classifier using different feature models. According to the result, this classifier achieves up to 67.3% accuracy using unigram, bigram, trigram, and lexicon feature model.

6. Conclusion

This work created a supervised statistical emotion analysis system that detects the sentiment of short unstructured textual messages such as tweets from Twitter. In this system, we implemented a variety of features based Among three classifiers, SMO classifier always outperforms the Naïve Bayes and J48 classifiers in every case. J48 is worst in all feature sets than SMO and Naïve Bayes. It takes a longer time than the others. In the future, we plan to adapt our sentiment analysis system to Myanmar languages other than English. Along the way, we continue to improve the Myanmar sentiment lexicons by generating them from larger amounts of data, and from different kinds of data, such as blogs, and Facebook posts in Myanmar. We are especially interested in algorithms that gracefully handle all kinds of sentiment modifiers including not only negations, but also intensifiers (e.g., very, hardly), and discourse connectives. on unigram, bigram and trigrams. We also included features derived from several sentiment lexicons: (1) sentiment140 unigram and bigram lexicons and (2) MPQA lexicon. Our experiments showed that SMO with unigram and bigram feature model using top 1000 features are superior in sentiment prediction on tweets in three classifiers. We are also interested in applying and evaluating the combination of unigram, bigram and trigram features with lexicons based features from tweets on data. According to the feature selection results, the lexicon-based features do not significantly affect the sentiment analysis in this work. We applied 24 lexicon-based features to combine the unigrams, hybrid unigrams and bigrams and the unigram, bigram and trigrams.

Feature selection is also used in these combined features and 1000 features set and 2000 features set are chosen for each feature models. The results of two selected feature sets of each model are not significantly different and sometimes, 1000 feature set of each model outperforms the 2000 feature set. Therefore, we chose 1000 feature set for our sentiment analysis. Among all feature models, hybrid unigram and bigram, and hybrid unigram, bigram and trigram model are outperforms the other models using different classifiers.

References

[1] A. Bakliwal, P. Arora, S. Madhappan, N. Kapre, M. Singh, and V. Varma, "Mining sentiments from tweets, In Proceedings of the 3rd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, WASSA '12, Jeju, Korea, 2012, pp. 11-18.

[2] A. C. Boucouvalas, "Real-time text-to-emotion engine for expressive internet communication", Emerging Communication: Studies on New Technologies and Practices in Communication, 5, 2002, pp. 305-318.

[3] A. Pak, and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining", In Proceedings of the 7th Conference on International Language Resources and Evaluation, LREC '10, Valletta, Malta, 2010,

[4] F. Aisopos, G. Papadakis, K. Tserpes, and T. Varvarigou, "Textual and contextual patterns for sentiment analysis over microblogs", In Proceedings of the 21st International Conference on World Wide Web Companion, WWW '12 Companion, New York, NY, USA, 2012, pp. 453-454.

[5] H. Saif, Y. He, and H. Alani, "Semantic sentiment analysis of twitter" in Proceedings of the 11th International Conference on The Semantic Web - Volume Part I, ISWC'12, (Berlin, Heidelberg), Springer-Verlag, 2012, pp. 508-524.

[6] J. Bellegarde, "Emotion analysis using latent effective folding and embedding", In Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text, Los Angeles, California, 2010.

[7] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language", Language resources and evaluation, 39 (2-3), 2005, pp.165-210.

[8] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith, "Part-of-speech tagging for Twitter: Annotation, features, and experiments," in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2, HLT '11, (Stroudsburg, PA, USA), Association for Computational Linguistics, 2011, pp. 42-47.

[9] R. Mihalcea, and H. Liu, "A corpus-based approach to finding happiness", In Proceedings of the AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs, AAAI Press, 2006, pp. 139-144.

[10] S. M. Mohammad, "#Emotional tweets", In Proceedings of the First Joint Conference on Lexical and Computational Semantics, *SEM '12, Montréal, Canada, 2012, pp. 246-255.

[11] V. Francisco, and P. Gervás, "Automated mark up of affective information in English texts", In P. Sojka, I. Kopecek, and Pala, K. (Eds.), Text, Speech and Dialogue, Vol. 4188 of Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2006, pp. 375-382.