|| Volume 1 ||Issue 4 ||July 2016||ISSN (Online) 2456-0774

INTERNATIONAL JOURNAL OF ADVANCE SCIENTIFIC RESEARCH AND ENGINEERING TRENDS

An Evaluation on the Efficiency of Hybrid Feature Selection in Spam Email Classification

Ms.Nida Mirza[1], **Ms.Tabinda Mirza**[2]

Student, Computer Science & Engineering, Everest College of Engineering & Technology, Aurangabad, India[1]

Student, Computer Science & Engineering, JNEC, Aurangabad, India[2]

***Abstract*— Spam has serious negative on the usability of email and network resources. Spam is flooding the internet with many copies of the same message, in an attempt to force the message on people who would not otherwise choose to receive it. And despite the evolution of anti spam software, such as spam filters and spam blockers, the negative effects of spam are still being felt by individuals and businesses alike. To prevent this advance techniques are necessary. Our proposed method divides e-mails in spam class and non spam class according to different attribute values of spam. an alternative approach using a neural network (NN) classifier brained on a corpus of e-mail messages from several users. The features selection used in this work is one of the major improvements.**

Keywords:-Spam, E-mail classification, Machine learning algorithms ,Emails classification, Document similarity , Document classification, Feature extraction, Subject classification, Content classification.

I INTRODUCTION

Lately unwanted commercial e-mail also known as spam, has become a huge issue over the internet. Spam is unwanted stuff that can be considered as waste of resources, storage area and usable bandwidth. Spammers these days are aware of several tricky methods to overcome the filtering properties of anti spam systems like using random sender addresses and/or append random values to the beginning or the end of the message subject line [11]. In recent studies and research, more than 50% of all emails are spam which accounts to more than 15 billion emails per day and it also adds up to the cost of internet users around $355 million per annum. Automatic e-mail filtering appears to be the most effective solution to encounter spam. Only several years ago most of the spam could be reliably dealt with by blocking e-mails coming from certain addresses or filtering out messages with certain subject lines. Machine learning is a popular approach used in e-mail filtering. Another approach is knowledge engineering, where a set of rules are specified according to which emails are categorized as spam or ham. A set of such rules should be created either by the user of the filter, or by some other authority. This method has a drawback as it does not promises any fruitful results because the rules must be constantly updated and maintained, which is a waste of time and it is not convenient for most users. Instead machine learning approach uses, a set of training samples, these samples is a set of pre classified e-mail messages. Machine learning approach is more efficient than knowledge engineering approach; it does not require specifying any rules [4]. A specific algorithm is used that helps the machine to learn classification rules from these e-mail messages.

Machine learning algorithms include Naïve Bayes, support vector machines, Neural Networks, K-nearest neighbor, etc. Machine learning approach has been widely studied and there are lots of algorithms can be used in e-mail filtering.

In this aspect, an email spam-based classifier is not only expected to accurately classify spam emails as spams, but also expected to classify non-spam emails as non-spam or ham. This is since both are considered conditions for evaluating the quality of its classification or prediction.

Four prediction metrics are used then to evaluate the quality of email prediction. True Positive (TP) indicates that the spam detection tool predicts that the email is spam and truly it was a spam. True Negative (TN) indicates that the tool or the email system predicts that the email is normal and not spam and correctly it was so. False Positive (FP) indicates that by mistake the tool predicts that a good email is spam (aka false alarms). Last, False Negative (FN) indicates also another mistake where it is predicted that a spam email is normal. As such, a perfect detection system should have the values: TP 100%, TN 100%, FP 0%, and FN 0%. In reality such perfect situation is impossible and impractical. TP and FP complement each other for 100% (i.e. their total should be 100%). Same thing is applied for TN and FN.

In addition to spam based classification, papers that conducted research in emails discussed other aspects such as: Automatic subject or folder classification, priority based filtering of email messages, emails and contacts clustering, etc. Some papers evaluated replies in emails to classify emails on different threads. Currently some email servers such as Gmail combine email together if they came as a reply.

The rest of the paper is organized as the following: Section two presents several research papers in email analysis. Section three presents goals and approaches. Section four presents experiment and analysis and paper is concluded with conclusion section.

II LITERATURE SURVEY

The present work defines the construction of a system which supports content-based message filtering, depending on Machine Learning techniques. Proposed system has relationships with the state of the art in content-based filtering, and with the field of policy-based personalization and, generally in email contents.

A.Spam–non-spam email classification :

Sculley and Wachman (2007) discussed also algorithms such as VSM for email, blogs, and web and link spam detection. The content of the email or the web page is analyzed using different natural language processing approaches such as: Bags of words, NGram, etc. The impact of a tradeoff parameter in VSM is evaluated using different setting values for such parameter. Results showed that VSM performance and prediction accuracy is high when the value of this parameter is high.

