Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 57 (2015) 821 – 829

Procedia Computer Science

3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)

Classification of Sentimental Reviews Using Machine Learning Techniques

Abinash Tripathy[a,*], Ankit Agrawal[b], Santanu Kumar Rath[c]

[a]Department of Computer Science and Engineering, National Institute of Technology, Rourkela, 769008, India
[b]Department of Computer Science and Engineering, National Institute of Technology, Rourkela, 769008, India
[c]Department of Computer Science and Engineering, National Institute of Technology, Rourkela, 769008, India

Abstract

Sentiment Analysis is the most prominent branch of natural language processing. It deals with the text classification in order to determine the intention of the author of the text. The intention can be of admiration (positive) or criticism (Negative) type. This paper presents a comparison of results obtained by applying Naive Bayes (NB) and Support Vector Machine (SVM) classification algorithm. These algorithms are used to classify a sentimental review having either a positive review or negative review. The dataset considered for training and testing of model in this work is labeled based on polarity movie dataset and a comparison with results available in existing literature has been made for critical examination.

Peer-review under responsibility of organizing committee of the 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)

Keywords: Sentiment Analysis; Naive Bayes (NB); Support Vector Machine (SVM); Classification; Polarity Movie Dataset

1. Introduction

Sentiment mainly refers to feelings, emotions, opinion or attitude[1]. With the rapid increase of World Wide Web, people often express their sentiments over internet through social media, blogs, rating and reviews. Due to this increase in the textual data, there is a need to analyze the concept of expressing sentiments and calculate the insights

* Corresponding author.
E-mail address: abi.tripathy@gmail.com

Peer-review under responsibility of organizing committee of the 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)
doi:10.1016/j.procs.2015.07.523

for exploring business. Business owners and advertising companies often employ sentiment analysis to discover new business strategies and advertising campaign.

Machine leaning algorithms are very often helpful to classify and predict whether a document represents positive or negative sentiment. Machine learning is categorized in two types known as supervised and unsupervised machine learning algorithms. Supervised algorithm uses a labeled dataset where each document of training set is labeled with appropriate sentiment. Whereas, unsupervised learning include unlabeled dataset where text is not labeled with appropriate sentiments[2]. This study mainly concerns with supervised learning techniques on a labeled dataset.

Sentiment analysis is usually implemented on three levels namely *sentence level*, *document level* and *aspect level*[3]. Document Level sentiment classification aims at classifying the entire document or topic as positive or negative. Sentence level sentiment classification considers the polarity of individual sentence of a document whereas aspect level sentiment classification first identifies the different aspects of a corpus and then for each document, the polarity is calculated with respect to obtained aspects.

In this study, an attempt has been made to transform the textual movie reviews to a numerical matrix where each column represents the identified features and each row represents a particular review. The matrix is given as input to machine learning algorithm in order to train the model. This model is then tested and different performance parameters are studied. The results obtained are critically examined on the basis of comparison with existing literature.

The following paper is organized as follows: section 2 presents the literature survey; section 3 describes the detailed methodology of proposed algorithms; section 4 explains the proposed approach; section 5 shows the implementation of proposed approach; section 6 gives a comparison of obtained results with other literatures and finally section 7 concludes the paper along with scope for future work.

2. Related Work

Pang *et.al.* have considered sentiment classification based on categorization aspect with positive and negative sentiments[4]. They have undertaken the experiment with three different machine learning algorithms i.e., Naive Bayes classification, Support Vector machine, and Maximum Entropy classification and are being applied over the n-gram technique.

Turney presents unsupervised algorithm to classify review as either recommended i.e., Thumbs up and not recommended i.e., Thumbs down[5]. The author has used Part of Speech (POS) tagger to identify phrases which contain adjectives or adverbs.

Dave *et.al.* have used structured review for testing and training, identifying features and score methods to determine whether the reviews are positive or negative[6]. They used classifier to classify the sentences obtained from web search through search query using product name as search condition.

Pang and Lee have labeled sentences in the document as subjective or objective[7]. They have applied machine learning classifier to the subjective group which prevents polarity classification from considering useless and misleading data. They have explored extraction of methods on the basis of minimum cut formulation

Whitelaw *et.al.* have presented a sentiment classification technique on the basis of analysis and extraction of appraisal groups[8]. Appraisal group represents a set of attribute values in task independent semantic taxonomies.

