ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.11

Email Spam Filtering using Classifiers in Data Mining

P.Priyatharsini[1], Dr. C.Chandrasekar[2]
M.Phil Scholar[1], Assistant Professor[2]
Department of Computer Science
Government Arts College, Udumalpet, India

Abstract:
E-mails are the most nontrivial means of communication in the recent years. Spam mails often cause inconvenient to the users. The mails are classified as Spam and ham. Unwanted mails are called as spam and genuine mails are called as ham. In this paper, the effective decision tree classifiers are used to classify whether the mail is spam or ham. Many filtering techniques are used to find the spam mails and filter them but the accuracy and performance of the algorithms is distinct from each other. Efficient filtering of spam mails is an important requirement in using the existing data mining algorithms. In this paper, six decision tree algorithms that are basically used as classifiers namely J48 or C4.5, Rndtree, BFtree, REPtree, LMT and simple CART are compared. These algorithms were studied, analyzed and test results are shown in WEKA tool for efficient spam filtering. The results are compared and RndTree algorithm shows almost 99% accuracy level in filtering the spam mails and this shows best results among other classifiers.

Index terms: classifiers, E-mail, Ham, Spam

I. INTRODUCTION

Spam is an unwanted usually commercial email sent to a large number of recipients. In internet spam has become an electronic thorn in the foot of the ubiquitous systems user. Spam can take away resources from users and service suppliers without compensation. [1]. Spammers collect e-mail addresses from group chats, websites, customer lists, newsgroups, and viruses which harvest users' address books, and are sold to other spammers. Their content varies from deal to real estate to pornography. Since, the cost of the spam is borne mostly by the recipient, many individual and business people send bulk messages in the form of spam. In recent years, spam emails lands up into a serious security threat, and act as a prime medium for phishing of sensitive information. Addition to this, it also spread malicious software to various users. Therefore, email classification becomes an important research area to automatically classify original emails from spam emails. Spam email are Extreme interesting problem for individuals and organizations because it is liable to misuse. Automatic email spam classification [4] contains more challenges because of unstructured information, more number of features and large number of documents. As the usage increases, all of these features may adversely affect performance in terms of quality and speed. Many recent algorithms use only relevant features for classification. Even though more number of classification techniques has been developed for spam classification, still 100% accuracy of predicting the spam email is questionable. So, identification of best spam algorithm itself became a Boring task because of features and drawbacks of every algorithm against each other.[2]. In this paper, spam dataset from UCI machine learning repository [3] is taken as input data for analyzing the various classification techniques using WEKA [5] data mining tool. In this work, feature selection is done first to select the related features for classification. After feature selection, six classification algorithms are taken for evaluation. In this evaluation process, different features are considered for choosing best spam filtering algorithm. The last, performance evaluation is done to analyze the various classification algorithms to select the best classifier for spam emails.

II. RELATED WORKS

Email spam is one of the major problems of the today's Internet, bringing financial damage to companies and annoying individual users. Among the approaches developed to stop spam, filtering is the one of the most important technique. Spam mail, also called unsolicited bulk e-mail or junk mail that is sent to a group of recipients who have not requested it. The task of spam filtering is to rule out unsolicited e-mails automatically from a user's mail stream. These unsolicited mails have already caused many problems such as filling mailboxes, engulfing important personal mail, wasting network bandwidth, consuming users' time and energy to sort through it, not to mention all the other problems associated with spam [1].Developments in the field of spam filtering uses Machine Learning algorithms. Machine learning algorithms are described as either 'supervised' or 'unsupervised'. The distinction is drawn from how the learner classifies data. In supervised algorithms, the classes are predetermined. These classes can be conceived of as a finite set, previously arrived at by a human. In practice, a certain segment of data will be labeled with these classifications. The machine learner's task is to search for patterns and construct mathematical models. These models then are evaluated on the basis of their predictive capacity in relation to measures of variance in the data itself. Unsupervised learners are not provided with classifications. In fact, the basic task of unsupervised learning is to develop classification labels automatically. Unsupervised algorithms seek out similarity between pieces of data in order to determine whether they can be characterized as forming a group. These groups are termed clusters.[6].

What is a Spam Filter?

