Sentiment Analysis Based Mining and Summarizing Using SVM-MapReduce

Jayashri Khairnar, Mayura Kinikar

Department of Computer Engineering, Pune University, MIT Academy of Engineering, Pune.

Abstract— The Sentiment Analysis is the process use to determine the semantic orientation of the reviews. There are many algorithms are exists for the sentiment classification. Support vector machines are a specific type of machine learning algorithm used for many statistical learning problems, such as text classification, spam filtering, face and object recognition, handwriting analysis and countless others. We have studied the SVM as the recent machine learning method for sentiment classification, this method later suppressed by using feature extraction method. We find a way to reduce the size of summary using LSA feature extraction method. In this paper we are extending and investigating the SVM method by addition of the parallel processing methods of sentiment classification such as MapReduce and Hadoop. The practical evaluation of SVM with and without MapReduce as well as LSA is presented in this paper.

Keywords— Sentiment Analysis, Support Vector Machine (SVM), Feature Extraction, Latent Semantic Analysis (LSA), MapReduce, Hadoop.

1 INTRODUCTION

With the evolution of web technology, there is a large amount of data available in the web for the internet users. These users not only use the available resources in the web but also give their suggestions and feedbacks which are much essential to organize and analyze their views for better decision making. In the real world, organizations and businesses always want to find consumer or public opinions about their products and services. Individual consumers also want to know the opinions of existing users of a product before purchasing it and others opinions about political candidates before making a voting decision in a political election. Nowadays, if one wants to buy a consumer product, one is no longer limited to asking one's friends and family for opinions because there are many user reviews and discussions in public forums on the Web about the product. Due to a large collection of opinions on the Web, some form of summary of opinions is needed. Sentiment analysis has grown to be one of the most active research areas in natural language processing. In fact, it has spread from computer science to management sciences and social sciences due to its importance to business and society as a whole. Recently many researchers found that sentiment classification accuracy is mainly affected by decision function used in machine learning methods. We simply used Support vector machine to analyze positive and negative opinions [2].SVM is a useful technique for data classification. Further LSA is used along with SVM in order to improve the performance. The method of LSA is used for features extraction as well as dimensionality reduction with good accuracy of text categorization and less computational overhead [9].

In this paper our main aim is to investigate the algorithm of SVM and improve further its performance by using the Hadoop and MapReduce. Here mainly MapReduce parallel programming model is presented along with SVM; propose a MapReduce and the Hadoop distributed classification method, and presented its practical evaluation.

The rest of the paper is organized as follows. In section II, related work is presented. In section III, parallelized SVM learning algorithm is introduced. In section IV, results and discussion is introduced. In section V, the conclusion is presented.

2 RELATED WORKS

In this section, support vector machine, latent semantic analysis and MapReduce programming model will be introduced briefly.

2.1 Support Vector Machine

Support vector machines were introduced in (Vapnik) and basically attempt to find the best possible surface to separate positive and negative training samples. In this module, a document is composed of sentences and a sentence is composed of terms, it is reasonable to determine the semantic orientation of the text from terms. SVM has been shown to be highly effective in traditional text categorization.SVM measure the complexity of hypothesis based on the margin with which they separate the data instead of the number of features. One remarkable property of SVM is that their ability to learn can be independent of the dimensionality of the feature space. To construct a feature vector of the document stop words are removed first and then each distinct word in the document is used to represent a feature [7]. Support Vector Machines (SVMs) are supervised learning methods used for

Manuscript received April 5, 2015
Manuscript revised April 20, 2015

classification. In this work, SVM is used for sentiment classification. Support vector machines perform sentiment classification task on review data. The kernel function plays a critical role in SVM and its performance. Here use RBF kernel for classification in high dimensional. Radial basis functions (RBF) have received significant attention, most commonly with a Gaussian of the form. LIBSVM is a well-known library for SVM that is developed by Chih-Chung Chang and Chih-Jen Lin [10]. LIBSVM is an integrated software for support vector classification, (C-SVC, nu-SVC), regression (epsilon-SVR, nu-SVR) and distribution estimation (one-class SVM). It supports multi-class classification. Employed SVM to perform the classification and LIBSVM package is used in the system and cross validation is conducted in the experiment.