Li *et.al.* have proposed various semi-supervised techniques to solve the issue of shortage of labeled data for sentiment classification[9]. They have used under sampling technique to deal with the problem of sentiment classification i.e., imbalance problem.

Wang and Wang have proposed a variance mean based feature filtering method that reduces the feature for representational phrase of text classification[10]. The final performance of the method was observed to be better as it only considered the best feature and also the computation time got decreased as incoming text classified automatically.

3. Methodology

Two approaches of sentiment classification are very often used in literature, which are known as binary sentiment classification and multi-class sentiment classification[11]. In binary sentiment classification each document or review of the corpus is classified into two classes i.e. either as positive or as negative. Whereas, in multi-class sentiment classification, each review can be classified into more than two classes i.e. as strong positive or positive or neutral or negative or strong negative. Generally, the binary classification is useful when two products need to be compared. In this study, implementation is done with respect to binary sentiment classification.

The case study on movie-reviews has been considered and repository of movie reviews is stored in unstructured textual format. This unstructured data need to be converted in to meaningful data in order to apply machine learning algorithms. The processing of unstructured data includes removal of vague information, removal of unnecessary blank spaces. This processed data is converted to numerical vectors where each vector corresponds to a review and entries of each vector represent the presence of feature in that particular review.

The vectorization of textual data to numerical vector is done using following methodologies.

- **CountVectorizer**: Based on the number of occurrences of a feature in the review, a sparse matrix is created[12].
- **Term Frequency -Inverse Document frequency (TF-IDF)**: The TF-IDF score is helpful in balancing the weight between most frequent or general words and less commonly used words. Term frequency calculates the frequency of each token in the review; but this frequency is offset by frequency of that token in the whole corpus[12]. TF-IDF value shows the importance of a token to a document in the corpus.

The supervised machine learning algorithm is applicable where the labeled dataset is available. The dataset used in this study is labeled dataset and each review in the corpus is either labeled as positive or negative. Two different machine learning algorithms considered in this study are as follows:

1. Naive Bayes (NB) Classifier: It is a probabilistic classifier which uses the properties of Bayes theorem assuming the strong independence between the features[13]. One of the advantage of this classifier is that it requires small amount of training data to calculate the parameters for prediction. Instead of calculating the complete covariance matrix, only variance of the feature is computed because of independence of features. For a given textual review 'd' and for a class 'c' (positive, negative), the conditional probability for each class given a review is $P(c|d)$. According to Bayes theorem this quantity can be computed using the following equation:

$$P(c \mid d) = \frac{P(d \mid c) * P(c)}{P(d)}$$

To further compute the term P(d|c), it is decomposed by assuming that *fi*'s are conditionally independent given d's class. This decomposition of P(d|c) is expressed in following equation:

$$P_{NB}(c \mid d) = \frac{P(c)\left(\prod_{i=1}^{m} P(f_i)|c\right)^{n_i(d)}}{P(d)}$$

2. Support Vector Machine (SVM) as a classifier: SVM is a non-probabilistic binary linear classifier[5]. In this study, SVM Model represents each review in vectorized form as a data point in the space. This method is used to analyze the complete vectorized data and the key idea behind the training of model is to find a hyperplane represented by $\vec{w}$. The set of textual data vectors are said to be optimally separated by

hyperplane only when it is separated without error and the distance between closest points of each class and hyperplane is maximum. After training of the model , the testing reviews are mapped in-to same space and predicted to belong to a class based on which side of the hyperplane they fall on.

Let $c_j \in \{1,-1\}$ be the class (positive , negative) for a document d_j, the equation for $\vec{w}$ is given by

$$\vec{w} = \sum_j \alpha_j c_j \vec{d}_j, \alpha_j \geq 0$$

Dual optimization problem gives the values for αj's. All the $\vec{dj}$ such that αj is greater than zero are termed as Support vectors as they are the only document vectors which are contributing to $\vec{w}$.