A spam filter is a program that is used to detect unsolicited and unwanted email and prevent those messages from getting to a user's inbox. Like other types of filtering programs, a spam filter looks for certain criteria on which it bases judgments. For example, the simplest and earliest versions (such as the one available with Microsoft's Hotmail) can be set to watch for particular words in the subject line of messages and to exclude

these from the user's inbox. This method is not especially effective, too often omitting perfectly legitimate messages (these are called *false positives*) and letting actual spam through. More sophisticated programs, such as Bayesian filters or other heuristic filters, attempt to identify spam through suspicious word patterns or word frequency.[7].

III. METHODLOGIES

Decision tree learning is a method commonly used in data mining. Decision Tree is a tree-structured plan of a set of attributes to test in order to predict the output. The example of decision tree is shown in Fig 1.This is the widely used learning method and it can be represented as If – Then rules.

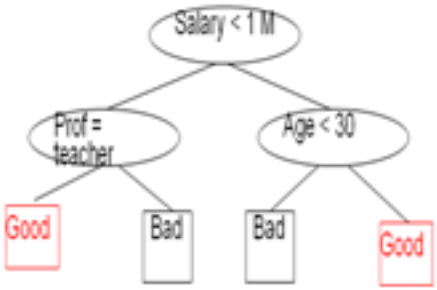

Figure.1. Decision Tree

In this paper, various decision tree classifiers are taken for evaluation and apart from other types of data mining classifiers are emphasized specifically on decision tree classifiers for the particular application of spam filtration technique. This is done because of decision tree filters are easy to implement and easy to understand. It provides an overall satisfactory performance as far as spam mail detection is concerned. The goal is to create a decision tree model and train the model so that it can predicts the value of a target variable based on several input variables. Each interior node corresponds to one of the input variables. There are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf.

REP Tree

[8][9]In the early 1984 Addison- Wesley, Judeu pearl Reduces Error Pruning (REP) Tree Classifier is a fast decision tree learning algorithm and is based on the principle of computing the information gain with entropy and minimizing the error arising from variance[10]. This algorithm is first recommended in. REP Tree applies regression tree logic and generates multiple trees in altered iterations. Afterwards it picks best one from all spawned trees. This algorithm constructs the regression/decision tree using variance and information gain. Also, this algorithm prunes the tree using reduced-error pruning with back fitting method. At the beginning of the missing values by splitting the corresponding instances into pieces.

SIMPLE CART ALGORITHM

It was developed by Leo Breiman in the early 1980's. CART is a recursive and gradual refinement algorithm of building a decision tree. Predict the classification situation of new samples of known input variable value, only need to trace back downwards the decision tree model, compare the threshold value of new sample and the node variable at every node, and select appropriate branches until leaf nodes are reached.

Step 1: All rows in a dataset are assigned to the root node.

Step 2: Each of the predictor variables is split at all its possible split points based on their values for the rows in the node considered.

Step 3: For each split point, the parent node is split into two child nodes by separating the rows with values lower than or equal to the split point and values higher than the split point for the considered predictor variable. For categorical predictor variables, each category of the variable will be considered in turn.

Step 4: The predictor variable and split point with the highest value of

I formula is given below

$$I(s/t)=2P_L\ P_R\sum^{m}_{j=1}|P(C_j|t_L)- P(C_j|t_R)$$

P_L & P_R are the probabilities of a sample to lie in left sub tree and right sub tree respectively and $P(C_j|t_L)$ or $P(C_j|t_R)$ are the probabilities that a sample is in the class C_j and in the left sub tree or right sub tree.

BF Tree

[11]In BF tree learners the "best" node is expanded first as compared to standard DT learners such as C4.5 and CART which expand nodes in depth-first order [12]. The "best" node is the node whose spilt leads to maximum reduction of impurity (e.g.Gini index or information gain) among all nodes available for splitting. The resulting tree will be the same when fully grown; just the order in which it is built is different. BF tree constructs binary trees, i.e., each internal node has exactly two outgoing edges. The tree growing method attempts to maximize within-node homogeneity. The extent to which a node does not represent a homogenous subset of cases is an indication of impurity. For examples, a terminal node in which all cases have the same value for the dependent variable is a homogenous node that requires no further splitting because it is "pure". The impurity measures for nominal dependent variables are entropy-based definition of information gain and gini index.

