Scientific Journal Impact Factor (SJIF): 4.364
International Journal of Modern Trends in Engineering and Research
IJMTER
ISSN (ONLINE) : 2349 - 9745
ISSN (PRINT) : 2393 - 8161

DUAL SENTIMENT ANALYSIS

AdhangleSahyadri P[1], BhavnaniMamta G[2] and KadamSneha S[3]
[1,2,3]Computer Dept ,AVCOE,Sangamner

Abstract- In recent years, Bag-of-words (BOW) is most popular way to model text in statistical machine learning (such as naïve Bayes, maximum entropy classifier, and support vector machines) approaches in sentiment analysis. On this basis, we propose a dual training & dual prediction algorithm which make use of original and reversed training reviews for learning a sentiment classifier and a dual prediction algorithm to classify the test reviews by considering two sides of one review. Finally, we are implementing machine learning approach by using Bayesian classifier which removes DSA's dependency on an external antonym dictionary for review reversion. Machine learning algorithm under supervised learning, we are implementing Naive Bayes for classification of these reviews for further implementation.We have developed a corpus-We conduct a wide range of experiments including two antonym dictionaries, BOW, classification algorithms like Naive Bayes. The results demonstrate the effectiveness of DSA in supervised sentiment classification. We have also developed a corpus- based method that can construct a pseudo-antonym dictionary, which removes DSA’s dependency on an external antonym dictionary for reversion of reviews

Keywords- Sentimental analysis, feedback, Opinion Mining, Classification, Sentiment Identification.

I. INTRODUCTION

Data Mining is one of the important step of the “Knowledge Discovery in Databases” processes or KDD, which relevance patterns from large datasets. Also, it includes the techniques and design of artificial intelligence, machine learning and statistics. Opinion mining or sentiment analysis is to evaluate the users’ opinions or thoughts which are in the form of unstructured data. To interpret and understand the person's views, emotions and understanding, the system must be made reliable and efficient. There are two techniques used in Sentiment Classification.

- Machine Learning Approach
- Lexicon Based Approach

Machine Learning techniques include supervised and unsupervised learning approaches. Supervised learning consists of some classifier such as Decision tree, Liner, Rule-based and Probabilistic classifiers. Lexicon Based approaches are confidential into Dictionary based and Corpus-based methods. The corpus-based method further divided into the statistical and semantic approach. BOW MODEL Sentiment classification is a essential task in sentiment analysis, with its aim to classify the sentiment of a given text. The familiar practice in sentiment classification follows the techniques in conventional topic-based text classification, where the bag-of-words (BOW) standard is typically used for text representation. In a review text is represented by a vector of independent words. A large number of researches in sentiment analysis aimed to appreciate BOW by consolidating linguistic knowledge [6], [10].

II. RELATED WORK

2.1 POLARITY SHIFT

Polarity shift is linguistic kind of phenomenon which can reverse the sentiment polarity of the text. Negation is the most essential type of polarity shift. For example, by adding a negation word

DOI:10.21884/IJMTER.2017.4094.XJ8OJ

“don’t” to a positive text “I like this movie” since the word “like”, the sentiment of the text will be reversed from positive to negative. By the BOW representation the two sentiment-opposite texts are acknowledged to be very similar .[10], [11].

2.2PROCESS OF SENTIMENT ANALYSIS:

Radically the sentiment analysis is done with the process as shown in the figure 1. There are total four prime steps:

- Text Extraction -This step implicates extracting words from text that influence the outcome of the result

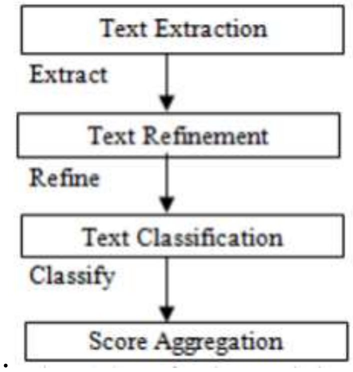

Fig 1. Sentiment Analysis Process

- Text Refinement -This step has involvement of refining text in form of relevant phrases, words etc.
- Text Classification – This step includes classification of text into its class as positive, negative.
- Score Aggregation – This step assembles total scores from classifier and then aggregates it forward to produce the total sentiment score.

