International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Accurate Sentiment Analysis using Enhanced Machine Learning Models

Rincy Jose[1], Varghese S Chooralil[2]

[1]Department of Computer Science and Engineering, Rajagiri School of Engineering and Technology, Kochi, India

Abstract: ***Sentiment analysis is the computational study of opinions, sentiments, subjectivity, evaluations, attitudes, views and emotions expressed in text. Sentiment analysis is mainly used to classify the reviews as positive or negative or neutral with respect to a query term. This is useful for consumers who want to analyse the sentiment of products before purchase, or viewers who want to know the public sentiment about a new released movie. Here i present the results of machine learning algorithms for classifying the sentiment of movie reviews which uses a chi-squared feature selection mechanism for training. I show that machine learning algorithms such as Naive Bayes and Maximum Entropy can achieve competitive accuracy when trained using features and the publicly available dataset. It analyse accuracy, precision and recall of machine learning classification mechanisms with chi-squared feature selection technique and plot the relationship between number of features and accuracy using Naive Bayes and Maximum Entropy models. Our method also uses a negation handling as a pre-processing step in order to achieve high accuracy.***

Keywords: Sentiment Classification, Negation Handling, sentiment Analysis, Feature Selection

1. Introduction

Sentiment analysis can be considered as the use of natural language processing, text analysis and computational linguistics to identify and extract sentiment information in source materials. Generally, sentiment analysis aims to find the attitude of a writer with respect to some relevant topic or the overall contextual polarity of a document.

The main task in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature level — whether the expressed opinion in a document, a sentence or a feature is positive, negative, or neutral. Document level sentiment analysis is the classification of the overall sentiments mentioned by the reviewer in the whole document text in positive, negative or neutral classes.

Sentiment Classification techniques can be roughly divided into machine learning approach, lexicon based approach and hybrid approach [1]. The Machine Learning Approach (ML) applies the famous ML algorithms and uses linguistic features. The Lexicon-based Approach relies on a sentiment lexicon, a collection of known sentiment terms. It is divided into dictionary-based approach and corpus-based approach which use statistical or semantic methods to find sentiment polarity. The hybrid Approach combines both approaches.

The accuracy of a sentiment analysis is based on how well it agrees with human judgments. This can be measured by using precision and recall [2].

In this paper we try to compare the accuracy of different enhanced machine learning sentiment analysis methods. They are Naïve Bayes and maximum entropy models with chi-squared feature selection technique and negation handling. Section 2 contains detailed study of these two methods. Section 3 implementation details and results. Section 4 is the conclusion.

2. Methodology

Our proposed system mainly consists of three modules. They are

A. Negation handling
B. Feature selection
C. Sentiment classification.

3.1 Negation handling

Negation handling is one of the factors that contributed significantly to the accuracy of our classifier. A major problem occurring during the sentiment classification is in the negation handling. Since here we use each word as feature, the word ―good" in the phrase ―not good" will be contributing to positive sentiment rather than negative sentiment .This will leads to the errors in classification. This type of error is due to the presence of ―not" and this is not taken into account. To solve this problem we applied a simple algorithm for handling negations using state variables and bootstrapping. We built on the idea of using an alternate representation of negated forms[3]. This algorithm stores the negation state using a state variable. It transforms a word followed by a n't or not into ―not_" + word form. Whenever the negation state variable is set, the words read are treated as ―not_" + word. When a punctuation mark is encountered or when there is double negation, the state variable will reset.

Many words with strong sentiment occur only in their normal forms in their training set. But their negated forms would be of strong polarity. We solved this problem by adding negated forms to the opposite class along with normal forms during the training phase. That is if we encounter the word ―bad" in a negative document during the training phase, we increment the count of ―bad" in the negative class and also increment the count of ―not_bad" for the positive class. This is to ensure that the number of ―not_" forms are sufficient for classification. This modification resulted in a significant improvement (1%) in classification accuracy due to bootstrapping of negated forms during training.

3.2 Feature Selection

The next step in the sentiment analysis is to extract and select text features. Here feature selection technique treats the documents as group of words (Bag of Words (BOWs)) which ignores the position of the word in the document. Here feature selection method used is Chi-square (x2).