Confusion matrix is generated to tabulate the performance of any classifier. This matrix shows the relation between correctly and wrongly predicted reviews. In the confusion matrix, TP (True Positive) represents the number of positive movie reviews that are correctly predicted whereas FP (False positive) gives the value for number of positive movie reviews that are predicted as negative by the classifier. Similarly, TN (True Negative) is number of negative reviews correctly predicted and FN (False Negative) is number of negative reviews predicted as positive by the classifier.[14]

Table 1: Confusion Matrix

	Correct Labels	
	Positive	Negative
Positive	True Positive(TP)	False Positive(FP)
Negative	False Negative(FN)	True Negative(TN)

From this confusion matrix, different Performance evaluation parameter like precision, recall, F-measure and accuracy are calculated. The table of confusion matrix formation is shown in table 1.

Precision: It gives the exactness of the classifier. It is the ratio of number of correctly predicted positive reviews to the total number of reviews predicted as positive.

$$precision = \frac{TP}{TP+FP}$$

Recall: It measures the completeness of the classifier. It is the ratio of number of correctly predicted positive reviews to the actual number of positive reviews present in the corpus.

$$Recall = \frac{TP}{TP+FN}$$

F-measure: It is the harmonic mean of precision and recall. F-measure can have best value as 1 and worst value as 0. The formula for calculating F-measure is presented as:

$$F - Measure = \frac{2*Precision*Recall}{Precision+Recall}$$

Accuracy: It is one of the most common performance evaluation parameter and it is calculated as the ratio of number of correctly predicted reviews to the number of total number of reviews present in the corpus. The formula for calculating accuracy is given as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

The dataset considered in this study is the Polarity movie review dataset which consist of 1000 positively labeled and 1000 negative labeled movie reviews[15]. This dataset does not contain separate reviews for training and testing purpose. Therefore, cross validation technique is used which randomly selects the training and testing set.

4. Proposed Approach

Labeled polarity movie dataset has been taken in the consideration which consist of 1000 positive and 1000 negative reviews15. Each movie review first undergoes through a preprocessing step, where all the vague information is removed. From the cleaned dataset, potential features are extracted. These features are words in the documents and they need to be converted to numerical format. The vectorization techniques are used to convert textual data to numerical format. Using vectorization, a matrix is created where each column represents a feature and each row represents an individual review. This matrix is used as input to classification algorithm and cross validation technique is applied to choose the training and testing set for each fold. Step-wise presentation of proposed approach is shown in the block diagram 1.

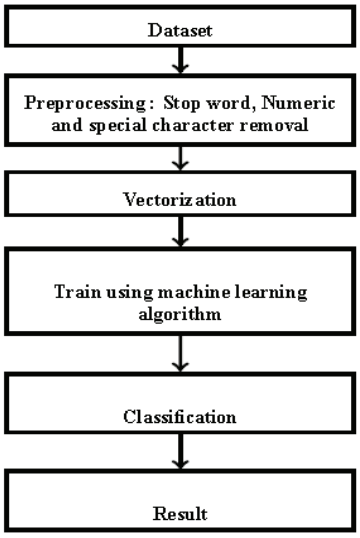

Fig. 1: Diagrammatic view of the proposed approach

4.1 Steps Followed for classification

Step 1. The polarity movie review dataset is considered for analysis which consist of 1000 positive and 1000 negative labeled reviews. For each review a separate text file is maintained.

Step 2. The reviews contain a large amount of vague information which need to be eliminated. In preprocessing step, firstly, all the special characters used like (!@) and the unnecessary blank spaces are removed. It is observed that reviewers often repeat a particular character of a word to give more emphasis to an expression or to make the

review trendy[16]. Words like *wooowwwwwww*, *oohhhhhh* falls in this category. The repetition of characters are also eliminated in this step. Most of the words that do not contribute to any sentiment used in English language are termed as stop words. So, second step in preprocessing involves the removal of all the stop words of English language.

Step 3. After cleaning the dataset in step 3, features can be extracted from it. The features are tokenized word of a review. These words need to be converted to numerical vectors so that each review can be represented in the form of numerical data. The vectorization of features are done using the following two methods.

- **CountVectorizer**: It transforms the review to token count matrix. First, it tokenizes the review and according to number of occurrence of each token, a sparse matrix is created.
  – Calculation of CountVectorizer Matrix: suppose we have three different documents containing following sentences.
  ”Movie is great”.
  ”Movie is Awful”.
  ”Movie is fine”.
  Matrix generated of size 3*5 because we have 3 documents and 5 distinct features. The matrix will look like given in table 2.

