ISSN 2047-3338

M-Learning Sentiment Analysis with Data Mining Techniques

A. Nisha Jebaseeli[1] and Dr. E. Kirubakaran[2]

[1]Department of Computer Science, Bharathidasan University Constituent College, Lalgudi, Trichy-India

[2]Outsourcing Department, Bharat Heavy Electricals Ltd., Trichy-India

[1]nishamarcia@yahoo.com, [2]e_kiru@yahoo.com

***Abstract*– The development of mobile communication and hand-held device offers new and innovative pedagogical approach in the learning system. The M-learning approach offers an anytime, anywhere learning scenario to the learners through mobile devices. Sentiment Analysis is the area where the user's reviews or opinions about the products are analyzed. Analyzing the opinion or sentiment available online gives significant amount of information about the product, through which a person or a company can gauge the product quality and its status in the market. This paper analyzes the sentiments about the M-learning system and also investigates the classification accuracy of Naïve Bayes algorithm. The accuracy of the Naïve Bayes algorithm is compared with random forest data mining algorithm and K nearest neighbor (KNN) algorithm for opinion mining and in specific for M learning systems.**

***Index Terms*– M-Learning, Opinion Mining, Sentiment Analysis, Random Forest, Naive Bayes and K Nearest Neighbor**

I. INTRODUCTION

THE development of communication technology has led to easy access of information through the internet. Nowadays, the use of mobile devices is increasing rapidly which in turn has popularized the pedagogical methods such as learning through mobile devices. Several mobile learning systems are available and also the user opinions about these systems are aired in the social blogs or review websites. To analyze the user reviews and classify into positive, negative and neutral is a tedious and also time-consuming task. Sentiment analysis is a latest research area which analyzes the sentiments or opinions of users of the product. This analysis will help the companies to improve the quality and also gain insight about user's opinion of their product.

Online review sites and blogs are some of the major source of opinions expressed by the people which are gathered using information retrieval technologies to find sentiments about products. The main goal of Opinion mining is to determine the polarity of comments. The comments are classified as positive, negative or neutral by extracting attributes and features of the object that have been commented on in each document [7], [8]. In the context of world, facts and opinion are the two main categories of textual information. Objective statements are facts of entities and events. Subjective statements are opinions that replicate the people's opinions or sentiments about event and entities. Sentiment analysis seeks to identify the viewpoint or opinion expressed in the text document; using information retrieval and computational linguistics. The opinion expressed on the topic is given significance rather than the topic itself [4], [6]. Sentiment analysis or opinion mining analyzes to extract the subjective information in source materials by applying natural language processing, computational linguistics and text analytics. User's opinion of product or anything is the fundamental part of information-gathering behavior. In Natural language processing, sentiment classification is broadly studied. For example, given a set of evaluative documents A, it determines whether each document A expresses a positive, negative or neutral opinion (or sentiment) on an object. For example, given a set of M-learning reviews, the system classifies them into positive reviews, negative reviews and neutral reviews. This is similar to a supervised classification method.

In this paper, we implement the sentiment analysis in M-learning system. This task helps to enhance the mobile learning system and also know about the user opinions of the M-learning system. M-learning is not just e-learning that is facilitated by mobile technology, but also the practice of coming to know through conversations and explorations across multiple contexts [5]. With the help of M-learning system, the learners are able to learn any subject, at any time, any place and just when the knowledge is required.

In this paper we investigate classification of opinion mining particularly of M-learning system not only based on opinion words but also corpus words which are frequently used in the documents under review. We also propose a methodology to eliminate words that are commonly used in the dataset under study. For example the word "learning" is irrelevant for classification of m learning reviews. We rank the corpus using Singular value decomposition and prepare our data for opinion mining. This paper is organized into the following sections. Section II briefly describes the materials and methods and classification algorithms, section III describes the results obtained and discusses the same.

II. MATERIALS AND METHODS

The proposed method uses M-learning system reviews as data due to the availability of a large number of reviews

online in (www.market.android.com). Only learner's opinions of free M-learning system are considered under study. From this 100 positive, 100 negative and 100 neutral reviews are selected. Three data mining tools are used to perform the sentiment classification.