2.2 Feature Extraction

In this module, LSA is used to find compact description of the data.LSA used filtering approach to further select the content of the summary based on users favor. LSA is a fully automatic mathematical / Statistical technique for extracting and inferring relations of expected contextual usage of words in passages of discourse [9]. Essentially, LSA is a theory and method to analyze relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms. That work was interested in addressing the issues of synonymy (there are many ways to refer to the same idea) and polysemy (most words have more than one distinct meaning). LSA regarded as a kind of extended vector space analysis model which decomposes the Term-Document matrix through the Singular Value Decomposition (SVD).Singular Value Decomposition (SVD) is a method that separates a matrix into three parts; left eigenvectors, singular values, and right eigenvectors. It can be used to decompose data such as images and text. Singular Value Decomposition is a powerful technique for dimensionality reduction. It is a particular realization of the Matrix Factorization approach. LSA applies singular-value decomposition (SVD) to the term-document matrix and a low-rank approximation of the matrix could be used to determine patterns in the relationships between the terms and concepts contained in the text.

2.3 MapReduce Programming

MapReduce programming model was proposed in 2004 by the Google, which is used in processing and generating data sets implementation. This framework solves many problems, such as data distribution, job scheduling, fault tolerance, machine to machine communication, etc. Hadoop Map Reduce is a programming paradigm and software framework which is used for writing applications that rapidly process data in parallel on large clusters of compute nodes [13]. Map Reduce is a programming model for data processing and is used to write programs that run in the Hadoop environment. Combine Hadoop Map Reduce with SVMs to develop a methodology for managing data sets. It is also highly scalable and also improves the accuracy in categorizing [1] [5].

3 PARALLELIZED SVM LEARNING ALGORITHM

In this section, parameters of SVM will be analyzed firstly and then based on the related study, a parallelized SVM learning algorithm based on MapReduce is proposed. For SVM, the selection of kernel function has a significant impact on the performance. RBFSVM, which Gaussian function is taken as a kernel function, shows a strong learning ability and is used in this paper. Performance analysis based on Cross-Validation accuracy where the accuracy rate gives the measure for classification performance.

Cross-Validation

Cross-Validation is used for analyzing the classification performance. In the “leave-one-out" method one item from the training data set is left out and the learning algorithm is trained on the rest of the items. The trained model is then used to predict the label of the one left out earlier. This process is repeated for each item of the training set by leaving it out and predicting its label from the trained model prepared from the rest of the items in the training set. Thus, a k-fold cross-validation was preferred where the training data set is broken into k sets of data, each of size n=k, where n is the size of the training data set. The learning algorithm is trained on k-1 sets and tested against 1. This process is repeated k times after which the mean accuracy is calculated. A small value of k makes the analyses more pessimistic and this helps in selecting the best model. Choosing too small a value for k, for instance, 3-fold is shown to result in wastage of data and more expensive. The accuracy (AC) is the proportion of the total number of predictions that were correct. In this paper, an improved incremental SVM algorithm will be proposed based on parallel method.

The MapReduce frameworks is inspired by map and reduce functions commonly used in functional programming. Users specify a map function that processes a key/value pair to generate a set of intermediate key/value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key. Figure 1 shows the data flow of the various stages of MapReduce. Hadoop is an open source platform based on the MapReduce framework, which can be applied in huge data mining well.

MapReduce programming model, by map and reduce function realize the Mapper and Reducer interfaces. They form the core of task.

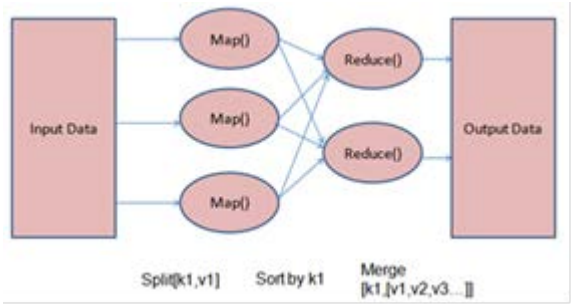

Figure.1 the MapReduce programming model

1. MAPPER

Map function requires the user to handle the input of a pair of key value and produces a group of intermediate key and value pairs. <Key, value> consists of two parts, value stands for the data related to the task, key stands for the "group number" of the value. MapReduce combine the intermediate values with same key and then send them to reduce function.

