SENTIMENT ANALYSIS OF FIGURATIVE SPEECH USING TWITTER DATA

VEDAANTA AGARWALLA
140001038

K SUDHARSAN
140001014

Department of Computer Science and Engineering
IIT INDORE

EMAIL : cse140001038@iiti.ac.in
cse140001014@iiti.ac.in

Sentiment Analysis of Figurative Speech Using Twitter Data

ABSTRACT :

Microblogging today has become a very popular communication tool among Internet users. A plethora of users share opinions on different aspects of their life every day. Twitter is the most popular microblogging platform. A wide range of features and methods for training sentiment classifiers for Twitter datasets have been researched in recent years with varying results. In this paper, we show how to automatically collect a corpus for sentiment analysis and opinion mining purposes. Using some training data, We build a sentiment classifier, which is able to classify tweets into positive and negative sets then test on the corpus. This sentiment classifier uses Support Vector Machines. The paper is based on the model made by DsUniPi team for their shared Sem eval task 2015.The method is supervised and makes use of structured knowledge resources, such as Senti-WordNet sentiment lexicon for assigning sentiment score to words and WordNet for finding word similarity.

1. INTRODUCTION

Microblogging websites have evolved to become a source of various kind of information. This is due to nature of microblogs on which people post real time messages about their opinions on a variety of topics, discuss current issues, complain, and express their sentiments for products they use in daily life. Twitter, with nearly 700 million users across the world and over 295 million messages per day, has quickly become a gold mine for organizations to monitor their reputation and brands by extracting and analyzing the sentiment of the tweets posted by the public about them, their markets, and competitors.

Sentiment analysis over Twitter data and other similar microblogs faces several new challenges due to the typical short length and irregular structure of such content. The main research direction in the literature of sentiment analysis on microblogs is focused on identifying new sets of features to add to the trained model for sentiment identification, such as microblogging features including hashtags, emoticons, the presence of intensifiers such as all-caps and character repetitions etc.

There are many ways in which social network data can be leveraged to give a better understanding of user opinion. Such problems are at the heart of natural language processing (NLP) and data mining research.

In this paper, we present a tool for sentiment analysis which is able to analyze Twitter data. We build a model which performs a 2-way task of classifying tweets into positive and negative classes. We use manually annotated Twitter data for our experiment. We also introduce two resources - a hand annotated

dictionary for emoticons that maps emoticons to their polarity and an acronym dictionary collected from the web with English translations of over 5000 frequently used acronyms.

2. LITERATURE SURVEY

Sentiment analysis has been handled as a Natural Language Processing task at many levels of granularity. Starting from being a document level classification task, it has been handled at the sentence level and more recently at the phrase level. Microblog data like Twitter, on which users post real time reactions to and opinions about "everything", poses newer and different challenges. Some of the early and recent results on sentiment analysis of Twitter data are by Go et al. (2009), (Bermingham and Smeaton, 2010) and Pak and Paroubek (2010). Go et al. (2009) use distant learning to acquire sentiment data. They use tweets ending in positive emoticons like as positive and negative emoticons like as negative. They build models using Naive Bayes, Maximum Entropy and Support Vector Machines (SVM), and they report SVM outperforms other classifiers. In terms of feature space, they try a unigram, a bigram model in conjunction with parts-of-speech (POS) features. They note that the unigram model outperforms all other models. Specifically, bigrams and POS features do not help. Pak and Paroubek (2010) collect data following a similar distant learning paradigm. They perform a different classification task though: subjective versus objective.
For subjective data they collect the tweets ending with emoticons in the same manner as Go et al. (2009). For data they scan twitter accounts of some of the most popular newspapers. They report that Part of Speech and bigrams both help (contrary to results presented by Go et al. (2009)). Both these approaches, however, are primarily based on n-gram models.
Another significant effort for sentiment classification on Twitter data is by Barbosa and Feng (2010). They use polarity predictions from three websites as noisy labels to train a model and use 1000 manually labeled tweets for tuning and another 1000 manually labeled tweets for testing. They however do not mention how they collect their test data. They propose the use of syntax features of tweets like retweets, hashtags, links, punctuations and exclamation marks in conjunction with features like prior polarity of words and POS of words.

