Comparative Study on Sentiment Analysis in Twitter with Lightweight Discourse Analysis

Naveen Kumar,Berk Atmaca
IMS,University of Stuttgart,
Germany

August 25, 2016

Abstract

We perform a comparative study on lightweight method for using discourse relations for sentiment analysis on short text message or tweets.This work is based on Mukherjee and Bhattacharyya (2012) paper on Sentiment Analysis in Twitter with Discourse Analysis. This method deals with noisy and unstructured data like twitter or data from other micro blogging platform which makes it even more complex as it cannot be handled by parsers. The approach analyses the impact of discourse connective and conditionals like but, although , etc on polarity of sentences.It also takes into account the influence of modals and negation in a sentence for polarity detection.Most of the baseline models uses bag of word approach for sentiment analysis.In this work, we create model for sentiment analysis to predict sentiment of a sentence.We train our model using SVM algorithm. The work presents a comparative study between base line model and models which takes into account discourse features and further analyses the better performance of discourse feature vector.

1 Introduction

Discourse relations in a sentence plays an important role for the semantic orientation in terms of polarity of the sentence. We performed an analysis on the impact, a discourse vector creates on the accuracy metrics in comparison to base line approach. The work compares discourse based approach with the state of art system, providing favorable impact of discourse on sentence segments.This work is focused on noisy and unstructured short text messages such as tweets. Tweets are short text messages which are limited to a maximum length of 140 characters which necessitates the expression of sentiment in short and precise manner.

The discourse connectives establish a coherent relations between phrases in a text which impacts sentiment of the sentence. Here is an example demonstrating its influence in the text. *"@user share 'em! I'm quite excited about Tintin, despite not really liking original comics. Probably because Joe Cornish had a hand in."*.The overall sentiment is positive in spite of having both positive and negative words. This is due to the discourse connective *despite* which gives more weight to previous discourse segment. The state of the art method of bag

of word approach does not take this into account and hence will impact the accuracy metrics.

In this work, we do comparative study on two different scenarios. The first is two class classification where the classes are *positive and negative*, while the other is three class classification with *positive, negative and objective non-spam.*The process includes data pre-processing, feature identification and extraction, training the model and testing it.The process is further repeated for two different scenarios, one of it goes ahead with the pre-processed data while the other takes stemming of the data into consideration. We can thus compare the results in this data setup.

2 Method

2.1 Preprocessing

We use the data crawled from twitter as our training data. This labeled dataset is preprocessed to remove junk information and to obtain a better structured text. Following are the main rules applied as part of the pre-processing steps :

- Urls present as part of the tweets are removed.
- Usernames in tweets usually starts with @ symbol (e.g. @MyCoolSelf). All strings starting with @ are replaced by "[@user]".
- Re-tweeted messages contain RT tags. We removed such tags.
- Stemming of the data is done on Porter Stemmed algorithm for stemming data experiment setup.
- Removal of tweets with class label "[objspam]", as it indicates objective spam category which is not taken in consideration in classification.

2.2 Feature Sets

We implement different features for sentiment classification which includes basic features like ngrams, advanced features like discourse connectives and also external lexical resources.

Ngrams: This is the base line model or state-of-art feature set for classification scenario. As feature set, we use unigram and bigram extracted from training data set.

Discourse Features. Discourse feature comprises of connectives,modals,negations and conditionals.The Discourse features are based on Mukherjee and Bhattacharyya (2012) and we are thus trying to compare results as mentioned in the above mention paper.Table 1,show different attributes which are essential for sentiment analysis. In this work, we take into consideration Conj_Infer,Conj_Fol,Conj_Prev, Conditionals,String_Mod and Negation.

POS Tag. We included Part of speech tag as one of the feature in the experimental setup. It also gave us an scope for using POS tag feature just for noun,adjective and adverbs and thus filter out the rest.

Relations	**Attributes**
Conj_Fol	*but, however, nevertheless, otherwise, yet, still, nonetheless*
Conj_Prev	*till, until, despite, in spite, though, although*
Conj_Infer	*therefore, furthermore, consequently, thus, as a result, subsequently, eventually, hence*
Conditionals	*If*
Strong_Mod	*might, could, can, would, may*
Weak_Mod	*should, ought to, need not, shall, will, must*
Neg	*not, neither, never, no, nor*

Table 1: Discourse Relations and Semantic Operators Essential for Sentiment Analysis.

Algorithm 1 Discourse Feature weights

$f_{ij} \leftarrow$ *Weight of the word in sentence , initialized to 1*
$flip_{ij} \leftarrow$ *Indicates weather the polarity of word should be flipped or not*
$hyp_{ij} \leftarrow$ *Indicates the presence of conditional or strong modal in sentence*
while $i < m$ **do** ▷ where m is number of sentences
  while $j < n$ **do** ▷ where n is number of words in a sentences
    hyp_{ij} *= 0*
    if $word_{ij} == Conditional || Strong_Mod$ **then**
      hyp_{ij} *= 1*
    end if
    f_{ij} *= 1*
    if $word_{ij} == Conj_Fol || Conj_Infer$ **then**
      k *= j+1*
      while $k < n$ **do** ▷ where n is number of words in a sentences
        f_{ik} *+ = 1*
        k*++*
      end while
    end if
    if $word_{ij} == Conj_Prev$ **then**
      k *= j-1*
      while $k < n$ **do** ▷ where n is number of words in a sentences
        f_{ik} *+ = 1*
        k*++*
      end while
    end if
    $flip_{ij}$ *= 1*
    if $word_{ij} == Neg$ **then**
      k *= 1*
      while $k < Negation_Window$ **do**
        $flip_{i,j+k}$ *= -1*
        k*++*
      end while
    end if
  end while
end while