C4.5/J48 Decision Tree Algorithm

[13]C4.5 is an algorithm used to generate a decision tree developed by Ross Quinlan. In early 1993 Morgan Kaufmann C4.5 is an extension of Quinlan's earlier ID3 algorithm. J48 is an open source Java implementation of the C4.5 algorithm in the weka data mining tool. Time Sleuth extends C4.5 use to temporal and causal discovery [14].The decision tree generated by C4.5 can be used for various classification problems. At each node of the tree the algorithm chooses an attribute that can further split the samples in subsets. Every leaf node represents a classification or decision. Some premises guide this algorithm, such as the following. C4.5 algorithm can easily handle missing values. As missing attribute values are not utilized in gain calculation by C4.5.

LOGISTIC MODEL TREE INDUCTION (LMT)

A Logistic Model Tree is an algorithm for supervised learning tasks which is combined with linear logistic regression and tree induction [15]. Logistic Model Tree creates a model tree with a standard decision tree structure with logistic regression functions at leaf nodes. Logistic Model Tree, leaves have a associated logic regression functions instead of just class labels. [16]

The steps used in LMT algorithm are given below:

Step 1: Growing Initial Tree the initial linear regression model is built for root node using Log it Boost algorithm for whole

dataset Log it Boost is run on the dataset for a fixed number of iterations.

Step 2: Splitting and stopping Splitting criterion used in LMT algorithm is same as that used in C4.5 algorithm. After splitting the dataset, logistic regression models are then built at the child nodes on the corresponding subsets of dataset using Logic Boost algorithm. The initial weights and probability estimates are taken from the parent node. Splitting and model building continues until at least 15 samples are present at node and a useful split is found.

Step 3: Tree pruning The CART algorithm is used for pruning of tree. CART pruning method uses a combination of training error and penalty term for model complexity to make pruning decisions.

RANDOM FOREST ALGORITHM (Rnd Tree)

The random decision forest was first proposed by ho in 1995 Random Forest are ensemble of un pruned binary decision trees, unlike other decision tree classifiers Random Forest grows multiple trees are creates a forest like classification. [17] Algorithm can be used for classification and regression.

Steps in Random Forest Algorithm:

Step 1: A random seed is chosen which pulls out a random collection of samples from training data set while maintaining the class distribution.[18]

Step 2: Selected dataset, a random set of attributes from the original data set is chosen based on user defined values. All the input variables are not considered because of enormous computation and high changes of over fitting.

Step 3: A dataset M is the total number of input attributes in the dataset, only R attributes are chosen at random for each tree R<M.

Step 5: The attributes from this set creates the test possible split using the gini index to develop a decision tree model. The process repeats for each of the branches until the termination condition stating that leaves are the nodes that are too small to split.

Step 6: Random Forest Tree follows the same methodology and constructs multiple trees for the forest using different set of attributes. Used a part of the training data set to calculate model error rate by an inbuilt error estimate.

SPAM DATASET

The spam dataset was taken from UCI machine learning repository and was created by Mark Hopkins, Erik Reeber, George Forman, Jaap Suermondt. Hewlett-Packard Labs. This dataset contains 4601 instances and 58 attributes (57 continuous input attribute and 1 nominal class label target attribute.[3].The class label has two values. 0- for not spam and 1-spam.

IV. EXPERIMENTAL RESULTS

Today, most of the data in the real world are incomplete containing aggregate, noisy and missing values. As the quality decision depends on quality mining which is based on quality data, pre-processing becomes a very important tasks to be done before performing any mining process. Major tasks in data pre-processing are performing feature reduction techniques. The feature reduction techniques used here are the ReliefF, Chi Square Attributeeval, CF subset evaluation methods. The Component Analysis is a dimension reduction technique which enables to visualize a dataset in a lower dimension without the loss of information. ReliefF algorithm detects conditional dependencies' between attributes and provides a unified view on the attribute estimation in regression and classification. It is more robust and can deal with incomplete and noisy data. It evaluates the worth of an attribute by computing the value of chi-squared statistic with respect to class. The dataset is evaluated with 10-fold cross validations in the training data set. The various algorithms before filtering and after filtering is analyzed using the tables that are given below.