3. DATA DESCRIPTION

Twitter is a social networking and microblogging service that allows users to post real time messages, called tweets. Tweets are short messages, restricted to 140 characters in length. Due to the nature of this microblogging service (quick and short messages), people use acronyms, make spelling mistakes, use emoticons and other characters that express special meanings, like all capital letters or repetition of letters to intensify the sentiment. Following is a brief terminology associated with tweets.

Emoticons: These are facial expressions pictorially represented using punctuation and letters; they express the user's mood.

Target: Users of Twitter use the "@" symbol to refer to other users on the microblog. Referring to other users in this manner automatically alerts them if a tweet is referred to them.

Hashtags: Users usually use hashtags to mark topics. This is primarily done to increase the visibility of their tweets.

Acronyms: Due to short character limit for a tweet users often use short terms to refer various things (e.g., *asap* for *as soon as possible).*

Retweet: Users generally use "RT" to share someone else's tweet with their followers.

4. PRE-PROCESSING OF DATA

For pre-processing twitter data, We use two new resources - an emoticon dictionary and an acronym dictionary. We prepare the emoticon dictionary by labeling 150 emoticons listed on Wikipedia with their emotional state. For example, ":-)" is labeled as positive whereas ":=(" is labeled as negative. We assign each emoticon a label from the following set of labels: Positive, Negative, and Neutral. We compile an acronym dictionary from an online resource.3 The dictionary has translations for 5,308 acronyms. For example, *bff* is translated to *best friends forever*. We pre-process all the tweets as follows:

- convert all the letters to lowercase
- replace all URLs with simple text, 'URL'
- replace targets (e.g. "@John") with simple text, 'AT_USER'
- remove additional white spaces
- replace hashtags with the hashtagged text
- replace all the emoticons with their sentiment polarity by looking up the emoticon dictionary
- replace all the acronyms with their full forms by looking up the acronym dictionary
- remove punctuation marks.

We replace a sequence of repeated characters by two characters (to avoid conversion of *goooood* to *god* instead of *good*), for e.g., convert *huuungry* and *hungryyyyyy* to *huungry* and *hungryy*, respectively.
Firstly we obtained manually annotated twitter data from online data resources. We choose this method as the method of downloading Tweets from Twitter API was very cumbersome and had a lot of noise.
The preferred method for parsing data was using the python library "Pandas" as it provides an easy and efficient functionality for data study.

This was followed by data cleaning. We firstly removed all the words which were preceded by "@" as they are usually referring to some username. We also use the techniques of stemming before retrieving SWN scores of the words.

Stemming is the term used in linguistic morphological and information retrieval to describe the process for reducing inflected (or sometimes derived) words to their word stem. base form—generally a written word form. Example stemming maps running to run.

This helps in getting accurate SWN score of the word as the chances that the root word is preset in the corpus are relatively higher.

STOP WORDS : These are the words which occur very frequently in texts but carry little or no sentimental value attached with them.

We also removed all the stop words from the tweet before calculating the SWN score.
We used a variety of features and their combinations for the SVM vector array.

5. CLASSIFICATION

After pre-processing a tweet, We use the remaining words to build a feature vector. The entire feature vector is a combination of the following features-:

Hashed words- The word following a hashtag are usually a very integral part of the tweet. They convey the sense and meaning of the tweet in a concise manner Thus we use the count of positive hashtags, negative hashtags as a feature in our SVM feature vector. The nature of the word is derived by extracting its SWN score.

Punctuation- The presence of question mark and exclamation mark in the tweet is also used as a feature. The feature takes binary values of 0/1 depending on whether the punctuator is present in the tweet.

Similarity- POS Tagging is used to classify text into noun,verb,adjective and adverb classes. POS-tagging is performed on the tweets with the use of nltk library and simplified tags (NN, VB, ADJ, RB). Words that belong to the same part of speech are used in semantic text similarity calculation *simt*. For this feature, we used Resnik's similarity measure. The value *simt* is calculated as the maximum similarity score of every combination of two words and their synonyms.

SWN score- Finally, the SentiWordNet score for each word in a tweet is calculated ignoring words that have been removed in the prepocessing. These include stop words and usernames. If the score of a word cannot be determined, then we calculate the SentiWordNet score of the stemmed word. The feature value was calculated as follows :-

The positive value of the word(POS) and the negative value(NEG) were retrieved. The feature vector was then assigned the value (1 + POS-NEG).