Table 2: Matrix Generated Under CountVectorizer Scheme

	Feature 1	Feature 2	Feature 3	Feature 4	Feature 5
Sentence 1	1	1	1	0	0
Sentence 2	1	1	0	1	0
Sentence 3	1	1	0	0	1

  Each 1 in a row corresponds to presence of a feature and 0 represents absence of a feature from particular document.
- **TF-IDF**: Its value represents the importance of a word to a document in a corpus. TF-IDF value is proportional to the frequency of a word in a document.
  – Calculation of TF-IDF value: suppose a movie review contain 100 words wherein the word *Awesome* appears 5 times. The term frequency (i.e., TF) for *Awesome* then (5 / 100) = 0.05. Again, suppose there are 1 million reviews in the corpus and the word *Awesome* appears 1000 times in whole corpus Then, the inverse document frequency (i.e., IDF) is calculated as log(1,000,000 / 1,000) = 3. Thus, the TF-IDF value is calculated as: 0.05 * 3 = 0.15.

Step 4. The numeric vectors can be given as input to the classification algorithm. The different classification algorithm used are as follows:

- Naive Bayes (NB) algorithm: Using probabilistic analysis, features are extracted from numeric vectors. These features help in training of the Naive Bayes classifier model [13].
- Support vector machine (SVM) algorithm: SVM plots all the numeric vectors in space and defines decision boundaries by hyperplanes. This hyperplane separates the vectors in two categories such that, the distance from the of each category to the hyperplane is maximum[5].

Initially, the dataset was not divided between testing and training subsets. So, k-fold cross validation technique is used, the number of folds used are 10.

Step 5. After training of model, confusion matrix is generated which shows the number of positive and negative reviews that are correctly predicted and number of positive and negative reviews that are wrongly predicted. For each fold, prediction accuracy is calculated based on this confusion matrix and final accuracy is given by taking the

mean of all the individual accuracies of 10 folds. However, individual accuracy of a particular fold can be much higher than the mean of all accuracies.

Step 6. For each model, values of precision, recall and F-measure as performance evaluation parameters are found out. The confusion matrix and a table containing performance evaluation parameter is generated. Finally, these obtained results are compared with the values obtained by other authors in literature.

5. Implementation

The implementation of above mentioned algorithms are carried out on Polarity movie review dataset. K-fold cross validation algorithm is implemented where single fold is considered for testing and remaining folds are considered for training. For each algorithm different Performance evaluation parameters and confusion matrix are obtained.

- **Naive Bayes Algorithm**: The confusion matrix obtained after implementation of Naive Bayes classification algorithm is shown in table 3.

Table 3: Confusion matrix for Naive Bayes classifier

	Correct Labels	
	Positive	Negative
Positive	11107	1393
Negative	2384	9666

The performance evaluation parameters obtained for Naive Bayes classifier is shown in table 4.

Table 4: Evaluation parameters for Naive Bayes classifier

	Precision	Recall	F-Measure
Negative	0.80	0.89	0.84
Positive	0.87	0.77	0.82

Maximum accuracy achieved after the cross validation analysis of Naive Bayes classifier is **0.8953**.

- **Support Vector Machine Algorithm**: The confusion matrix obtained after implementation of Support Vector Machine algorithm is shown in table 5.

Table 5: Confusion matrix for Support Vector Machine classifier

	Correct Labels	
	Positive	Negative
Positive	11102	1398
Negative	1688	10812

The performance evaluation parameters obtained for Support Vector Machine classifier is shown in table 6.

Table 6: Evaluation parameters for Support Vector Machine classifier

	Precision	Recall	F-Measure
Negative	0.87	0.89	0.88
Positive	0.89	0.86	0.88

Maximum accuracy achieved after the cross validation analysis of Support Vector Machine classifier is **0.9406**.

6. Comparative Analysis

This section compares the output obtained using the proposed method with the output obtained in other manuscripts. To compare the result, two manuscripts are considered i.e., the manuscript by Pang and Lee[7] and another by Read[17]. Both manuscript used the same polarity dataset with 1000 positive and 1000 negative reviews. The following Table 7 shows the comparison of obtained output with the other literatures on same dataset.

