ISSN 2047-3338

Feature Reduction using Principal Component Analysis for Opinion Mining

Jeevanandam Jotheeswaran[1], Loganathan R.[1] and Madhu Sudhanan B.[2]
[1]Department of Computer Science & Engineering, HKBK College of Engineering, Bangalore, India
[2]Anna University of Technology, Coimbatore, India
jeeva.hkbk@gmail.com, logu73@yahoo.com

***Abstract*—Opinions express viewpoints of users, and reviews gives information about how a product is perceived. Online reviews are now popularly used for judging quality of product or service and influence decision making of the users while selecting a product or service. Opinions are increasingly available in form of reviews and feedback at websites, blogs, and microblogs which influences future customers. As it is not feasible to manually handle the huge amount of opinions generated online, Opinion mining uses automatic processes for extracting reviews and discriminate relevant information with sentiment orientation. In this paper, it is proposed to extract the feature set from movie reviews. Inverse document frequency is computed and the feature set is reduced using Principal Component Analysis. The effectiveness of the pre-processing is evaluated using Naive Bayes and Linear Vector Quantization.**

***Index Terms*– Opinion Mining, IMDb, Inverse Document Frequency (IDF), Principal Component Analysis (PCA), Naïve Bayes and Linear Vector Quantization**

I. INTRODUCTION

IN the recent years Sentiment analysis (also called Opinion Mining) has become the key research tool to identify relevant answers for the much complex questions such as "What my customer wants /think?" The entire service industry revolves around the above question, various review methods are used for customer feedback. With the rapid growth of the Internet, feedbacks and review of product has migrated to online forums from word to mouth. Electronic communities (like face book, mouthshut.com and many other online consumer forums) provide a wealth of information. The reviews generated benefit both the users and the business concerns and "who" says "what" and "how" they say it, matters [1].

Microblogging services allow users to share content by posting frequent, short text updates. Of these services, Twitter has been by far the most popular – expanding rapidly from 94K users in April 2007 [5] to over 200 million unique users by August 2011, with over 200 million posts or "tweets" generated per day [2]. Users can track the content generated by other users based on non-reciprocal "follower" relations. Recently, a variety of researchers have considered Twitter as a target for applying sentiment analysis and opinion mining techniques. Pak & Paroubek [7] collected Twitter data for this purpose and trained a Naive Bayes classifier on both n-grams and part-of-speech tags to identify positive and negative tweets. Davidov et al., [8] performed sentiment classification using different types of features, including punctuation, words, and n-grams. Noisy labels for training were selected based on a small number of pre-specified Twitter hash tags and smileys.

The objective of Sentiment analysis is to make sure the ranking of the helpful votes on the reviews, are informative and fitting in to the time frame. There are many research directions [3], e.g., sentiment classification where opinions are classified positive or negative; subjectivity classification deals with the subjective or objective of a sentence and its associated opinion; feature / topic-based sentiment analysis assigns positive or negative sentiments to topics or product features. The sentiment analysis focuses on conveying polarity or strength to opinion expressions for determining the objectivity-subjectivity orientation of a document [4] or the polarity of an opinion sentence in a document [6].

So far the reviews either use the volume of reviews or link structures to predict the trend of product sales [9], [10] without taking into consideration the effect of the sentiments in the reviews. Though there is a strong correlation between the volume of reviews and sales, using the volume or the link structures alone do not provide satisfactory prediction performance [9], [10].

In contrast to previous work, in this paper we describe a system that computes the inverse document frequency (IDF) of words in the movie review and select features using Principal Component Analysis (PCA). The effectiveness of the features thus selected is evaluated using LVQ classifier.

II. RELATED WORK

A. Domain-Driven Data Mining (D3M)

In the past few years, domain-driven data mining has emerged as an important new paradigm for knowledge discovery [11], [12]. Motivated by the significant gap between the academic goals of many current KDD methods and the real-life business goals, D3 advocates the shift from data centered hidden pattern mining to domain-driven Actionable Knowledge Discovery (AKD).

B. Review Mining

With the rapid growth of online reviews, review mining has attracted a great deal of attention. Early work in this area was primarily focused on determining the semantic orientation with the rapid growth of online reviews; review mining has attracted a great deal of attention. Early work in this area was primarily focused on determining the semantic orientation of reviews. Among them, some of the studies attempt to learn a positive/negative classifier at the document level. Pang et al. [6] labeled the polarity of IMDB movie reviews using three machine learning approaches (Naive Bayes, Maximum Entropy, and Support Vector Machine).