A chi-square test also referred to as a statistical hypothesis test in which the sampling distribution of the test statistic is a chi-square distribution when the null hypothesis is true. The chi-square test is used to determine whether there is a significant difference between the expected frequencies and the observed frequencies in one or more categories.

Assume n be the total number of documents in the collection, pi(w) be the conditional probability of class i for documents which contain w, Pi be the global fraction of documents containing the class i, and F(w) be the global fraction of documents which contain the word w. Then, the x2-statistic of the word between word w and class i is defined[1] as:

$$\chi_i^2 = \frac{n \cdot F(w)^2 \cdot (p_i(w) - P_i)^2}{F(w) \cdot (1 - F(w)) \cdot P_i \cdot (1 - P_i)}$$

PMI is another method of measuring the correlation between terms and classes. x2 is better than PMI as it is a normalized value. So, these values are more comparable across terms in the same class. So x2 is used in our experiment.

3.3 Sentiment Classification

Classification is done on the extracted features.

1) Naive Bayes Classifier

Naïve Bayes is a probabilistic learning approach which assumes, the terms in documents are independent. Suppose a collection of N documents dj for j=1toN, and each document is represented as a number of terms dj = {t1, t2 . . . tn}, then the probability of a document dj occurring in class positive or negative is given as[4]:

$$P(c_k \mid d_j) = P(c_k) \prod_{i=1}^{I} P(t_i \mid c_k).$$

Where P (c_k) is the prior probability of a document occurring in class ck and P(t_i |c_k) is the conditional probability of the term ti occurs in a document of class ck. Here the terms are opinion carrying words. P (t_i | c_k) and P(c_k) are calculated from the training data. We use NB as the base of classification. That is, it uses Bayes Theorem to predict the probability that the feature set of given document belongs to a particular class. We have made use of NLTK for NB classification.

2) Maximum Entropy Classifier

Maximum Entropy models are feature-based models. Maximum Entropy makes no independence assumptions for its features, unlike Naive Bayes. The model is represented by the following [8]:

$$P_{ME}(c|d,\lambda) = \frac{\exp[\Sigma_i \lambda_i f_i(c,d)]}{\Sigma_{c'} \exp[\Sigma_i \lambda_i f_i(c,d)]}$$

Where c is the class, d is the document, and λ is a weight vector. The weight vectors decide the significance of a feature in this classification. A high weight means that the feature is a strong indicator for the class. The weight vector is calculated by numerical optimization of the λi's in order to maximize the conditional probability. Here we used the Python NLTK library[7] to train and test using the Maximum Entropy method. For training the weights i used conjugate gradient ascent.

3. Implementation Details

I have used publicly available movie review data set "rt-polaritydata" for training and testing the two classifiers. We trained the classifiers on 7996 instances and tested on 2666 instances. Thus we measured accuracy, precision and recall for different number of features.

Table 3.1: Classifier Accuracy Without negation handling

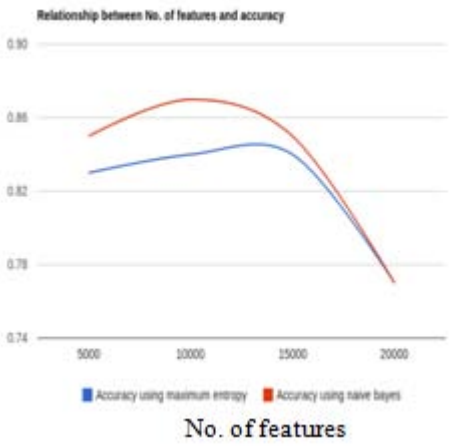
Number of features	Naive Bayes	Max Entropy
All words	.77	.77
5000	.85	.83
10000	.85	.84
15000	.85	.84
20000	.77	.77