2. REDUCER

Reduce function is also provided by the user, which handles the intermediate key pairs and the value set relevant to the intermediate key value. Reduce function mergers these values, to get a small set of values. The process is called "merge ". But this is not simple accumulation. There are complex operations in the process. Reducer makes a group of intermediate values set that associated with the same key smaller. In MapReduce framework, the programmer does not need to care about the details of data communication, so <key, value> is the communication interface for the programmer in MapReduce model. <Key, value> can be seen as a "letter", key is the letter's posting address, value is the letter's content. With the same address letters will be delivered to the same place. Programmers only need to set up correctly <key, value>, MapReduce framework can automatically and accurately cluster the values with the same key together. Map tasks and Reduce task is a whole, cannot be separated. They should be used together in the program.

MapReduce algorithm process is described as follows:

Map phase:

Step 1: Hadoop and MapReduce framework produce a map task. Each <Key, Value> corresponds to a map task.

Step 2: Execute Map task, process the input <key, value> to form a new <key, value>. This process is called "divide into groups". That is, make the correlated values correspond to the same key words. Output key value pairs that do not required the same type of the input key value pairs. A given input value pair can be mapped into 0 or more output pairs.

Step 3: Mappers output is sorted to be allocated to each Reducer.

Reduce phase:

Step 4: Shuffle. Input of Reducer is the output of sorted Mapper. In this stage, MapReduce will assign related block for each Reducer.

Step 5: Sort. In this stage, the input of reducer is grouped according to the key (because the output of different mapper may have the same key). The two stages of Shuffle and Sort are synchronized.

4 RESULTS AND DISCUSSION

There are many metrics which we have studied for evaluating the effectiveness of machine learning methods. The most commonly used metrics are precision, recall, F-measure and accuracy. In order to find out this performance metrics we have to do the understanding of if the classification of a document was a true positive (TP), false positive (FP), true negative (TN), or false negative (FN) as showing in below confusion matrix table 1.

Table I. Confusion matrix

		Predicted	
		Positive	Negative
Actual	Positive	A	B
	Negative	C	D

The accuracy (AC) is the proportion of the total number of predictions that were correct. It is determined using the equation:

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

Precision (P) is the proportion of the predicted positive cases that were correct, as calculated using the equation:

$$\text{Precision} = \frac{tp}{tp + fp}$$

The recall or true positive rate (TP) is the proportion of positive cases that were correctly identified, as calculated using the equation:

$$\text{Recall} = \frac{tp}{tp + fn}$$

An F-measure that combines precision and recall is the harmonic mean of precision and recall, the traditional F-measure or balanced F-score:

$$F = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Scenarios

Following performance evaluation graphs are showing the performance of dataset for all methods. This configuration is applied for SVM, SVM-LSA, SVM-MapReduce and SVM-MapReduce-LSA algorithms. In this paper, we collected the movie reviews from Internet Blogs. Since the original data are a hypertext markup language (HTML) document, HTML-tag-removal process is required to extract the text information. Training data are necessary for SVM to train a classification model, and manual classifica-

tion is performed to classify the training reviews into positive or negative reviews. In this performance of the system is analyzed by increasing the dataset from 100 to 400 reviews. We have evaluated our proposed approach with dataset, which is available at http://www.cs.cornell.edu/people/pabo/movie-review-data/polarity_html.zip.

Table II Table for Accuracy performance

Records	100	200	300	400
SVM	0.62	0.62	0.63	0.64
SVM-LSA	0.80	0.72	0.72	0.84
SVM-MapReduce	0.71	0.63	0.68	0.65
SVM-MapReduce-LSA	0.89	0.85	0.90	0.94

Table II show accuracy performance of SVM, SVM-LSA, SVM-MapReduce and SVM-MapReduce-LSA algorithms is evaluated by varying datasets from 100 to 400 records. The Figure 2 show accuracy graph for Algorithms scenario in which SVM-MapReduce-LSA have better accuracy as compare to other algorithms.