Features Used	2-Class	2-Class (Stemmed)	3-Class	3-Class (Stemmed)
Ngram (Base Line Model)	67.79	67.87	43.78	44.55
Ngram + Discourse	67.87	67.89	46.03	46.12
Ngram + Discourse + POS Tags	67.55	67.58	43.006	43.39
Bigram + Discourse	67.89	67.84	46.96	47.643
Bigram + Discourse + SentiWordNet	67.95	-	46.23	-

Table 2: Contributions of different features to the cross validation accuracy

Senti WordNet. SentiWordNet is a lexical resource for opinion mining. SentiWordNet assigns to each synset of WordNet three sentiment scores: positivity, negativity, objectivity. We cannot use senti wordnet in case of stemmed data setup, as senti wordnet cannot give output from stemmed words.

3 Experiment

3.1 Data

We use the data provided as part of Mukherjee and Bhattacharyya (2012) work. The data is manually annotated with four class labels : positive, negative, objective non spam and objective spam. We use positive and negative data set for 2 class classification setup while include objective non spam data for 3 class classification setup. We get rid of objective spam data as its worthless in our experimental setup. Initially we have preprocessed data, but for the experimental setup of stemmed data setup, the does is processed through Porter stemmer algorithm.

3.2 Training

The model is trained using linear SVM as the base paper of Mukherjee and Bhattacharyya (2012) presents us with the approach of using linear SVM classifier for training the model and further the results were cross validation accuracy.We follow the same line of work and the results shown are the cross validation accuracy measures.

3.3 Feature Selection

In order to select the best features, we add the features incrementally to the model.Table 2 shows the result of this feature selection experiment.

4 Result

The results shown in Table 2 refers to the cross validation accuracy in different experimental setup.In 2-class classification setup, we get the best accuracy measure with just pre-processed text without any stemming by taking *Bigram + Discourse + SentiWordNet* features. In 3-class classification the accuracy measure for stemmed data use case by taking *Bigram + Discourse* features is best.

Features Used	Positive	ObjNonSpam	Negative
Precision	48.99	44.44	48.72
Recall	33.73	1.70	79.83

Table 3: Test Data Accuracy Measures for 3-class classification

Features Used	Positive	Negative
Precision	67.81	50.0
Recall	99.80	0.40

Table 4: Test Data Accuracy Measures for 2-class classification

The cross validation accuracy results in Table 2 refers to the point mention in the introduction of this report, that is *Discourse* feature does makes an impact on the accuracy measures.It may be a slight improvement in accuracy but the matter of fact is it does a positive impact as feature vector. In the case of 2-class classification , the best results are achieved by using set of 3 features: *Discourse, Bigrams, SentiWordNet*. The comparison between the base line model and this model is the differnece is accuracy by 0.16%. In the case of 3-class classification, the best results were obtained by using the feature set of *Discourse and Bigrams* . The difference in accuracy measure between this model and base line model is 3.093%.The above results are evident to the fact that discourse feature are relevant in increasing the accuracy measure.

The accuracy measure in Table 2 does not reflects to a good classification model.Hence, we conducted an error analysis study on the best obtained result. We divide the data into training and test data to further get detailed analysis on the underlying predictions for each class.

4.1 Error Analysis

In the case of 3-class classification, we can see from Table 3 that recall for ObjNonSpam class is significantly low for the predictions. This indicates that the coverage for ObjNonSPam class is rather poor for the model which gives best cross validation accuracy results using *Discourse and Bigram* features.The test data used for testing the model for 3-class classification has : 504(Positive), 600(Negative) and 235(Neutral) as annotated classes.The accuracy measure for test data is 48.76.

In 2-class classification , the test data specification are : Positive(515) and Negative(245). The accuracy measure for test data is 67.76 using features *Discourse, Bigrams and SentiWordNet*. The analysis on the results in Table 4 shows that the recall for Negative class is very poor. It shows the coverage in model to distinguish or classify negative class instance is rather low and that's reflected in the test data results.

The account of the issue of recall in both, 3-class and 2-class classification reflects to the lack of features which can significantly detect the Neutral and Negative classes in the two experimental setup respectively. The low recall reflects to the problem of losing out relevant documents from that class.

5 Summary

In this work, we compared the results for base line model and models using discourse features, to verify the impact on accuracy measures presented by Mukherjee and Bhattacharyya(2012). The works checks the impact of discourse feature in different experiment setup for 2-class and 3-class classification.The project further does error analysis on the best results and finds out the issue for lower accuracy measures, as significantly lower recall for Neutral class in 3-class classification problem and Negative class for 2-class classification problem.

6 Future scope

For further improving the result, we would like to concentrate on the feature improvements by experimenting with features including filtering out the features based on mutual information. We can use some opinion word dictionaries or sentiment lexicons for further improving the accuracy measures and focus on increasing the recall measure. The error analysis can further widen its scope of analyzing results for different experiment setup and compare the precision/recall measures for each.

References

[1] Subhabrata Mukherjee and Pushpak Bhattacharyya, *Sentiment Analysis in Twitter with Lightweight Discourse Analysis*, 2012