IJCAT - International Journal of Computing and Technology, Volume 4, Issue 5, May 2017
ISSN : 2348 - 6090
www.IJCAT.org
Impact Factor: 0.835

A Novel Approach to Solve the Challenges in Sentiment Analysis

[1] Sunil B. Mane; [2]Ankita Chaudhari; [3] Ruchita Khairnar; [4] Shivani Charkha

[1,2,3,4] Department of Computer Engineering and Information Technology, College of Engineering, Pune COEP), Maharashtra, India

Abstract **- With the increased use of Internet nowadays for sharing opinions, views and sentiments about products, services, people and organizations; social networking and micro-blogging sites are gaining popularity. Twitter, one of the largest social networking sites is used by many people to share their opinions, life events and views about various topics. Here we are performing sentiment analysis on these tweets which are highly unstructured and heterogeneous using various machine learning algorithms like Multinomial Naive Bayes, SVM, Random Forrest, Decision Tree. There are various challenges involved in sentiment analysis of these tweets. This paper focuses on solving some of these challenges like sarcasm detection, negation handling, slang words standardization and emojis handling.**

Keywords - Sentiment Analysis, Twitter, Sarcasm, Machine Learning Algorithms, Data Preprocessing, Sentiment Analysis Challenges.

1. Introduction

Nowadays, people have started expressing their views, opinions, sentiments, emotions about various topics through micro-blogging sites like Twitter. Twitter users vary from common man to celebrities, politicians, company representatives and even country presidents thus providing a common platform for sharing their opinions about plethora of topics. Twitter messages known as tweets are limited to 140 characters length. Because of this short text length, more often they contain abbreviations, spelling mistakes, slangs, new words , an increased use of emojis, punctuations, hashtags which are a type of tagging for Twitter messages thus making it a difficult task for extracting opinions from these tweets. The Sentiment Analysis of Twitter data is an ongoing research. It has various applications in tracking customer reviews, knowing customers opinion about a certain product or service, analyzing the sentiment about the current trending topic, survey responses and forecasting market movement, etc. Opinion Mining of tweets involves number of challenges which act as has become an important and interesting field of research. This arouses a need to rapidly and accurately predict their sentiment. Sentiment analysis is the computational study of opinions, reviews, views, attitudes, emotions of people about varied products, services, organizations, topics, people,etc and classifying them into positive or negative category. Sentiment analysis is also referred as opinion mining. It is a classification process. The applications of Sentiment Analysis are endless in several fields. Sentiment analysis can be done at three levels [1]: Document level, Sentence Level, Aspect Level. Different techiques can be used to classify tweets. Most of these techniques can be classified under two categories [2]: Machine Learning and Lexicon Based. Lexicon based approach is divided further into two categories Dictionary based approach and Corpus based approach [3]. Supervised and Unsupervised are two main approaches of Machine Learning Technique [3].

This paper aims of assigning positive and negative sentiment class to tweets and it comes under the domain of "Natural Language Processing" and "Machine Learning" and it heavily uses the techniques that comes under these domains. This approach uses labeled datasets (positive, negative and sarcastic, non-sarcastic) and pre-processes them to get clean data. Further features are extracted using n-grams (unigrams and bigrams), tf-idf, sentiment score, capitalization and pos-tagging. Extracted features vectors are used by supervised machine learning classifier to build the model. Sentiment Classification is performed on test Data which is combination of positive, negative and sarcastic tweets and thus the performance is validated using evaluation metrics: Precision, F1-score, Recall, Accuracy and Confusion matrix.

2. Challenges in Sentiment Analysis

There are many challenges involved in classifying tweet into positive and negative sentiment class. Following are the prominent challenges [4] in sentiment analysis of tweets:

1. Sarcasm Detection: The sarcastic tweets expresses negative sentiment using positive words. Thus the classifier would incorrectly assign sentiment to these tweets. *"I love attending lectures on Weekend"*
2. Thwarted Expression: In some tweets only part of the tweet conveys the right sentiment. *"The play should be fantastic, the actors are brilliant and the story is great"*
3. Domain Dependence: Sometimes a single word can have different meanings, sense and polarity in different domains. *"She has long hairs", "Her hair bath takes long time".* In the first sentence, long has positive polarity and in the second sentence it has negative polarity.
4. Entity Detection: It is necessary to find out the entity toward which sentiment is directed and then separate the text to find sentiment of each entity. *"I hate you but I love her"*
5. Explicit Negation Handling: Negation handling can not only be done by using words like 'not', 'never' but also using less negative words like 'avoid', 'refrain'. *"Students try avoiding the mentor meetings in college"*