This score was computed for all n-grams upto three. Two implementations, one with a single feature for all n-grams and one with three different features were tested.

6. LINEAR SVM

Suppose some given data points each belonging to one of three classes, and the goal is to decide which class a new data point will be in. In the case of support vector machines, a data point is viewed as a p-dimensional vector (a list of p numbers), and all such points are separated with a (p-1)-dimensional hyperplane. This is called a linear classifier. There are many hyperplanes that might classify the data. One reasonable choice as the best hyperplane is the one that represents the largest separation or margin, between the classes. So, the hyperplane, so that the distance from it to the nearest data point on each side is maximized, is chosen. If such a hyperplane exists, it is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier or equivalently, the perceptron of optimal stability.
Given some training data **D**, a set of ***n*** points of the form:
$\mathbf{D} = \{ (\boldsymbol{x_i} , \boldsymbol{y_i}) \mid \boldsymbol{x_i} \in \mathbf{R}_P , \boldsymbol{y_i} \in \{ 0, 1, 2 \} \}$,
where $\boldsymbol{y_i}$ indicates the class to which the point $\boldsymbol{x_i}$ belongs and each $\boldsymbol{x_i}$ is a p-dimensional real vector, We want to find the maximum-margin hyperplane that divides the points having $\boldsymbol{y_i}$=0, $\boldsymbol{y_i}$ = 1 and $\boldsymbol{y_i}$ = 2. If the training data are linearly separable, two hyperplanes can be selected in a way that they separate the data and there are no points between them, and then their distance is maximized.

7. RESULT

We evaluate the performance of our approach measuring the accuracy for the test data set. The most useful features are pos-tags and SentiWordNet score. Hashtags also seem to contribute and the rest of the selected features contribute marginally. Semantic Similarity(Resnik) was implemented but did not contribute much to the accuracy and thus was removed from the final documentation of the code. These results are coherent with sentiment analysis literature where prior polarity along with POS-tagging seem to add most value to a classifier, and other features add up only marginally. We tried two different classification algorithms namely – Linear SVM and Naïve Baye's. Linear SVM gave better results.The code gave a final accuracy

of around **63.77%** in predicting the sentiment of a random given tweet.

8. CONCLUSION

Microblogging nowadays became one of the major types of the communication. A recent research has identified it as online word-of-mouth branding (Jansen et al., 2009). The large amount of information contained in microblogging web-sites makes them an attractive source of data for opinion mining and sentiment analysis. In our paper, We used an available corpus to train our sentiment classifier. My classifier model is able to determine positive and negative sentiments of test tweets. The classifier model is based on linear SVM which is trained with the feature vectors of each training tweet.
It is clear that SVM outperforms other classifiers. But, it would be interesting to see how a hybrid of other classifiers (like Naive Bayes classifier) with SVM would perform. Other features like considering each sentence separately can also be included to produce better results.

9. REFERENCES

[1] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau, "Sentiment Analysis of Twitter Data," in Proc. ACL 2011 Workshop on Languages in Social Media. pp. 30–38 (2011).

[2] Efthymios Kouloumpis, Theresa Wilson, and Johanna Moore, "Twitter Sentiment Analysis: The Good the Bad and the OMG!," in Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM-2011), 2011

[3] Hassan Saif, Yulan He, and Harith Alani, "Semantic sentiment analysis of twitter," in the 11th International Semantic Web Conference (ISWC 2012), 11-15 November 2012, Boston, MA, USA.

[4] Bing Liu, "Sentiment Analysis and Opinion Mining," Morgan and Claypool Publishers, May 2011.

[5] Alexander Pak and Patrick Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," in Proceedings of LREC.K. Elissa, "Title of paper if known," unpublished.

[6] Alec Go, Richa Bhayani, and Lei Huang, "Twitter sentiment classification using distant supervision," technical report, Stanford (2009).

[7] Albert Bifet, and Elibe Frank, "Sentiment knowledge discovery in Twitter streaming data," in Proceedings of the 13th International Conference on Discovery Science (pp. 1–15). Berlin, Germany: Springer (2010).