2.3 DUAL SENTIMENT ANALYSIS

Dual sentiment analysis (DSA) is implement to address the polarity shift problem in sentiment classification. The original and reversed reviews are designed in a one-to-one

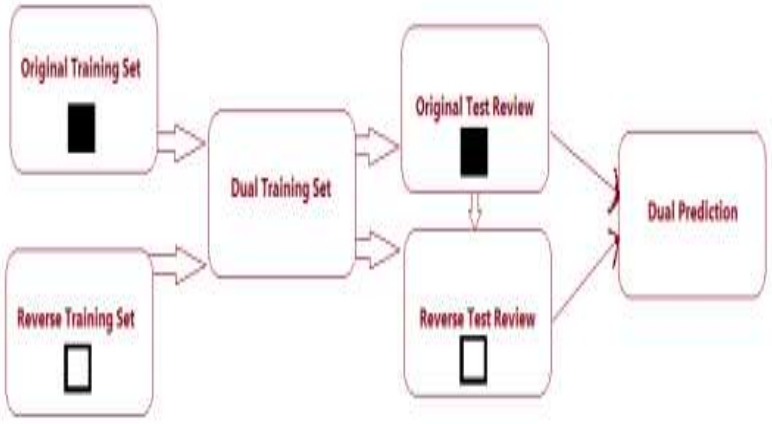

Fig 2. Process of Dual Sentiment Analysis

correspondence. In DSA a dual training (DT) algorithm and a dual prediction (DP) algorithm appropriately, to make use of the reversed and original samples in pairs for training a statistical classifier and make predictions. Also DSA framework is implemented for 3-class (positivenegative-neutral) sentiment classification, by getting the neutral reviews into consideration in both dual prediction [1] and dual training

Three Classes of Sentiment Analysis

a. Positive Sentiments:

These are the good words about the target in consideration. If the positive sentiments are raised, then it's be good. If the positive reviews about the commodities are more, then it is bought by many customers.

b. Negative Sentiments:

These are the bad words about the target in deliberation.It is discarded from the optional list, If the negative sentiments are increased and In case of commodity reviews, if the negative reviews about the commodities are more, no one intend to buy it.

Table.1 AN EXAMPLE OF CREATING REVERSED TRAINING REVIEWS

	Review Text	Class
Original review	*I don't like this song. It is boring.*	Negative
Reversed review	*I like this song. It is interesting.*	Positive

Thereafter, we propose a dual training (DT) algorithm and a dual prediction (DP) algorithm respectively, to make use of the reversed and original samples in pairs for training a statistical classifier and make predictions. In DT, the classifier is learnt by maximizing a combination of likelihoods of the reversed and original training data set. In DP, predictions are made by considering two sides of one review. That is, we measure not only how positive or negative the original review is, but also how negative or positive the reversed review is.

We further extend our DSA framework from polarity (positive vs. negative) classification to three-class (positive vs. neutral vs. negative) sentiment classification, by taking the neutral reviews into consideration in both dual training and dual prediction. As neutral text is part of input data, we have to have taken them into consideration.

2.4 DATA EXPANSION TECHNIQUE

The data expansion technique has been seen in the field of handwritten recognition [3], [40], where the performance of the handwriting recognition systems was significantly improved by adding some synthetic training data.

In the field of natural language processing and text mining, expanding the amount of labeled data through a Web search using unique expressions in definitions for the task of word sense disambiguation. Fujita and Fujino [11] proposed a method that provides reliable training data using sentences from an external dictionary.