Table.1. Details showing the performance of the classifiers in ReliefF filtering method

Algorithms	Test time (in sec)	Correctly classified instances (out of 4601 instances)	Accuracy (in %)	False positive (in %)
REPTree	0.25	4355	94.5	0.92
Simple cart	0.16	4431	97.00	0.69
BFTree	0.22	4455	97.5	0.46
C4.5/J48	0.25	4471	98.15	0.45
LMT	0.5	4534	99.5	0.34
RndTree	**0.26**	**4598**	**99.99**	**0**

Table.2. Details showing the performance of the classifiers in ChiSquare Attribute leval method

Algorithms	Test time (in sec)	Correctly classified instances (out of 4601 instances)	Accuracy (in %)	False positive (in %)
REP Tree	0.26	4357	95.00	0.99
Simple CART	0.15	4431	95.30	0.72
BF Tree	0.26	4451	96.00	0.45
C4.5/J48	0.16	4477	98.00	0.46
LMT	0.94	4534	97.03	0.26
Rnd Tree	**0.24**	**4598**	**99.04**	**0**

Table.3. Details showing the performance of the classifiers before using the filtering methods

Algorithms	Test time (in sec)	Correctly classified instances (out of 4601 instances)	Accuracy (in %)	False positive (in %)
REP Tree	0.8	4274	94.58	1.68
Simple CART	8.33	4253	93.67	1.23
BF Tree	9.57	4267	92.25	1.73
C4.5/J48	2.0	4278	93.05	1.96
LMT	771.35	4262	94.32	1.73
Rnd Tree	**0.24**	**4184**	**91.45**	**2.53**

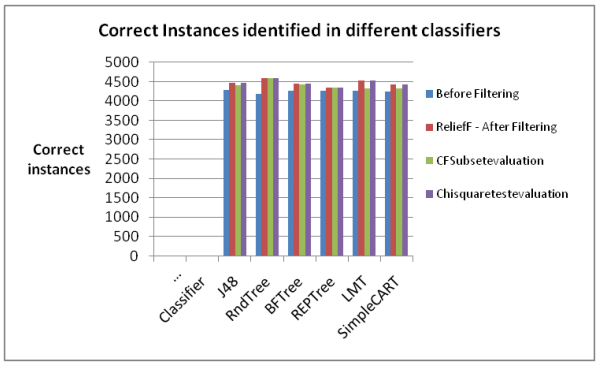

Figure.3. Results of correctly classified instances before and after using WEKA filters

Accuracy

Accuracy of a classifier was defined as the percentage of the dataset correctly classified by the method. This is given in Fig.4.

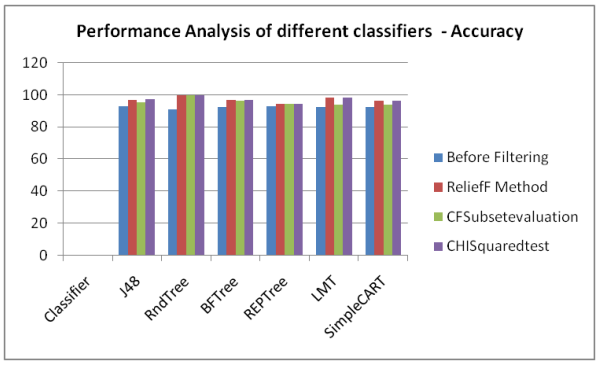

Figure.4. Results showing the accuracy of the classifier before and after using WEKA filters.

Accuracy is more than 96% (approx) in ReliefF and chisquare evaluation method than in CFsubset evaluation methods. The accuracy is 90% before filters are applied to the classifiers. The accuracy of the above algorithms is compared with each other before filtering and after filtering. ReliefF, CFsubsete valuation method and Chisquare attribute valuation methods are used for spam filtering techniques for feature selection. ReliefF filtering and Chi-square evaluation methods produce more correct instances than the CFSubsete valuation method. ReliefF filtering and Chi squared attribute evaluation yields best results for the two classifiers Rnd Tree algorithm and LMT algorithm.