The data is preprocessed using statistical tools. All the 300 reviews retrieved from the blogs are stored as text document. Then the documents are loaded in the Statistica text mining tool for preprocessing. In this step, a table is constructed using all words found in the input documents, which includes indexing and counting of documents and words, i.e., a matrix of frequencies of words that specifies the number of times that word occurs in each document. This initial process is further refined by the use of stop word list and stemming of words; to exclude certain common words such as "the" and "a" (stop word lists) and to combine different grammatical forms of the same words such as "traveling," "traveled," "travel," etc.

Singular value decomposition (SVD) is used to find the importance of the word. The main idea behind the SVD is taking the high dimensional, highly variable set of data points and reducing it into a lower dimensional space of the input matrix that expose the substructure of the original data. Test data is created by scoring of the words. The word frequency and the weightage given to the words by SVD are used for scoring process.

Before training the classification algorithm, some manual preprocessing methods are done in the data. The word count less than 15 and greater than 85 are deleted from the data list in the Excel sheet obtained from the Statistica. After this filtering process, the word importance is multiplied by the data in the excel sheet. The manual preprocessing is for strengthening the importance of the data.

The classification algorithms are trained using the numerical data obtained from the above process. The matrix of values obtained from Statistica tool can be loaded in the weka tool. With the help of weka tool the classification accuracy of three machine learning algorithm were obtained. Finally, the accuracy of the three algorithms is compared.

A. Classification Algorithms

The following classification algorithms are used in the weka tool.

1) Random Forest

The Random forest algorithm was developed by Leo Breiman and Adele Cutler which is used for classification. Random forest with generation of 25 trees was investigated. Random forest algorithm for classification uses an ensemble of classification trees [1]. A bootstrap sample of the data and a random subset of variables are used to build the classification trees. Thus, the trees built are not as rational as decision tree. The random forest integrates multiple unstable classifiers, thus improving the performance of the final classifier giving better overall performance than an individual classifier [2].

Each tree is got from the random vector values sampled independently with same distribution giving rise to a combination of tree predictors to form Random forests. As the number of trees increases the generalization error for forests converges to a limit; depending on the strength of individual trees and correlation among the trees [3]. Random vectors created help the growth of each tree in the forest. If δ_k is the random vector generated for the kth tree, which is independent of other random vectors $(\delta_1,....,\delta_{k-1})$; the tree is grown using the random vector δ_k and the training set. The classifier resulting from the vector is $h(x,\delta_k)$ where x is the input vector. So a random tree consists of set of trees with classifiers got from independent identically distributed random vectors; and each tree casts a vote for the class at input x. as the number of trees increase, the generalization error converges to

$$P_{X,Y}\left(P_\delta\left(h(X,\delta)=Y\right)-\max_{j\neq Y}P_\delta\left(h(X,\delta)=j\right)<0\right) \qquad (1)$$

Where Y, X are random vectors and $P_{X,Y}$ is generalization error probability over the X, Y space.

Lower the correlation, higher the accuracy of the random forest. The correlation is minimized by the randomness used while maintaining the strength. Each node is built into tree using randomly selected inputs. Random forest is an effective tool in prediction.

Table I: Summary of Results

	Naïve Bayes	**KNN**	**Random Forest**
Correctly Classified Instances	165	158	180
Incorrectly Classified Instances	135	142	120
Kappa statistic	0.325	0.29	0.4
Mean absolute error	0.3153	0.3156	0.3119
Root mean squared error	0.4751	0.5617	0.4218
Relative absolute error	70.94%	71%	70.18%
Root relative squared error	100.78%	119.16%	89.49%
Total Number of Instances	300	300	300

2) K Nearest Neighbor

KNN is a simple machine learning algorithm. In this algorithm, the objects are classified based on the majority of its neighbor. The class assigned to the object is most common among its k nearest neighbors. The KNN classification algorithm classifies the instances or objects based on their similarities to instances in the training data [10]. In KNN, selection is based on majority voting or distance weighted voting.