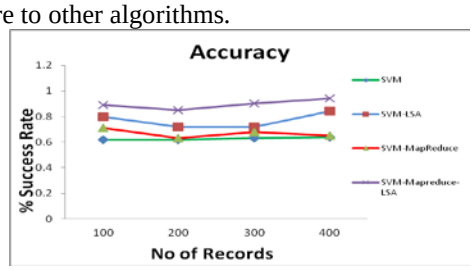

Figure 2.Accuracy curve for movie review dataset

Table III. Table for Time in seconds

Records	100	200	300	400
SVM	12	22	33	43
SVM-LSA	10	7	10	10
SVM-MapReduce	11	8	11	11
SVM-MapReduce-LSA	6	7	10	10

Table III show time performance of SVM, SVM-LSA, SVM-MapReduce and SVM-MapReduce-LSA algorithms is evaluated by varying datasets from 100 to 400 records. The Figure 3 show time curve, in which SVM-MapReduce-LSA take minimum time as compare to other algorithms.

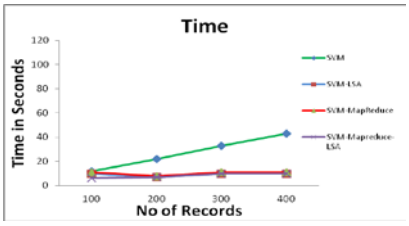

Figure 3.Time curve for movie review dataset

Table IV Table for Precision

Records	100	200	300	400
SVM	0.61	0.61	0.63	0.63
SVM-LSA	0.79	0.71	0.71	0.93
SVM-MapReduce	0.70	0.63	0.65	0.65
SVM-MapReduce-LSA	0.87	0.86	0.89	0.95

Table IV show Precision performance of SVM, SVM-LSA, SVM-MapReduce and SVM-MapReduce-LSA algorithms is evaluated by varying datasets from 100 to 400 records. The Figure 4 show precision graph for Algorithms scenario in which SVM-MapReduce-LSA have better performance as compare to other algorithms.

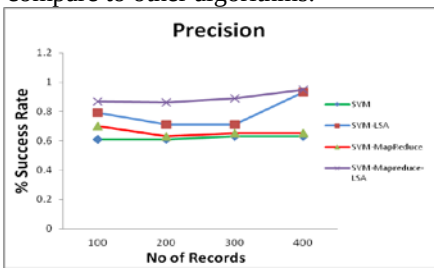

Figure 4.Precision curve for movie review dataset

Table V show Recall performance of SVM, SVM-LSA, SVM-MapReduce and SVM-MapReduce-LSA algorithms is evaluated by varying datasets from 100 to 400 records. The Figure 5 show recall graph for Algorithms scenario in which SVM-MapReduce-LSA have better performance as compare to other algorithms.

Table V Table for Recall

Records	100	200	300	400
SVM	0.65	0.65	0.65	0.65
SVM-LSA	0.83	0.74	0.74	0.74
SVM-MapReduce	0.74	0.72	0.72	0.72
SVM-MapReduce-LSA	0.91	0.83	0.92	0.92

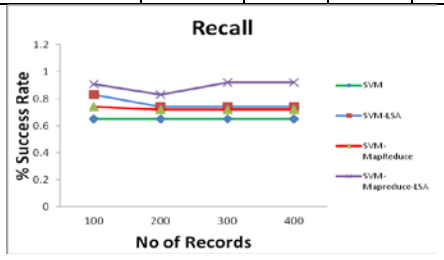

Figure 5.Recall curve for movie review dataset

Table VI. Table for F-Measure

Records	100	200	300	400
SVM	0.62	0.62	0.63	0.63
SVM-LSA	0.80	0.72	0.72	0.82
SVM-MapReduce	0.71	0.67	0.68	0.68
SVM-MapReduce-LSA	0.88	0.84	0.90	0.93

Table VI show F-Measure performance of SVM, SVM-LSA, SVM-MapReduce and SVM-MapReduce-LSA algorithms is evaluated by varying datasets from 100 to 400 records. The Figure 6 show F-Measure graph for Algorithms scenario in which SVM-MapReduce-LSA have better performance as compare to other algorithms.