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

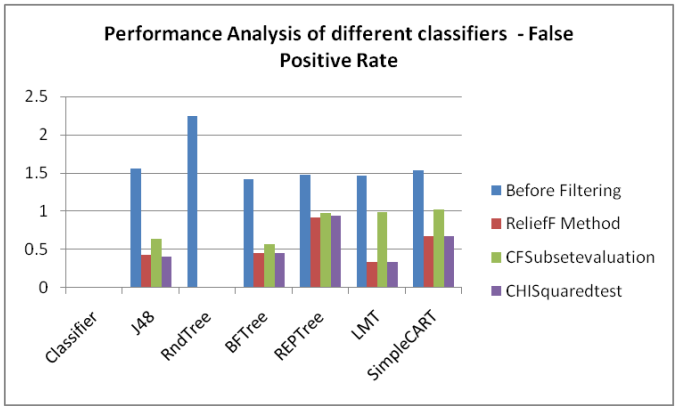

Figure.5. Results of the classifiers in predicting the False positive rate

The False Positive Rate for the above algorithms is specified and Rnd Tree algorithm showed the best in yielding 0% false positive rate. The accuracy of the above algorithm seems to be the best classifier among the other algorithms. The LMT algorithm showed 0.34% false positive rate in the ReliefF method and Chisquared test filters. The LMT algorithm has a great disadvantage with respect to the time taken to execute the test data set. The 10-fold cross-validation is applied to the data set but the LMT algorithm works for hours together to produce the results.

Time

The Time taken for the algorithms to execute are tabulated and summarized in the fig 6.The LMT algorithm takes more time to execute than other algorithms. This algorithm has a great disadvantage in completing time. The 10-fold cross validations are done for each algorithm and LMT takes more than an hour to execute (771.35sec) in the training data set. The details of the time taken to execute the algorithms are stated

V.CONCLUSION

E-mail spam classification needs more attention to identify the major threats and reduce the unwanted information from the spammers. Many researchers have been going on to identify the best classifier in spam filtering. Among all the decision tree classifiers compared in this paper, the execution time, accuracy and low false positive rate has been exhibited only in Rndtree classifier. The accuracy of RndTree is 99% than the LMT classifier with an average of 98% and with the false positive of 0.34% in Chisquare and ReliefF filtering Techniques. The RndTree Classifier shows best performance than other decision tree classifiers.

VI. REFERENCES

[1]. Androutsopoulos, I.; Koutsias, J.; Chandrinos, K.Paliouras, G and Spyropoulos, C. 2000. *An evaluation of naive bayesian anti-spam filtering.*

[2].R.Kishore Kumar, G.Poonkuhali, P.Sudhakar, "Comparitive study on E-mail Spam Classifier Using data Mining Techniques" Proceedings of the International Multi Conference of Engineers and Computer Scientists 2012 Vol 1. March 14-16, 2012 HongKong.

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

[4].Patrick Ozer, "Data Mining Algorithms for classification, Radboud University Nijmegen, Jan 2008.

[5]. Weka. WEKA (Data Mining Software)Available at:http:// www.cs.waikato.ac.nz/ml/weka/.2006

[6].http://monkpublic.library.illinois.edu/monkmiddleware/public/analytics/clusterclassification.html

[7].http://searchmidmarketsecurity.techtarget.com/definition/spam-filter

[8]. Judea pearl, Heuristics, Addison- Wesley, 1984

[9]. Mansour, Y(1997) " Pessimistic Decision tree pruning based on tree size" proc.14th International Conference on Machine learning 195-201.

[10]. S.K. Jayanthi and S. Sasikala. 2013. REPTree Classifier for identifying Link Spam in Web Search Engines. IJSC, Volume 3, Issue 2, (JAN 2013), 498-505.

[11]. The Education Innovator, V.S Department of education Newsletter (September 25, 2005), accessed 10 April 2008

[12]. Shi, Haijian. 2007, Best-First Decision Tree learning. Masters Degrees Theses. University of Waikato Masters Theses

[13]. Quinlan, J.R. C4.5; Programs for machine learning Morgan Kaufmann publisher, 1993

[14]. Ian H. witten; eibe Frank; Mark A.Hall(2011) "Data Mining; Practical machine learning tools and techniques, 3rd Edition", Morgan Kaufmann, San Francisco.p.191.

[15]. Niels Landwehr Istitute for computer science, University of Freiburg, Germany. Landwehr@informatik.uni-freiburg.de.

[16]. Joao Gama, "Function Tree" Machine learning, PP.219-250, 2004

[17]. Ho, tin kam (1995). Random Decision Forest (PDF) Procedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, Q C, 14-16 Augest 1995, PD 278-282.

[18]. Leo Breiman. 2001. Random Forest. Machine Learning.