KNN is unsupervised text classification algorithm and its work efficiently when the training set is large. Consider the vector A and set of M labeled instances $\{ai, bi\}_1^M$. The classifier predicts the class label of A on the predefined N classes. The KNN classification algorithm finds the k nearest neighbors of A and determines the class label of A using

majority vote [9]. KNN classifier applies Euclidean distances as the distance metric.

$$Dist(X,Y) = \sqrt{\sum_{i=1}^{D}(Xi - Yi)^2} \qquad (2)$$

3) Naïve Bayes

It is a supervised learning method and statistical method for classification [11]. A Bayesian classifier is a simplest probabilistic classifier based on Bayes theorem. In text classification, Bayes rule is used to determine the class or group a document falls into by determine the most probable class or group [12]. Word Frequency is used to train the classifier. The classifier maps from a discrete or continuous feature space X to a discrete set of labels Y.

Consider given a set of variables, X={x_1,x_2, . . . x_n}, constructing the posterior probability from a set of possible outcomes Y={y_1,y_2, . . .,y_n}.Using Bayes rule

$$P(Y_i \mid x_1, x_2, ..., x_n) = p(x_1, x_2, ..., x_n \mid C_i)\, p(C_i) \qquad (3)$$

Since Naive Bayes assumes that the conditional probabilities of the independent variables are independent we can decompose to:

$$p(X \mid Y_j) = \prod_{k=1}^{n} p(x_k \mid Y_j) \qquad (4)$$

The posterior can be rewritten as:

$$p(Y_j \mid X) = p(Y_j) \prod_{k=1}^{n} p(x_k \mid C_j) \qquad (5)$$

Based on the above Bayes' rule we label a new case X with a class level Y_j that achieves the highest posterior probability.

III. RESULT AND DISCUSSION

The 80% of the opinions processed in the dataset were used for training and the remaining for testing. The summary of results and classification accuracy obtained is tabulated in Table I and II. Fig. 1 shows the graph of classification accuracy.

Table II: Classification Accuracy

Technique used	Classification Accuracy %
Naïve Bayes	55
KNN	52.67
Random Forest	**60**

Table III: The precision and recall for various algorithms

Technique used	Precision	Recall	F-Measure
Naïve Bayes	0.561	0.55	0.549
KNN	0.529	0.527	0.527
Random Forest	0.601	0.6	0.6

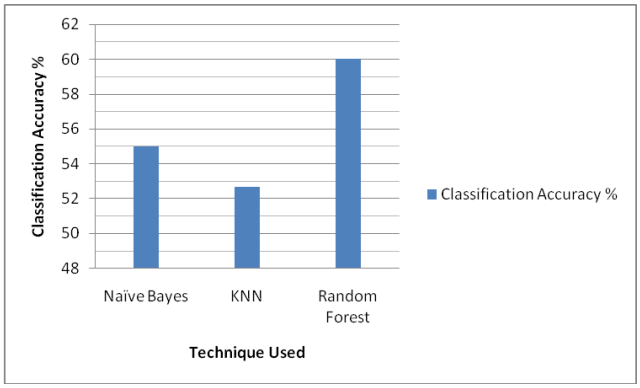

Fig. I: Classification accuracy of various classification techniques

Table III lists the precision and recall for various classification techniques. Fig. 2 and Fig. 3 show the graph for precision, recall and F Measure respectively.