To the best of our knowledge, the data expansion technique proposed here is the first work that conducts data expansion in sentiment analysis. Different from the above techniques, the original and reversed reviews are constructed in a one-to-one correspondence. Another novel point of this work is that we expand the data set not only in the training stage, but also in the test stage. The original and reversed test review is used in pairs for sentiment prediction.

2.5 DATA EXPANSION BY CREATING REVERSED REVIEWS

In this section, we introduce the data expansion technique of creating sentiment-reversed reviews. This system is based on an antonym dictionary, for each original review, the reversed review is created according to the following rules:

Text Reversion: If there is a negation, we first detect the scope of negation. All sentiment words which are out of the scope of negation are reversed to antonyms. In the scope of negation, negation words (e.g., “*no*”, “*not*”, “*don’t*”, etc.) are removed, but the sentiment words are not reversed; Label Reversion: For each of the training reviewes, the class label is reversed to its opposite (i.e., positive to negative, or vice versa), as the class label of the reversed review.

III. DUAL TRAINING ANALYSIS

In this section, we present our dual sentiment analysis (DSA) framework in detail. Fig. 1 illustrates the process of a DSA algorithm. It contains two main stages: 1) dual training (DT) and 2) dual prediction (DP). In the follow-ing two subsections, we will introduce them respectively.

3.1 Dual Training

In the training stage, all of the original training samples are reversed to their opposites. We have refer to them as “original training set” and also “reversed training set” respectively. In our data expansion technique, there is a one-to-one correspondence between the original and reversed re-views. The classifier is trained by maximizing a combination of the likelihoods of the original and reversed training samples. This process is called dual training (DT).

We need to check semantic relations between these synonym sets. Using the antonym thesaurus it is possible to obtain the words and their opposites.

The WordNet antonym dictionary is direct and simple However, in many languages other than English, such an antonym dictionary may not be readily available. Even if we can get an antonym dictionary, it is still hard to guarantee vocabularies in the dictionary are domain consistent with our tasks.

For simplicity, in this paper we derive the DT algorithm by using the logistic regression model as an example. Note that our method can be easily adapted to the other classifiers such as naïve Bayes and SVMs.[3] In the experiments, all of the three classification algorithms are examined. To solve this , we develop a corpus-based method to construct a pseudo-antonym dictionary. This corpus-based pseudo-antonym dictionary can be learnt using the labeled training data only. The basic idea is to first use mutual information to identify the most positive-relevant and the most negative-relevant features, rank them in two separate groups, and pair the features that have the same level of sentiment strength as pair of antonym words.

3.2 Dual Prediction

Let us use the example in Table 1 again to explain why dual prediction works in addressing the polarity shift problem. This time we assume “*I like this book. It is boring*” is an original test review, and “*I don’t like this book. It is interesting*” is the reversed test review. In traditionalBOW, “*like*” will contribute a high positive score in predicting overall orientation of the test sample, despite of the negation structure Hence, it is very like that the original test review will be mis-classified as Positive. While in DP, due to the removal of negation in that review, “*like*” this time the plays a positive role.The probability that the reversed review being classified into Positive must be high. In DP, a weighted combination of two component predictions is used as the dual prediction output. In this manner, the prediction error of the original test sample can also be compensated by the prediction of the reversed test sample.This can reduce some errors caused by polarity shift. In the experimental study, we will extract some real examples from our experiments to prove the effectiveness of both dual training and dual prediction.

3.3 DSA with Selective Data Expansion

Let us first observe two reviews which are a bit more complex than the previous examples:

- Review (a): *The book is very interesting, and the price isvery cheap. I like it.*
- Review (b): *The book is somehow interesting, but the priceis too expensive. I don’t dislike it.*

In review (a), the sentiment is very strong and the polarity shift rate is low. In this case, the original review itself is a good labeling instance, and the reversed review will also be a good one. In review (b), the sentiment polarity is less distinct. In this case, the sentiment polarity of the reversed review is also not distinct and confident. Therefore, creating reversed review for review (a) is not that necessary in comparison with review (b).