Some studies classify the documents at finer level using words for classification. The words are classified into "good" and "bad," group and then the overall "goodness" or "badness" score for the documents are estimated using certain functions. Kamps and Marx [13] evaluated the semantic distance from a word to good/bad with WordNet. Extending previous work on plain two-class classification, reviews were determined using different rating scales such as number of stars [14, 15]. Liu, et al., [16] proposed framework for comparing opinions of competing products based on multiple feature dimensions. Visualization of the strength and weakness of the product was done using "Opinion Observer."

C. Assessing the Review Helpfulness

Compared to sentiment mining, identifying the quality of online reviews has received relatively less attention. A few recent studies along this direction attempt to detect the spam or low-quality posts that exist in online reviews. Jindal and Liu [17] presented a categorization of review spams, and propose some novel strategies to detect different types of spams. Liu et al. [18] proposed a classification-based approach to discriminate the low quality reviews from others, in the hope that such a filtering strategy can be incorporated to enhance the task of opinion summarization.

D. Recommender systems

Recommender systems have emerged as an important solution to the information overload problem where people find it more and more difficult to identify the useful information effectively. Studies in this area can generally be divided into three directions: content-based filtering, Collaborative Filtering (CF), and hybrid systems. Content based recommenders rely on rich content descriptions of behavioral user data to infer their interests, which raises significant engineering challenges as the required domain knowledge may not be readily available or easy to maintain.

As an alternative, collaborative filtering takes the rating data as input, and applies data mining or machine learning algorithms to discover usage patterns that constitute the user models. When a new user comes to the site, his/her activity will be matched against those patterns to find likeminded users and items that could be of interest to the users are recommended.

III. METHODOLOGY

In this paper, online movie reviews is used as data due to its popularity and availability online. The movie reviews are sourced from Internet Movie Database (IMDb), an online database related to movies, television shows, and actors. Bo Pang and Lillian Lee created a benchmark dataset of movie-review documents from the IMDb archives. The dataset is labeled with overall sentiment polarity (positive or negative) or subjective rating (e.g., two stars). This dataset is used to evaluate the proposed method in this research paper. During preprocessing, commonly occurring words which have no relevance to polarity of the document is listed as stop words and words having the same root word is stemmed. A corpus of words from the document is prepared and the importance on each word with respect to the corpus is computed using the inverse document frequency. The feature set dimension is reduced using Principal component analysis and Learning Vector Quantization is used to classify the opinion.

A. Inverse Document Frequency

The documents in the dataset are modeled as vector *v,* for a given set of documents ' and a set of terms', in the ' dimensional space$^{\pi}$. This is a vector space model. When a term 'occurs in the document', the number of occurrence of the term is given by term frequency which is denoted by *freq(x,a)*. The association of a term ' with respect to the given document' is measured by the term-frequency matrix *TF(x, a)*. The term frequencies are assigned values depending on the occurrence of the terms, so *TF(x, a)* is assigned either zero if the document does not contain the term or a number otherwise. The number could be set as *TF(x, a)*= 1when term ' occurs in the document ' or uses the relative term frequency. The relative term frequency is the term frequency versus the total number of occurrences of all the terms in the document. The term frequency is generally normalized by eq. (1):

$$TF(x,a)=\begin{cases}0 & freq(x,a)=0\\ 1+\log\left(1+\log\left(freq(x,a)\right)\right) & otherwise\end{cases} \quad (1)$$

Inverse Document Frequency (IDF) represents the scaling factor. The importance of a term 'is scaled down if term occurs frequently in many documents due to its reduced discriminative power. The *IDF(a)*is defined as follows in eq. (2):

$$IDF(a)=\log\frac{1+|x|}{x_a} \quad (2)$$

x_a is the set of documents containing term *a*.

Similar documents have similar relative term frequencies which are exploited to find similar documents. Similarity measures are used to find similarity among a set of documents or between a document and a query. Cosine measure is generally used to find similarity between documents; the cosine measure is got by eq. (3):

$$sim(v_1,v_2)=\frac{v_1.v_2}{|v_1|\,|v_2|} \quad (3)$$

where v_1 and v_2 are two document vectors, $v_1 . v_2$ defined as

$$\sum_{i=1}^{a} v_{1i} v_{2i}$$ and $$|v_1| = \sqrt{v_1 . v_1}$$.

B. Principal Component Analysis

Principal Components Analysis (PCA) is applied to reduce the dimensions of the inputs when the dimensions of the input are large and the components are highly correlated. PCA determines a smaller set of artificial variables which will represent the variance of a set of observed variable. The artificial variables calculated are called principal components. These principal components are used as predictor or criterion variable in other analysis. The variables are orthogonalized by the PCA and principal components with largest variation are chosen and components with least variation are eliminated from the dataset. The PCA is applied as follows on a set of data.