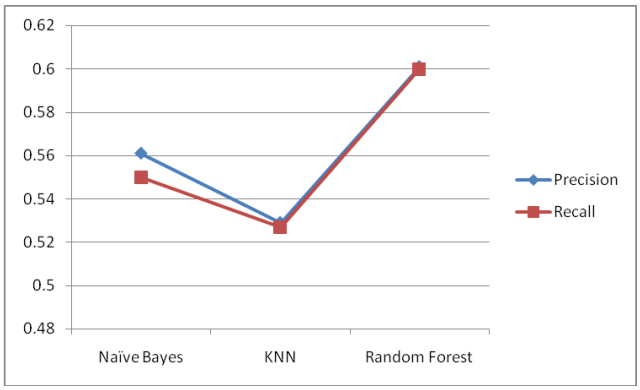

Fig. 1: Precision and Recall of various classification techniques

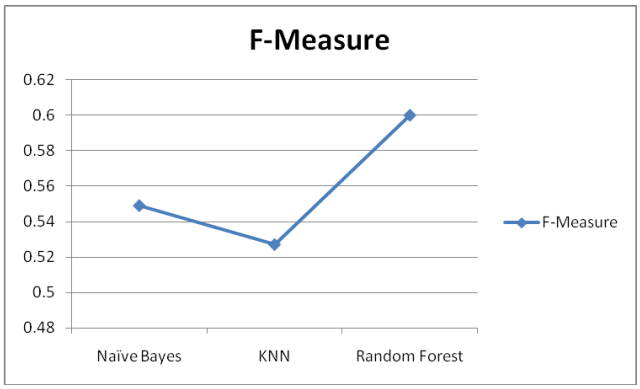

Fig. 3: F Measure

It is observed that the precision and recall for both the classes of opinion that is positive and negative does not vary much. However it is to be noted that both precision and recall values are above average and cannot be claimed as excellent. However the obtained results can be used to cluster large amounts of data for further study.

IV. CONCLUSION

In this paper, it was proposed to investigate the efficiency of Naïve Bayes, k Nearest Neighbor and Random forest classifier to predict opinions as positive or negative specifically for M learning systems. 300 opinions were obtained of which 100 opinions were positive, 100 opinions were negative and 100 were neutral. The data was preprocessed by removing commonly occurring words and rarely occurring words. SVD was used to rate the importance of the words. The obtained data was used as the input for the random forest algorithm using differing number of trees. Accuracies in the range of 55% to 60% were obtained. Further work needs to be done to improve the classification accuracy.

REFERENCES

[1] L. Breiman, "Random forests," *Mach. Learning*, vol. 45, pp. 5–32, 2001

[2] Y. Saeys, T. Abeel, and Y. Van de Peer, "Robust feature selection using ensemble feature selection techniques," in *Proc. ECML/PKDD, Part II (LNAI 5212)*, 2008, pp. 313–325.

[3] Freund, Y. and Schapire, R. [1996] Experiments with a new boosting algorithm, Machine Learning: Proceedings of the Thirteenth International Conference, pp. 148-156

[4] Peter D. Turney and Michael L. Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems, 21(4):315–346.

[5] Sharples. M, Lonsdale. P, Meek. J, Rudman. P. D, & Vavoula. G. 2007. "An Evaluation of MyArtSpace: a Mobile learning Service for school Museum trips". In Proceedings of mLearn 2007, Melbourne, Australia.

[6] Bing Liu . Exploring User Opinions in Recommender Systems. Proceeding of the second KDD workshop on Large Scale Recommender Systems and the Netflix Prize Competition, Aug 24, 2008, Las Vegas, Nevada, USA.

[7] Dave, D., Lawrence, A., and Pennock, D. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. Proceedings of International World Wide Web Conference (WWW'03), 2003.

[8] Turney, P. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. ACL'02, 2002.

[9] Yang Song, Jian Huang, Ding Zhou, Hongyuan Zha, and C. Lee Giles, **"**IKNN: Informative K-Nearest Neighbor Pattern Classification", PKDD 2007, LNAI 4702, pp. 248–264, 2007.

[10] Christof Monz, "Machine Learning for Data Mining", Queen Mary University of London.

[11] Chai, K.; H. T. Hn, H. L. Chieu; "Bayesian Online Classifiers for Text Classification and Filtering",Proceedings of the 25th annual international ACM SIGIR conference on Research andDevelopment in Information Retrieval, August 2002, pp 97-104.

[12] DATA MINING Concepts and Techniques,Jiawei Han, Micheline Kamber Morgan Kaufman Publishers, 2003.

A. Nisha Jebaseeli received her M.Tech degree in Bharathidasan University, Trichy, India in 2009. Now she is working as a Assistant Professor with Department of Computer Science, Bharathidasan University Constituent College, Lalgudi, Trichy, India and also she is pursuing PhD (Computer Science) in Bharathidasan University.

Dr. E. Kirubakaran obtained his B.E (Honours.) degree in Mechanical Engineering from Regional Engineering College Trichy in the year 1978 and joined in Bharat Heavy Electricals Ltd. Tiruchirappalli in 1979. He obtained M.E. in Computer Science from Regional Engineering College in 1984. In 1999, he obtained his Ph.D. degree in Computer Science from Bharathidasan University through Regional Engineering College. In 2005, he got his M.B.A. degree from IGNOU.

Presently he is Additional General Manager, Outsourcing Department, at BHEL, Trichy For more than 31 years he has been working in designing, Developing and maintenance of Software Systems at BHEL, Trichy.