Improving Spam Mail Filtering Using Classification Algorithms With Partition Membership Filter

C. Neelavathi[1], Dr. S. M. Jagatheesan[2]

[1]*Research Scholar, Department of Computer Science, Gobi Arts & Science College, T.N., India*

[2] *Associate Professor, Department of Computer Science, Gobi Arts & Science College, T.N., India*

Abstract : E-mail is one of the most popular and frequently used ways of communication due to its worldwide accessibility, relatively fast message transfer and low sending cost. The different classification algorithms (JRip, Filtered Classifier, K-star, SGD, Multinomial) which are used for classify the email as spam or not. However these algorithms has number of drawbacks such that lack of useful and relevant features that can distinguish between spam and non-spam email increase data dimensionality that decreases accuracy. To overcome these problems, Random Tree algorithm is used. In the proposed algorithm, Random Tree classifier generates the best outcome in terms of accuracy, kappa statistics and less error rate.

KeyWords : *Classification, Random Tree, Spam mail, Accuracy, Error rate*

1. INTRODUCTION

Now a day, emails have become a common and important communication for most internet users. Spam[2] or unwanted electronic mail has become a major problem for organizations and private users. Data mining[8] is primarily used today by companies with a strong consumer focus retail, financial, communication, and marketing organizations. For experiment, spam base dataset[3] can be experimented from UCI repository. Preprocessing is done and different classification methods are compared based on their performance. Rules are generated based on test options. Classification[6] is a data mining (machine learning) technique which has a set of predefined classes and determine in which class a new object belongs to it. There are large numbers of classifiers available which are used to classify the data such as bayes, function, rule, lazy, meta, decision tree etc. Data preprocessing is a data mining technique that involves transforming raw data into an understandable format. For preprocessing partition membership filter is used.

2. CLASSIFICATION

Classification[5] is to analyze the input data and to develop an accurate description or model for each class using the features present in the data. Classification is much more accurate for mapping classes. The set of possible classes is known in advance. The different classification algorithms (JRip, Filtered Classifier, K-star, SGD, Multinomial) which are used for classify the email as spam or not. The Random Tree classifier generates the best outcome in terms of accuracy and error. Classification results are compared based on following category:

Error rate - Error rate of a classifier was defined as the percentage of the dataset incorrectly classified by the method. It is the probability of misclassification of a classifier.

Accuracy - Accuracy of a classifier was defined as the percentage of the dataset correctly classified by the method. The accuracy of all the classifiers used for classifying spam dataset.

Recall - Recall of the classifier was defined as the percentage of errors correctly predicted out of all the errors that actually occurred.

Precision - Precision of the classifier was defined as the percentage of the actual errors among all the encounters that were classified as errors.

Time taken - It can measure the running time of the classifier or algorithm.

Recall= TP / [TP + FN]

Precision= TP / [TP + FP]

Accuracy= [TP + TN] / [TP + FN + TN + FP]

TP (True Positive): The spam which is correctly detected as the spam.

FP (False Positive): The ham email which is predicted as the spam by mistake.

TN (True Negative): The ham email which is correctly predicted as the ham email.

FN (False Negative): The spam email which is predicted as the ham mail by mistake.

***Filter*:**

Filters are used to preprocess the data to remove the noisy data in data mining. There are two types of filters supervised and unsupervised filter. Filters can be applied to both training and test dataset. In Supervised Partition Membership, filter converts numeric values to nominal and distributes into selected number of bins equally.

3. ADVANTAGES AND LIMITATIONS OF CLASSIFICATION ALGORITHMS

3.1 Random Tree :

Advantages: Runs efficiently on large data bases. Handles thousands of input variables without variable deletion. Maintains accuracy when a large proportion of the data are missing. Provides methods for balancing error in unbalanced data sets.

Limitations: Correlations among attributes are ignored.

3.2 JRip:

Advantages: It builds models that can be interpreted easily. Can make use of both categorical and continuous values. It can handle noisy data.

Limitations: In case of a small training set, the JRip algorithm does not work very well(less accurate/efficient).

3.3 Filtered classifier:

Advantages: It is robust with regards to the search space. Classifier can be updated online and that to at very little cost given the fact that new instances with known classes are presented.

Limitations: Expensive testing of each instance. This is problematic for datasets with a large number of attributes.

3.4 K-star:

Advantages: The benefits are that it provides a consistent approach to handling of real valued attributes, symbolic attributes and missing values. Entropic distance is then used to retrieve the most similar instances from the data set.