3. Literature Survey

A lot of research has been done in the field of "sentiment analysis" and it is an ongoing field of research. Bo Pang and Lillian Lee [5] provides a broad overview of different approaches and techniques in sentiment analysis. The paper [5] encourages to solve challenges in the field and openly makes resources available for the required work. Vishal A. Kharde and S.S.Sonawane in their paper gave a comparative analysis of the various machine learning approaches as well as lexicon based approaches for sentiment classification. They [4] also discussed briefly about the challenges in extracting the sentiments from unstructured and heterogeneous text. Challenges like Sarcasm Detection, Thwarted Expression, Order Dependence, Entity Recognition, Handling Comparisons, Internationalization, etc. are discussed in the paper. They [4] also propose that more clean the data is, more accurate the classification of text into appropriate classes take place.
Most of the papers remove emoticons while cleaning the data, Hao Wang and Jorge A. Castanon on the other hand proposed that emojis express extreme sentiment and they should not be removed while preprocessing and should be treated with caution [11].

Categorizing tweets into sarcastic and non-sarcastic tweets is a challenge in itself. They [7] propose the use of #sarcasm to gather sarcastic tweets from Twitter. They achieved an accuracy of 75 % by using this approach. [6] sites the importance of feature engineering in sarcasm detection. According to the paper, use of multiple features makes sarcasm detection more accurate. The set of features used in their work consisted of : n-grams, sentiments, parts of speeches, capitalizations, and topics. This features when used together gave better results than their individual performance. After doing literature survey on Sarcasm Detection, following three methods had the best results :

A Multidimensional Approach:
A multidimensional approach takes into consideration four different properties: Style, Unexpectedness, Signatures, Emotional Scenarios each containing several attributes related to sarcasm. They [8] have taken textual features of Twitter data to detect sarcasm.

Sarcasm as positive word and negative situation:
A sarcastic statement is assumed to be of the form: [Positive Word][Negative Situation] in the paper [10]. Thus the bootstrap algorithm relies on the assumption that more often the negative situation are preceded by positive seed words. For eg: "I enjoy being hated". Here the positive seed word is 'love' and the negative situation is 'being ignored' hence making the sentence sarcastic.

Semi-Supervised Recognition:
[9] classifies text in amazon reviews and tweets based on CW and HFW using syntactic and pattern based features. Each sentence generates multiple patterns allowing 1-6 slots for content word and 2-6 high frequency word.

4. Problem Definition

This paper mainly focuses on the challenge of Sarcasm Detection and takes into account Emojis Handling, Slang Word Standarization on twitter data and makes a comparative analysis of different machine learning algorithms. For Sarcasm Detection , combination of features like n-grams, capitalization, sentiment score, part of speech are used together to generate feature vcctors. Emojis Handling and Slang Word Standaridazation are handled during the preprocessing stage by using lookup tables.

5. Proposed Methodology

The sentiment analysis of tweets consists of the following stages: Data Collection, Data Preprocessing, Feature Extraction, Training Classifier, Sentiment Classification.
The proposed model trains two classifiers namely classifier1 and classifier2 wherein former is trained on positive and negative tweets and the latter is trained on sarcastic and non-

sarcastic tweets. These classifiers together build the model and then the test data in the form of tweets is classified into positive and negative sentiment classes.

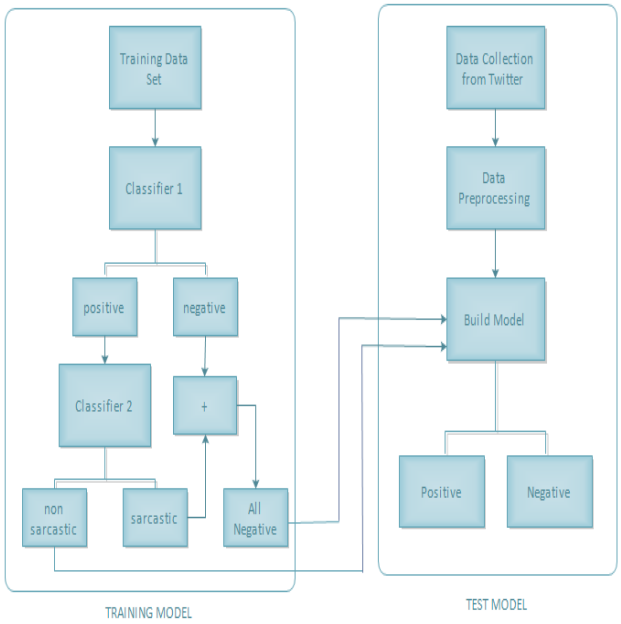

Fig. 1 Proposed System Architecture

5.1 Data Collection

Data is acquired in the form of raw tweets from twitter API which is publicly available. The tweets are collected using python library "tweepy" with search field as seed word tags like #joy, #happy, #:) #sad, #disappointed, #:(, etc. which are categorized into positive and negative classes and the language of tweets is English. And for the sarcastic tweets, #sarcasm and #sarcastic is used to fetch positive instances of sarcasm. Even the positive data set of tweets fetched with positive seed tags had some instances of sarcasm in it which were manually removed.