Zhuang et al.'s (2008) paper focused on trying to find Botnets. Botnets are groups responsible for spreading spam emails. Methods are evaluated to detect such sources of spam campaigns that share some common features. Spammers however try to change spam emails through some intended mistakes or obfuscations especially in popular filtered keywords. Certain finger prints are defined where all emails that have those finger prints are then clustered together.

Zhou et al. (2010) proposed a spam-based classification scheme of three categories. In addition to typical spam and not spam categories, a third undetermined category is provided to give more flexibility to the prediction algorithm. Undecided emails must be re-examined and collect further information to be able then to judge whether they are spam or not. Authors used Sculley and Cormack, 2008 and UCI Machine Learning Repository, as their experimental email dataset (machine learning repository).

Pérez-Díaz et al.'s (2012) paper 2012 evaluates applying rough set on spam detection with different rule execution schemes to find the best matching one. UCI Spam base is used in the experimental study (machine learning repository).

B. Support Vector Machines:

In this section, support vector machine is applied to the dataset.

Table 1.10-fold cross validation error of SVM with different kernel functions on dataset

Kernel Function	Overall Error %	Spams Caught (SC) %	Blocked Hams (BH) %
Linear	1.18	93.8	0.47
Degree-2 Polynomial	2.03	85.7	0.27
Degree-3 Polynomial	1.64	89.7	0.40
Degree-4 Polynomial	1.70	90.5	0.60
Radial Basis Function	2.61	81.4	0.32
Sigmoid	13.4	0	0

Table 1 shows the 10-fold cross validation results of SVM with different kernels applied to the dataset with extracted features. As it is shown in the table, linear kernel gains better performance compared to other mappings. Using the polynomial kernel and increasing the degree of the polynomial from two to three shows improvement in error rates, however the error rate does not improve when the degree is increased further. Finally, applying the sigmoid kernel results in all messages being classified as hams.

The learning curve for SVM with linear kernel validated using cross validation is shown in figure 3. From this figure, there is a meaningful distance between accuracy of trained model on training set and test set. While the overall training set error of the model is far less than error rate for naive Bayes, the test set error is well above that rate. This characteristic shows the model might be suffering from high variance or over fitting on the data. One option we can explore in this case is reducing the number of features. However, the simulation results show degradation in performance after this reduction. For instance, choosing 800 best features based on MI with the labels and training SVM with linear kernel on the result yields to 1.53% overall error, 91.5% SC, and 0.53% BH.

While applying SVM with different kernels increases the complexity of the model and subsequently the running time of training the model on data, the results show no benefit compared to the multinomial naive Bayes algorithm in terms of accuracy.

The efficiency of a learning method does play an important role in the decision of which technique to select. The most important aspect of efficiency is the computational complexity of the algorithm, even though storage necessities can also turn into a problem as many user profiles have to be maintained. Neural networks and genetic algorithms are much limited in speed as compared to other learning methods as several iterations are needed to determine whether or not a document is relevant [4]. Instance based methods slow down performance as more training cases turn out to be accessible because each and every example has to be analyzed in contrast to all the unseen documents. However, such systems do not offer a filtering strategy level with help of which user can develop the result of the classification process to elect how and to which level filtering process is carried out to remove unnecessary and useless information.

A novel distributed data mining approach, called Symbiotic Data Mining (SDM) [7] that unifies Content Based Filtering (CBF) with Collaborative Filtering (CF) is described. The goal is to reuse local filters from distinct entities in order to improve personalized filtering while maintaining privacy. In paper [26] the effectiveness of email classifiers based on the feed forward back propagation neural network and Bayesian classifiers are evaluated. Results are evaluated using accuracy and sensitivity metrics. The results show that the feed forward back propagation network algorithm classifier provides relatively high accuracy and sensitivity that makes it competitive to the best known classifiers. A fully Bayesian approach to soft clustering and classification using mixed membership models based on the assumptions on four levels: population, subject, latent variable, and sampling scheme was implemented in [8]. In paper [1]-[3], automatic anti spam filtering becomes an important member of an emerging family of junk-filtering tools for the Internet, which will include tools to remove advertisements. The author separate distance measures for numeric and nominal variables, and are then combined into an overall distance measure.

III SYSTEM ARCHITECTURE OF PROPOSED SYSTEM

A. System Architecture:

In this section, a summary of tasks followed in this paper to utilize a personal large content of emails for emails' data mining is described.

1.Data collection stage:

A dataset of 130 test messages , 350 train messages 130 test spam messages, 350 general mails and a set of ham keywords and spam keywords.

2. Emails parsing and pre-processing: A Java MIME parser is then used to parse information from those emails to generate a dataset that include one record for each email with the following information parsed: Email file name, email body, from, subject, and sending date.

3. Emails' dataset data mining.

A tool is self developed to further parse all text from all emails and calculate frequency of words.

Four classes are proposed to label the nature of emails users may have: E-Commerce, Banking, Credit Car, and Others.