- A dataset which has a mean of zero is formed by subtracting the mean of the data from each data dimension.
- Covariance matrix is calculated.
- Eigenvectors and Eigenvalues of the covariance matrix are calculated.
- Principal components of the dataset are represented by the highest Eigenvalues and the Eigenvalues of less significance are removed and form a feature vector.
- A new dataset is derived.

C. Learning Vector Quantization

Vector quantization encodes input vectors by finding "representatives" or "code-book vectors" that is an approximation to the original input space. A set of prototype vectors defines the codebook. The input space is divided up into Voronoi tessellation. An input is assigned to a cluster that is its closest prototype. The distance is usually measured by Euclidean distance.

Learning Vector Quantization (LVQ) is a supervised classification algorithm which is based on self organizing maps with input vectors and weights or Voronoi vectors. In LVQ, the input data point with the class information allows known class labels of input to locate best classification label to each Voronoi cell. New inputs are classified on the basis of the Voronoi cell it falls into. The LVQ algorithm during training moves the Voronoi cell boundary for improved classification. The input classes are checked against the Voronoi cell and move the weights accordingly as follows:

1. When input x and Voronoi vector/weight $w_{I(x)}$ have the same class label, then it is moved closer together by $\Delta w_{I(x)}(t) = \beta(t)(x - w_{I(x)}(t))$.
2. When input x and associated Voronoi vector/weight $w_{I(x)}$ have the different class labels, then it is moved apart by $\Delta w_{I(x)}(t) = -\beta(t)(x - w_{I(x)}(t))$.
3. Voronoi vectors/weights W_j corresponding to other input regions remain the same.

where $\beta(t)$ is a learning rate that decreases with the number of iterations / epochs of training.

IV. RESULTS AND DISCUSSIONS

For the experiments, 125 positive and 125 negative movie reviews were used. A corpus of 439 terms was extracted after stop words and stemming. The importance of terms was computed using Inverse document frequency. Principal Component Analysis (PCA) was used to reduce the features. The classification accuracy obtained from LVQ and compared with Naïve Bayes classifier and Classification and Regression Tree (CART) is shown in Fig. 1.

The classification accuracy obtained through LVQ is better than Naïve Bayes by a factor of 5%. Fig. 2 shows the Root Mean Squared Error (RMSE).

The precision & recall for the positive opinion and negative opinion for all the three classifiers is shown in Fig. 3.

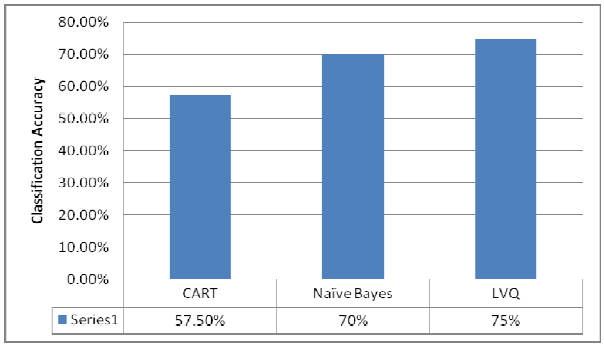

Fig. 1: Classification accuracy obtained

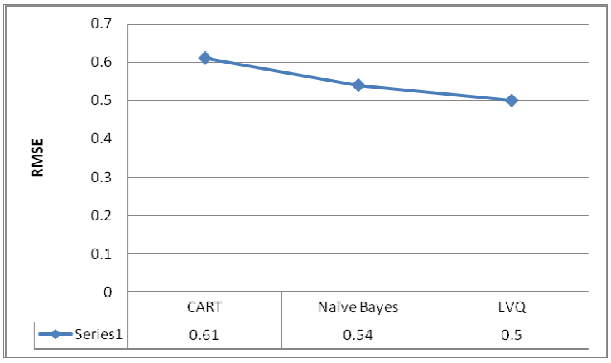

Fig. 2: The root mean squared error for each classifier

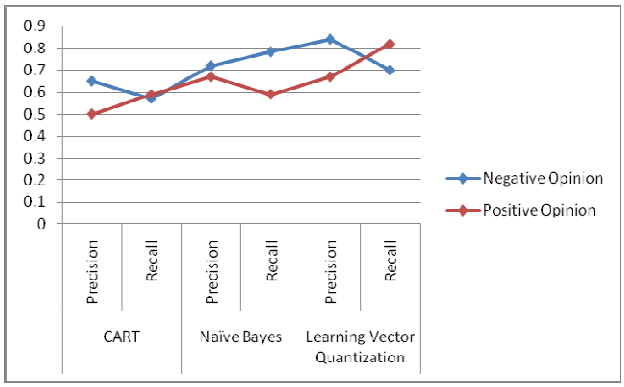

Fig. 3: Precision and recall

From Fig. 3 it can be seen that the recall is low for positive opinions in Naïve Bayes and CART which reduced the classification accuracy. Similarly it is seen that precision for positive opinion is quite low for all the three classifiers.