5.2 Data Preprocessing

The first step after the labeling of datasets into positive, negative and sarcastic is Data Preprocessing. These tweets are very informal and unstructured. They contain slangs, grammatical mistakes, abbreviations, emojis, emoticons, punctuations, links, usernames, hashtags. The Data Cleaning and Preprocessing step converts the noisy text into clean and appropriate form to extract proper sentiment. Following preprocessing tasks are performed step by step:

- Escaping HTML Parser
- Removal of URL's
- Removal of usernames starting with '@'
- Removal of punctuation characters
- Standardizing words
- Split Attached Words
- Non-removal of Hash Tags
- Handling Slangs and Acronyms
- Handling Emojis and Emoticons
- Removal of Stop Word

5.3 Feature Extraction

In order to build model for sentiment classification, labeled data has to be converted into feature vectors. Creative Features play an important role in the success or failure of model in accurately classifying text into sentiment classes. Thus for training the classifier on sarcastic tweets, combination of features like N-grams, POS Tagging, Capitalzation, Sentiment Score are used to generate the feature vector. The tweet is divided into two and three parts and the sentiment score for each part is calculated using SentiWord Net library and the contrast in the scores of the parts is inserted into the feature vector. The Python NLTK library is used for extracting N-grams and for POS Tagging.

5.4 Sentiment Classification

The paper uses Supervised Machine Learning approach wherein the pre-labeled raw data is preprocessed and then used for training. After the classification of test data into positive and negative sentiment classes, the positive data is then classified into sarcastic and non-sarcastic tweets. The sarcastic tweets are labeled as negative. Thus the final output is in the form of test data being classified into positive and negative tweets. The following classification algorithms are used to classify tweets into positive and negative sentiment classes:

1. Bernoullis Naive Bayes
2. Multinomial Naive Bayes
3. Linear SVM
4. Random Forest
5. Decision Tree
6. Maximum Entropy (Logical Regression)

The sklearn module of Python is used to for the implementation of the classification algorithms. The sarcastic and non-sarcastic tweets are trained using Bernoullis Naive Bayes, Random Forest and Maximum Entropy classifiers.

6. Results and Discussion

The paper performs sentiment classification on tweets that are obtained from Twitter. The size of training datasets is as follows:

Positive Tweets: 1000
Negative Tweets: 1000
Sarcastic Tweets: 2000

The raw data is cleaned using various preprocessing techniques. The techniques that have a good impact on improving the accuracy of sentiment classification are Handling emoticons, slang words standardization and non removal of hashtags. Given below is a graph depicting the accuracy results of sentiment classification on Y axis and dependent variables on X axis. The dependent variables are raw data and preprocessed data. The graph shows an increase in accuracy by 2% after preprocessing the raw data using the above techniques. The accuracy of sentiment classification of raw data is found to be 74.21% and that of preprocessed data is 76.35%.

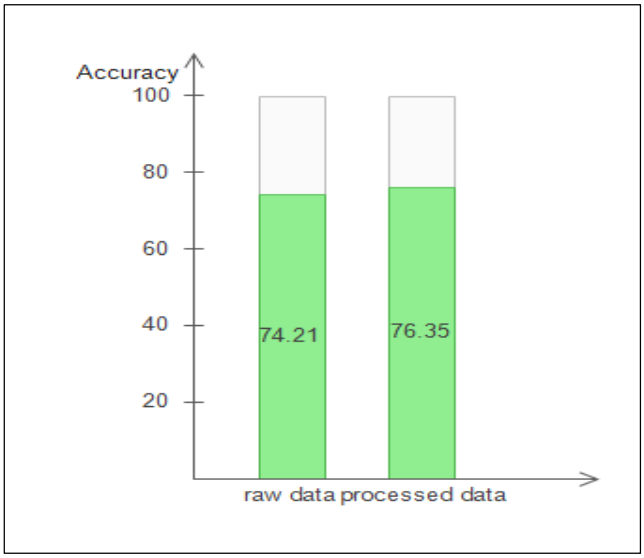

Fig. 2 Effect of Data Preprocessing

The graph given below shows the accuracy of sentiment classification before handling sarcasm and after handling sarcasm of preprocessed data. In the first case, the preprocessed data is classified into positive and negative sentiment classes without handling sarcasm. After that the challenge of Sarcasm Detection is handled and the preprocessed data is classified into positive and negative classes where the sarcastic tweets are labeled negative. There is an increase in accuracy by 8% on handling the challenge of Sarcasm Detection. The accuracy without handling Sarcastic tweets was 74.21% and after handling the challenge it reached to 82.92%.