Consequently, we propose a sentiment degree metric for selecting the most sentiment-distinct training reviews for data expansion.

IV. THE ANTONYM DICTIONARY FOR REVIEW REVERSION

So far we have presented the DSA model. However, we notice that DSA highly depends on an external antonym dictionary for review reversion. How to construct a suitable antonym dictionary by applying DSA into practice? It still remains an important problem.

4.1 The Lexicon-based Antonym Dictionary

In the languages where lexical resources are abundant, there is straightforward way is to get the antonym dictionary directly from the well-defined lexicons, such as WordNet[4] in English. WordNet is a lexical database which is used to group the English words into sets of synonyms called synsets, provides short, general definitions, and records the various

It is important to notice that, rather than a commonsense antonym dictionary, it is a “pseudo” antonym dictionary, Here, “pseudo” means a pair of antonym words are not really semantic-opposite, but have opposite sentiment strength. As we have stated in Section 3, both the original and created reviews are represented as a vector of independent words in the BOW representation. Therefore, it is not that important whether the created review is grammatically correct or not. We just need to maintain the level of sentiment strength in review reversion. Ap- parently, the mutual information provides a good measure of the contextual sentiment strength. Therefore, the condition of the same level sentiment strength can be re- the positive- and negative-relevant words with the same ranking posititions as antonyms. Moreover, because the pseudo-antonym dictionary islearnt from the training corpus, it has a good property: language-independent and domain-adaptive. This prop-erty makes the DSA model possible to be applied into a wider range, especially when the lexical antonym diction-ary is not available across different languages and do-mains.

In the experimental study, we will evaluate the effect of the MI-based pseudo-antonym dictionary by conducting experiments on two Chinese datasets to. We also compare the results of two kinds of antonym dictionaries on the English multi-domain sentiment datasets, and provide some discussions on the choice of them in real practice.

4.2 Machine Learning with Naïve Bayes-

Bayes Classifier: The mathematics A naive Bayes classifier applied Bayes Theorem in an attempt to suggest possible classes for any given text. To do this, there is need of number of previously classified documents of the same type. The theorem is as follows:

Bayes Classifier example: Content of document analysis As an example, let us try and find the probability that a content(the document) can be classified as positive (the class). At first glance the theorem can be confusing, so let’s simplify it a bit by breaking down the various components:

$P(A|B)$

This formula can be read as the probability of A, the , given B, the Content. This is the end result we’re looking for.

$P(B|A)$

This can be read as the probability of B, the content, given A, the class. This is determined by previously gathered information.

P(A)

This is the probability of A – the class. It’s independent of all other probabilities.

P(B)

This is the probability of B – the Content. It’s independent of all other probabilities.

$$P(Positive/Content)= \frac{P(Content/Positive).P(Positive)}{P(Content)}$$

Since the probability of the Content, *P(Content),* is constant, it can be disregarded in our calculations. We’re only interested in the probability of the Content given the class, *P(Content|positive),* and the probability of the class, *P(positive):*

For the sake of this example, let’s say there’s three possible classes: positive, negative and neutral. That gives any Content a one in three (or 33%) chance of falling into any of those classes. That gives us *P(positive) = 0.33333*.

P(Content|positive)

To calculate *P(Content|positive),* we need a training set of Contents that were already classified into the three categories. This gives us a basis from which to compute the probability that a Content will fall into a specific class. Since the chances are relatively low that we’ll find a specific Content in the training set, we’ll tokenize the Content and for each word calculate the probability in the training set. This gives us the formula of:

*P(Content|positive) = P(T1|positive) * P(T2|positive) * .. * P(Tn|positive)*

Where T1 to Tn is all the words in the Content.

P(Ti|positive)

To determine the probability of a specific word falling into the category we’re testing, we’ll need the following from the training set:

- The number of times Ti occurs in Contents that were marked as positive in the training set.
- The total number of words of Contents that were marked as positive in the training set.