Limitations: It has long training time. Difficult to understand the learned function (weights).

3.5 SGD :

Advantages: Efficiency. Ease of implementation (lots of opportunities for code tuning).

Limitations: SGD requires a number of hyper parameters such as the regularization parameter and the number of iterations.

SGD is sensitive to feature scaling.

3.6 Multinomial Naivebayes :

Advantages: It has great computational efficiency and classification performance. It gives accurate results for most of the classification and prediction problems.

Limitations: The precision of algorithm decreases if the amount of data is less.

Algorithm of Random tree :

An impurity function is a function defined on the set of all K-tuples of numbers (p1, ..., pK) satisfying pj = 0, j = 1, ...,K,$\sum jPj = 1$ with the properties:

1. Ø is a maximum only at the point (1K, 1K, ..., 1K).
2. Ø achieves its minimum only at the points (1, 0, ..., 0),(0, 1, 0, ..., 0), ..., (0, 0, ..., 0, 1).
3. Ø is a symmetric function of p1, ..., pK, i.e., if permute pj , remains constant.

Given an impurity function, define the impurity measure i(t) of a node t as,

i(t) = Ø(p(1 | t), p(2 | t), ..., p(K | t)) , (1)

where p(j | t) is the estimated probability of class j within node t. Goodness of a split s for node t, denoted by Ø(s, t), is defined by,

Ø(s, t) = Øi(s, t) = i(t) − pRi(tR) − pLi(tL) , (2)

Where pR and pL are the proportions of the samples in node t that go to the right node tR and the left node tL respectively. Define I (t) = i(t)p(t), that is, the impurity function of node t weighted by the estimated proportion of data that go to node t. The impurity of tree T, I (T) is defined by,

$I(T) = \sum_{t\in T} I(t) = \sum_{t\in T} i(t)p(t)$ (3)

Note for any node t the following equations hold:

p(tL) + p(tR) = p(t)

pL = p(tL)/p(t), pR = p(tR)/p(t)

pL + pR = 1

Define

ΔI (s, t) = I (t) − I (tL) − I (tR)

= p(t)i(t) − p(tL)i(tL) − p(tR)i(tR)

= p(t)(i(t) − pLi(tL) − pRi(tR))

= p(t)Δi(s, t)

Possible impurity function:

1.Entropy: $\sum_{j=1}^{k} p_{j \log \frac{1}{pj}}$. If $p_{j=0}$ use the limit $\lim_{pj\to 0} p_{j \log pj=0}$

2.Misclassification rate: 1 − maxjpj .

3.Gini index: $\sum_{j=1}^{k} pj\ (1 - pj) = 1 - \sum_{j=1}^{k} pj$

4. EXPERIMENTAL RESULTS

The experiment can be conducted in weka tool. weka is a machine learning tool to analyze various datasets in data mining. Accuracy can be calculated from formula given as follows:

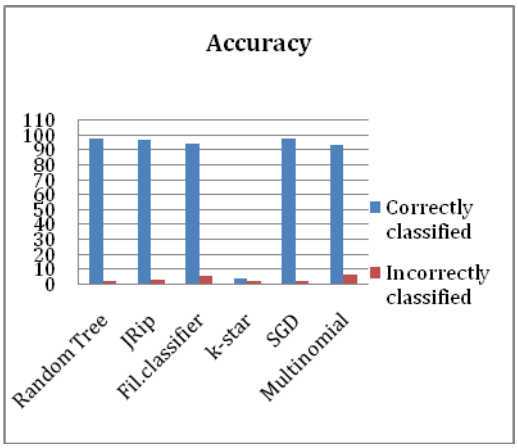

Figure 1 : Accuracy Rate

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

The Figure 1 shows the accuracy rate of existing classification[10] techniques JRip, filtered Classifier, K-star, SGD, Multinomial and proposed technique of Random Tree classifier. Random Tree is an effective algorithm for estimating missing data and maintains accuracy when a large proportion of dataset. Here x axis denotes existing & proposed algorithm and y axis denotes accuracy in %. The accuracy of proposed algorithm[1] is increased than existing algorithms.