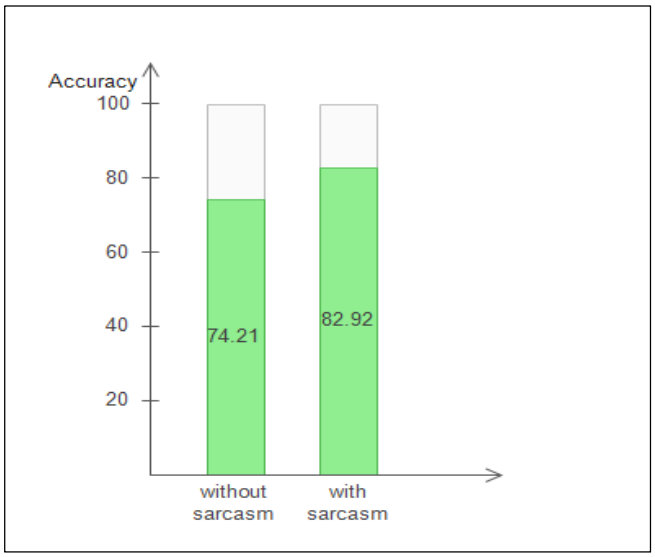

Fig. 3 Effect of Handling Sarcasm

7. Conclusion

In this paper, we have used Twitter data in English language to perform sentiment analysis and solve the challenges. We have built two models: one for classifying tweets into positive and negative sentiment classes and the other the model solves the challenge of Sarcasm Detection. On dealing with the emojis, emoticons, slang word standardization and non-removal of hash-tags, the resultant accuracy increased by 2%. After the handling of the challenge of sarcasm detection on preprocessed data, there was an increase in the accuracy result by 8%. Thus more clean the raw data is, more is the accuracy obtained. Sarcasm Detection plays a vital role in Sentiment Classification and increases its accuracy without affecting the performance.

References

[1] Waala Medhat, Ahmed Hassan and Hoda Korashy,"Sentiment analysis algorithms and applications: A survey", Ain Shams Engineering Journal, Vol. 5, Iss. 4, Year. 2014, pp. 1093-1113.

[2] Chetan Kaushik and Atul Mishra, "A scalable, lexicon based technique for sentiment analysis", International Journal in Foundations of Computer Science & Technology (IJFCST), Vol.4, No.5, Year. 2014

[3] Vimalkumar B. Vaghela, and Bhumika M. Jadav, "Analysis of Various Sentiment Classification Techniques", International Journal of Computer Applications, Vol. 140, Year. 2016, pp. 0975 – 8887

[4] Vishal A. Kharde and S.S.Sonawane, "Sentiment Analysis of Twitter Data: A Survey of Techniques", International Journal of Computer Applications, Vol. 139, pp. 0975 – 8887, Year. 2016.

[5] Bo Pang and Lillian Lee, "Opinion mining and sentiment analysis", Foundations and trends in information retrieval, Vol. 2, pp. 1 – 135, Year. 2008.

[6] Chun-Che Peng, Mohammad Lakis and Jan Wei Pane, "Detecting Sarcasm in Text: An Obvious Solution to a Trivial Problem", Foundations and trends in information retrieval, Year. 2015.

[7] Christine Liebrecht, Florian Kunneman and Antal van den Bosch, "The perfect solution for detecting sarcasm in tweets #not", Proceedings of the 4th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 29-37, Year. 2013

[8] Antonio Reyes, Paolo Rosso and Tony Veale, "A multidimensional approach for detecting irony in twitter", Vol. 47, pp. 239-268, Year, 2013.

[9] Dmitry Davidov and Oren Tsur and Ari Rappoport, "Semi-Supervised Recognition of Sarcastic Sentences in Twitter and Amazon", CoNLL '10 Proceedings of the Fourteenth Conference on Computational Natural Language Learning, pp. 107-116, Year, 2010.

[10] Ellen Riloff , Ashequl Qadir, Prafulla Surve , Lalindra De Silva, Nathan Gilbert and Ruihong Huang, "Sarcasm as Contrast between a Positive Sentiment and Negative Situation", EMNLP, pp. 704-714, Year. 2013.

[11] Hao Wang and Jorge A. Castanon, "Sentiment expression via emoticons on social media", BIG DATA '15 Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), Year. 2015.