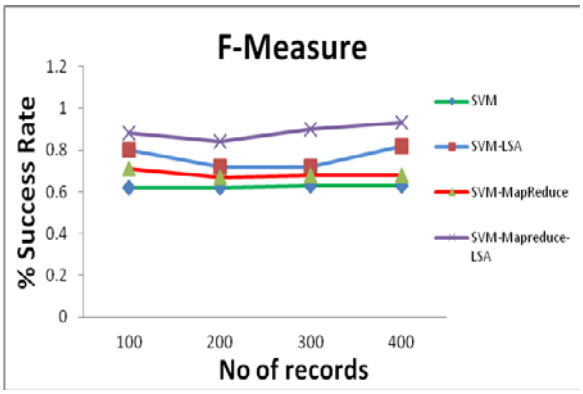

Figure 6.F-Measure curve for movie review dataset

We have observed all the expected results for precision, recall, F-measure, time and accuracy rates. We claim that from above results, our proposed or extended method of sentiment classification is more accurate and efficient as compared to SVM method and hence we will further like to do analysis and investigation over the same.

5. Conclusion

Sentiment classification is applied to the reviews, and summarization is based on sentiment-classification results. In this paper, we have discussed first most commonly used SVM, LSA, and then most recent is MapReduce based approach for sentiment classification. However we found that there is still place for improvement in terms of accuracy and efficiency of SVM method, and hence we have proposed to add the approach of MapReduce and Hadoop together with SVM to improve the accuracy and efficiency of sentiment classification approach. We have presented the programming model for MapReduce as well as Hadoop and how it's included in SVM. The practical results showing that proposed method of sentiment classification resulted into better as compared to existing one and hence we will further like do carry more investigation over the same. In future extend SVD to very large data set that can only be stored in secondary storage and also use more combination of n-grams and feature weighting that gives a better accuracy level.

References

[1] Jun Zhao, Zhu Liang, and Yong Yang," Parallelized Incremental Support Vector Machines Based on MapReduce and Bagging Technique" IEEE International Conference on Information Science and Technology Wuhan, Hubei, China; March 23-25, 2012.

[2] L. Zhuang, F. Jing, and X.-Y. Zhu, "Movie review mining and summarization," in Proc. 15th ACM Int. Conf. Inf. Knowl. Manage. 2006, pp. 43–50.

[3] Y. Lu, C. Zhai, and N. Sundaresan, "Rated aspect summarization of short comments," in Proc. 18th Int. Conf. World Wide Web, New York: ACM, 2009, pp. 131–140.

[4] M. Hu and B. Liu, "Mining and summarizing customer reviews," in Proc.10thACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, 2004, pp.168–177.

[5] Godwin Caruana, Maozhen Li, and Man Qi," A MapReduce based Parallel SVM for Large Scale Spam Filtering" IEEE, Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2011.

[6] Sergio Herrero-Lopez, "Accelerating SVMs by integrating GPUs into MapReduce Clusters" IEEE, 2011.

[7] B.Pang, L.Lee, and S.Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques,"inProc.ACL-02Conf.Empirical Methods Natural Lang. Process., 2002, pp. 79–86.

[8] S. H. Choi, Y.-S. Jeong, and M. K. Jeong, "A hybrid recommendation method with reduced data for large-scale application," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 40, no. 5, pp. 557–566, Sep. 2010.

[9] Chien-Liang Liu, Wen-Hoar Hsaio, Chia-Hoang Lee, Gen-Chi Lu, and Emery Jou, "Movie Rating and Review Summarization in Mobile Environment", IEEE VOL. 42, NO. 3, MAY 2012.

[10] (2001). LIBSVM: A library for support vector machines [Online]. Available: http://www.csie.ntu.edu.tw/cjlin/libsvm.

[11] V. Hatzivassiloglou and K. R. McKeown, "Predicting the semantic orientation of adjectives," in Proc. 8th Conf. Eur. Chap. Assoc. Comput. Linguist, Morristown, NJ: Assoc. Comput. Linguist. 1997, pp. 174–181.

[12] Hadoop. http://hadoop.apache.org/

[13] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In Proceedings of the 6th USENIX Symposium on Operating Systems Design and Implementation, pages 137-149, 2004.