V. CONCLUSION

In this paper, it is proposed to investigate the classification efficiency of Opinion mining using Learning Vector Quanitzation classifier. IMDb Movie review dataset was used and features was extracted from the review documents using inverse document frequency and the importance of the word computed. Principal component analysis was used for feature selection based on the importance of the work with respect to the entire document. The classification accuracy obtained by LVQ was 75% which is 5% higher than the Naïve Bayes, though it was observed that the precision for positive opinions was quite low. This phenomenon was observed not only on LVQ but other classifiers including CART and Naïve Bayes. Further work needs to be done to improve the classification accuracy of positive opinion.

REFERENCES

[1] Anindya Ghose and Panagiotis G. Ipeirotis on Estimating the Helpfulness and Economic Impact of Product Reviews: Mining Text and Reviewer Characteristics in IEEE transactions on Knowledge and Data engineering, vol. 23, no. 10, October 2011

[2] http://blog.twitter.com/2011/08/your-world-more-connected.html

[3] Ramanathan Narayanan, Bing Liu and Alok Choudhary on Sentiment Analysis of Conditional Sentences in EMNLP,2009

[4] P. Turney, "Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews," *Proc. 40th Ann. Meeting Assoc. for Computational Linguistics* (ACL 02), Assoc. for Computational Linguistics, 2002, pp. 417−424.

[5] A. Java, X. Song, T. Finin, and B. Tseng, "Why we twitter: understanding microblogging usage and communities," in Proc. Joint 9th WEBKDD and 1st SNA-KDD Workshop, 2007, pp. 56–65.

[6] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs Up? Sentiment Classification Using Machine Learning Techniques," *Proc. 2002 Conf. Empirical Methods in Natural Language Processing* (Emnlp 02), Assoc. for Computational Linguistics, 2002, pp. 79–86.

[7] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining," in Proc. 7th Conference on International Language Resources and Evaluation (LREC'10), 2010.

[8] D. Davidov, O. Tsur, and A. Rappoport, "Enhanced sentiment learning using twitter hashtags and smileys," in Proceeding of the 23rd international conference on Computational Linguistics (COLING), 2010.

[9] D. Gruhl, R. Guha, R. Kumar, J. Novak, and A. Tomkins, "The Predictive Power of Online Chatter," Proc. 11th ACM SIGKDD Int'l Conf. Knowledge Discovery in Data Mining (KDD), pp. 78-87, 2005.

[10] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information Diffusion through Blogspace," Proc. 13th Int'l Conf. World Wide Web (WWW), pp. 491-501, 2004.

[11] L. Cao, C. Zhang, Q. Yang, D. Bell, M. Vlachos, B. Taneri, E. Keogh, P.S. Yu, N. Zhong, M.Z. Ashrafi, D. Taniar, E. Dubossarsky, and W. Graco, "Domain-Driven, Actionable Knowledge Discovery," IEEE Intelligent Systems, vol. 22, no. 4, pp. 78-88, July/Aug. 2007.

[12] L. Cao, Y. Zhao, H. Zhang, D. Luo, C. Zhang, and E.K. Park, "Flexible Frameworks for Actionable Knowledge Discovery," IEEE Trans. Knowledge and Data Eng., vol. 22, no. 9, pp. 1299- 1312, Sept. 2009.

[13] J. Kamps and M. Marx, "Words with Attitude," Proc. First Int'l Conf. Global WordNet, pp. 332-341, 2002.

[14] B. Pang and L. Lee, "Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales," Proc. 43rd Ann. Meeting on Assoc. for Computational Linguistics (ACL), pp. 115-124, 2005.

[15] Z. Zhang and B. Varadarajan, "Utility Scoring of Product Reviews," Proc. 15th ACM Int'l Conf. Information and Knowledge Management (CIKM), pp. 51-57, 2006.

[16] B. Liu, M. Hu, and J. Cheng, "Opinion Observer: Analyzing and Comparing Opinions on the Web," Proc. 14th Int'l Conf. World Wide Web (WWW), pp. 342-351, 2005.

[17] N. Jindal and B. Liu, "Opinion Spam and Analysis," Proc. Int'l Conf. Web Search and Web Data Mining (WSDM), pp. 219-230, 2008.

[18] J. Liu, Y. Cao, C.-Y. Lin, Y. Huang, and M. Zhou, "Low-Quality Product Review Detection in Opinion Summarization," Proc. Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP), pp. 334-342, 2007.