Table 7: Comparison of Proposed work with existing literatures

	Pang and Lee[7]	**Read [17]**	**Proposed Approach**
Naïve Bayes	0.864	0.789	0.895
SVM	0.8615	0.815	0.940

From the table 7, it is found out that the accuracy obtained in present method is better in compare to the accuracy obtained in both manuscript. Pang and Lee in their paper used 10 fold cross validation to perform the classification which is same as present approach where as Read used 3 fold cross validation for classification. It is considered that the higher the no of fold for cross validation, the result is much generalized. Thus, 10 fold cross validation is considered in this case in compare to that of 3 fold cross validation by Read.

7. Conclusion

In this study, an attempt has been made to classify sentiment analysis for movie reviews using machine learning techniques. Two different algorithms namely Naive Bayes (NB) and Support Vector Machine (SVM) are implemented. These two algorithms have also been implemented earlier by different researchers and results of all versions of implementation have been compared. It is observed that SVM classifier outperforms every other classifier in predicting the sentiment of a review.

In this study, only two different classifiers have been implemented. In future, other similar classification strategies under supervised learning methodology like maximum entropy classifier, stochastic gradient classifier, K nearest neighbor and others can be considered to implement and a comparison of results can be presented with SVM classifier.

References

1. S. Argamon, K. Bloom, A. Esuli, and F. Sebastiani, “Automatically determining attitude type and force for sentiment analysis,” in *Human Language Technology. Challenges of the Information Society*. Springer, 2009, pp. 218–231.

2. Y. Singh, P. K. Bhatia, and O. Sangwan, “A review of studies on machine learning techniques,” *International Journal of Computer Science and Security*, vol. 1, no. 1, pp. 70–84, 2007.

3. R. Feldman, “Techniques and applications for sentiment analysis,” *Communications of the ACM*, vol. 56, no. 4, pp. 82–89, 2013.

4. B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 2002, pp. 79–86.
5. P. D. Turney, "Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 417–424.
6. K. Dave, S. Lawrence, and D. M. Pennock, "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews," in *Proceedings of the 12th international conference on World Wide Web*. ACM, 2003, pp. 519–528.
7. B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," in *Proceedings of the 42nd annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2004, p. 271.
8. C. Whitelaw, N. Garg, and S. Argamon, "Using appraisal groups for sentiment analysis," in *Proceedings of the 14th ACM international conference on Information and knowledge management*. ACM, 2005, pp. 625–631.
9. S. Li, Z. Wang, G. Zhou, and S. Y. M. Lee, "Semi-supervised learning for imbalanced sentiment classification," in *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, vol. 22, no. 3, 2011, p. 1826.
10. Y. Wang and X.-J. Wang, "A new approach to feature selection in text classification," in *Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference on*, vol. 6. IEEE, 2005, pp. 3814–3819.
11. H. Tang, S. Tan, and X. Cheng, "A survey on sentiment detection of reviews," *Expert Systems with Applications*, vol. 36, no. 7, pp. 760–773, 2009.
12. R. Garreta and G. Moncecchi, *Learning scikit-learn: Machine Learning in Python*. Packt Publishing Ltd, 2013.
13. A. McCallum, K. Nigam *et al.*, "A comparison of event models for naive bayes text classification," in *AAAI-98 workshop on learning for text categorization*, vol. 752. Citeseer, 1998, pp. 41–48.
14. K. Mouthami, K. N. Devi, and V. M. Bhaskaran, "Sentiment analysis and classification based on textual reviews," in *Information Communication and Embedded Systems (ICICES), 2013 International Conference on*. IEEE, 2013, pp. 271–276.
15. P. dataset. Polarity dataset version 2.0, sentiment anaysis dataset. [Online]. Available: http://www.cs.cornell.edu/people/pabo/movie-review-data/review polarity.tar.gz
16. S. Amir, M. Almeida, B. Martins, J. Filgueiras, and M. J. Silva, "Tugas: Exploiting unlabelled data for twitter sentiment analysis," *SemEval 2014*, p. 673, 2014.
17. J. Read, "Using emoticons to reduce dependency in machine learning techniques for sentiment classification," in *Proceedings of the ACL Student Research Workshop*. Association for Computational Linguistics, 2005, pp. 43–48.