***Error rate*:**

Error rate of a classifier was defined as the percentage of the dataset incorrectly classified by the method. It is the probability of misclassification of a classifier. Error rate can be calculated from formula given as follows:

$$\text{Error rate} = \frac{FP + FN}{TP + TN + FP + FN}$$

Random Tree[9] algorithm which gives accuracy rate of 97.08 %, kappa statistics 0.938, Time Taken to build model 0.56 sec which establishes best result than other algorithms. JRip algorithm gives accuracy rate of 96.47%, kappa statistics 0.926, Time Taken to build model 33.62 sec. Filtered Classifier gives accuracy rate of 93.87%, kappa statistics 0.873, Time Taken to build model 2.75sec. K-Star algorithm gives accuracy rate of 97.04 %,

kappa statistics 0.937, Time Taken to build model 0.7 sec. The figure 2 shows the error rate of algorithms.

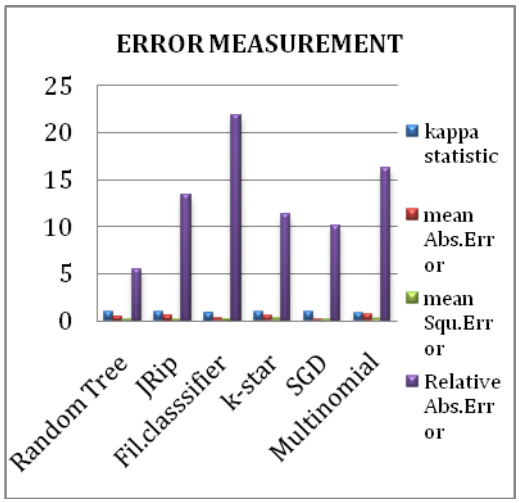

Figure 2 : Error Rate

5. CONCLUSION

Spam mails are becoming a very serious problem for the networks and the productivity of the users. Through this work, the objective which was to analyze the six selected classification algorithms based on Weka and various spam filtering techniques. The result shows the best classifier algorithm is Random Tree classifier for UCI Spambase dataset and performance of each of these six(JRip, Filtered classifier, K-Star, SGD, Multinomial, Random Tree) algorithms can be improved if the dataset is preprocessed using Partition Membership Filter. Among the spam filtering techniques described random Tree generates the best spam mail filtering results in terms of more accuracy and less false positive rate. The future work will involve the combination of the any two specified algorithms to enhance the accuracy so that the spam mail can become more accurate in case of weakly identified.

6. REFERENCES

[1]. Bernhard P fahringer, "Random model trees: an effective and scalable regression method" University of Waikato, New Zealand, http://www.cs.waikato.ac.nz/bernhard.

[2]. Chirag Nathwani, Viralkumar Prajapati, Deven Agravat "Comparative Study Of Web Spam Detection Using Data Mining", International Journal of Computer Applications (0975 – 8887), Volume 68, No.18, April 2013, pp- 26-29.

[3].http://archive.ics.uci.edu/ml/datasets/ Spambase.-UCI Machine Learning Repository.

[4]. Ian H. Witten, Eibe Frank & Mark A. Hall., "Data Mining Practical Machine Learning Tools And Techniques, Third Edition." Morgan Kaufmann Publishers.

[5]. K. Wisaeng , "A Comparison Of Different Classification Techniques For Bank Direct Marketing", International Journal of Soft Computing and Engineering (IJSCE), Volume-3, Issue-4, September 2013, pp-116-119.

[6]. Mr. Hiren Gadhvi, Ms. Madhu Shukla "Comparative Study Of Classification Algorithms For Web Spam Detection", International Journal of Engineering Research & Technology (IJERT), Volume 2, Issue 12, December 2013, pp- 2497-2501.

[7]. P. Yasodha, N.R. Ananthanarayanan, "Comparative Study Of Diabetic Patient Data's Using Classification Algorithm In WEKA Tool", International Journal of Computer Applications Technology and Research, Volume 3, Issue 9, pp- 554 – 558.

[8]. S. Syed Shajahaan, S. Shanthi, V. ManoChitra, "Application Of Data Mining Techniques To Model Breast Cancer Data", International Journal of Emerging Technology and Advanced Engineering, Volume 3, Issue 11, November 2013, pp- 362-369.

[9]. Sneh Lata Pundir, Amrita "Feature Selection Using Random Forest In Intrusion Detection System" International Journal of Advances in Engineering & Technology, July 2013, Volume 6, Issue 3, pp. 1319-1324.

[10]. X. Li and N. Ye, "A Supervised Clustering And Classification Algorithm For Mining Data With Mixed Variables," IEEE Transactions on Systems, Man, and Cybernetics Part A, Volume 36, no. 2, pp. 396–406, 2006.