Table 3.2: Classifier Accuracy With negation handling

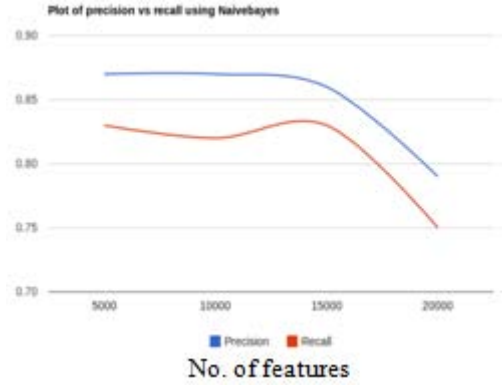
Number of features	Naive Bayes	Max Entropy
All words	.77	.77
5000	.86	.84
10000	.86	.85
15000	.86	.85
20000	.77	.77

From this we can understand that naive bayes out performs max entropy in terms of accuracy and speed. It is also clear that "rt-polarity" dataset show highest accuracy when it is trained with features between 5000 and 15000.

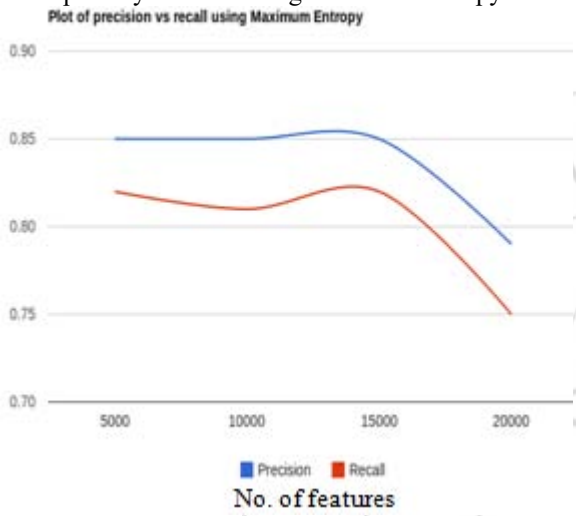
The graph given below shows relationship between number of features and accuracy when it is trained and tested with "rt-polarity" dataset.



The graph given below shows relationship between number of features and recall/precision when it is trained and tested with "rt-polarity" dataset using naive bayes classifier.



The graph given below shows relationship between number of features and recall/precision when it is trained and tested with ―rt-polarity" dataset using maximum entropy classifier.



4. Conclusion

In this paper, we analyzed and compared performance of two machine learning sentiment analysis techniques by using a movie review dataset. From the accuracy comparison we reached at a conclusion that naïve bayes outperforms max entropy. We experiment on these two techniques by applying negation handling also. It results in 1% improvement in classification accuracy. Another accuracy improvement done is that application of chi-square feature selection mechanism.

References

[1] Walaa Medhat a, Ahmed Hassan b, Hoda Korashy b, ―Sentiment analysis algorithms and applications: A survey", Ain Shams Engineering Journal –science direct 2014.
[2] N. D. Valakunde, Dr. M. S. Patwardhan, ―Multi-Aspect and Multi-Class Based Document Sentiment Analysis of Educational Data Catering Accreditation Process", 2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies.
[3] Alexandre Trilla,Francesc Alías, ―Sentence-Based Sentiment Analysis for Expressive Text-to-Speech", IEEE transactions on audio, speech, and language processing, vol. 21, no. 2, february 2013.
[4] Vivek Narayanan1, Ishan Arora2, Arjun Bhatia3, ―Fast and accurate sentiment classification using an enhanced Naive Bayes model", 2012.
[5] Bo Pang1, Lillian Lee2,"Opinion Mining and Sentiment Analysis", Foundations and Trends in Information Retrieval" Vol. 2, Nos. 1–2 (2008).
[6] Bing Liu, ―Sentiment Analysis and Opinion Mining", Morgan &Claypool Publishers, May 2012.
[7] Python Natural Language Toolkit - http://www.nltk.org/
[8] Ravikiran Janardhana ,Department of Computer Science, University of North Carolina at Chapel Hill, ―Twitter Sentiment Analysis and Opinion Mining",2012.

Author Profile

Rincy Jose received the B.Tech. Degree in Computer Science from NSS College of Engineering in 2012 and M.Tech. Degree in Computer Science and Information System from Rajagiri School of Engineering and Technology in 2015