There are various ways in which you can get these numbers, so we will not go into specifics here

As an example, let’s look at the word “food”, with the following numbers:

- Number of times *food* occurs in positive Contents: 455
- Number of words in positive Contents: 1211

So to calculate the relative probability of *food* occurring in the the *positive* category, we divide 455 by 1211, giving us 0.376. Since food can have positive, neutral and negative interpretations, it’s not surprising that its relative probability is 37%. This process now needs to be repeated for each word in the Content.

Since we now have the ability to calculate the probabilities that each word in the Content can be classified as positive, let’s calculate the probability that the whole Content can be classified as positive.

*P(positive|Content)=P(Content|positive)*P(positive)*

For this example, let’s say the Content was “I love good food”, and the probabilities we calculated were 25%, 62.5%, 74% and 42.5% respectively.

*P(positive|Content)=P(Content|positive)*P(positive)*

*= P(T1|positive) * .. * P(Tn|positive) * P(positive)*

*= 0.25 * 0.625 * 0.74 * 0.425 * 0.33*

= 0.016216406

This procedure can now be used to find the relative probability for each of the classes. From the training set, we calculate P(negative|Content) as 0.000003125 and P(neutral|Content) as 0.0082809375. Once we have the probability for each class, we n compare the classes, and use the highest ranked class as the class for the document. Intuitively, it makes sense to classify the sentence " *I love good food"* as positive, but we have now mathematical proof, which is based on gathered data, that can be classified as positive.

For sentiment classification by supervised learning approach we will use this algorithm.

V. CONCLUSION AND FUTURE WORK

In this Project we aim to serve, a novel data expansion approach, called dual sentiment analysis (DSA), to address the polarity shift problem in sentiment classification. The basic idea of DSA is to create reversed reviews that are sentiment-opposite to the original reviews, and make use of the original and reversed reviews in pairs to train a sentiment classifier and make predictions. The experimental results show that using a selected part of training reviews for data expansion can yield better performance than that using all reviews.

Finally, to remove DSA's dependency on an external antonym dictionary, we propose a corpus-based method to construct a pseudo-antonym dictionary.

In this paper, we focus on creating reversed reviews to assist supervised sentiment classification. In the future, we can generalize the DSA algorithm to a wider range of sentiment analysis tasks. We also plan to consider more complex polarity shift patterns such as transitional, sub-junctive and sentiment-inconsistent sentences in creating reversed reviews.

REFERENCES

[1] A. Abbasi, S. France, Z. Zhang, and H. Chen, "Selecting attributes for sentiment classification using feature relation networks," IEEE Trans-actions on Knowledge and Data Engineering (TKDE), vol. 23, no. 3, pp.447-462, 2011.

[2] E. Agirre, and D. Martinez, "Exploring automatic word sense disambiguation with decision lists and the Web," Proceedings of the COLINGWorkshop on Semantic Annotation and Intelligent Content, pp. 11-19,2000.

[3] J. Cano, J. Perez-Cortes, J. Arlandis, and R. Llobet, "Training set expansion in handwritten character recognition," Structural, Syntactic,and Statistical Pattern Recognition, pp. 548-556, 2002.

[4] Y. Choi and C. Cardie, "Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis," Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 793-801, 2008.

[5] I. Councill, R. MaDonald, and L. Velikovich, "What's Great and What's Not: Learning to Classify the Scope of Negation for Improved Sentiment Analysis," Proceedings of the Workshop on negation and speculation in natural language processing, pp. 51-59, 2010.

[6] S. Das and M. Chen, "Yahoo! for Amazon: Extracting market sentiment from stock message boards," Proceedings of the Asia Pacific Finance Association Annual Conference, 2001.

[7] K. Dave, S. Lawrence and D. Pen-nock, "Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews,"Proceedings of the International World Wide Web Conference (WWW), pp.519-528, 2003.