We tried also to use clustering to assist in classification. Rather than labeling emails manually by users, we can cluster sets of emails based on some aspects through algorithms and then we need only to pick a name for developed clusters to come up with an email classification scheme. There are several approaches that can be used for clustering unstructured data to create vector space or bag of words model. Most repeated words or top frequency words are used to represent document features. From the complete email dataset words and their frequency will be collected. Stemming is then applied to remove irrelevant words or words, pronouns, verbs, adjectives that are used to connect and complete statements and hence cannot uniquely categorize a statement or a document.

If the popular word exists in the subject email, value is one else value is zero. The model can be reversed where top frequency words can be in columns and rows can represent different emails. Due to the large number of documents, a complete clustering process can be time consuming.

The following algorithm is developed first to perform elementary clustering to save time in initial clustering evaluation:

- Pick a random document from the emails' collection (call it seed1)
- Evaluate the similarity of seed1 to every other email in the collection via cosine similarity
- Save the 100 most similar emails as the seed1 cluster for cosine similarity.
- Repeat for multiple seed emails.

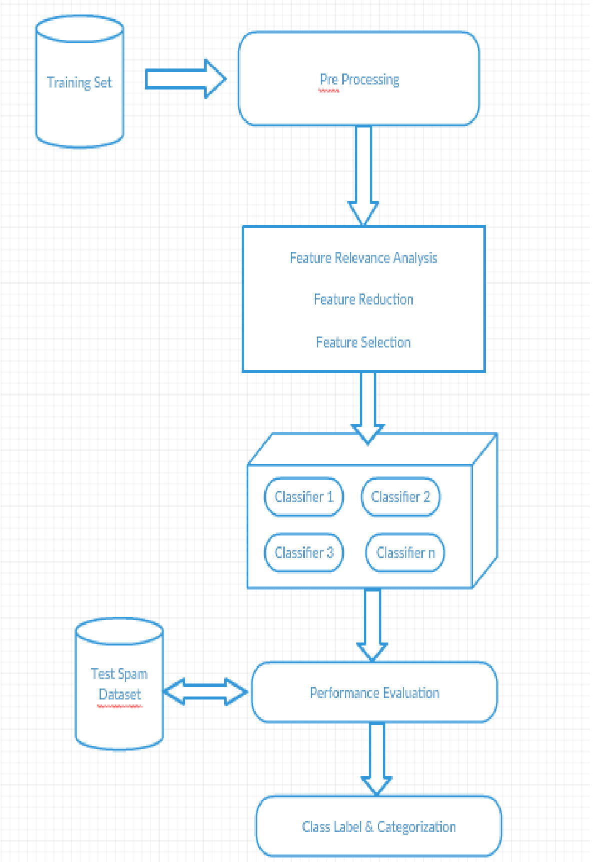

Figure: 1. Filtered Inbox Conceptual Architecture and the flow messages follow, from sending to reception

B.Machine Learning In E-Mail Classification :

Learning here means understood, observe and represent information about some statistical phenomenon. In unsupervised learning one tries to uncover hidden regularities (clusters) or to detect anomalies in the data like spam messages or network intrusion. In e-mail filtering task some features could be the bag of words or the subject line analysis. Thus, the input to e-mail classification task can be viewed as a two dimensional matrix, whose axes are the messages and the features. E-mail classification tasks are often divided into several sub-tasks. First, Data collection and representation are mostly problem-specific (i.e. e-mail messages), second, e-mail feature selection and feature reduction attempt to reduce the dimensionality (i.e. the number of features) for the remaining steps of the task. Finally, the e-mail classification phase of the process finds the actual mapping between training.

We advent proposed job by illustrating a two level hierarchical technique, from a ML point of view, for that we consider that it is preferable to determine and terminate “neutral” sentences, then separate “non-neutral” sentences from the class of interest instead of doing everything in one step [7]. This technique is inspired from the related strategies which show benefits in partitioning text and/or short texts with the help of a hierarchical strategy. First level step is to group short texts according to labels with crisp Neutral and Non-Neutral labels. In the second stage, soft classifier works on crisp group of non-neutral short texts. For each short text, it produces estimated appropriateness or “gradual membership”, without taking any “hard” decision on any of them. This list of ratings is then used by the subsequent phases of the filtering process. Later on phases of the filtering process uses such a list of grades.

.Naïve Bayes classifier method:

Bayesian classifier is working on the dependent events and the probability of an event occurring in the future that can be detected from the previous occurring of the same event [12]. This technique can be used to classify spam e-mails; words probabilities play the main rule here. If some words occur often in spam but not in ham, then this incoming e-mail is probably spam. Naïve bayes classifier technique has become a very popular method in mail filtering software. Bayesian filter should be trained to work effectively. Every word has certain probability of occurring in spam or ham e-mail in its database. If the total of words probabilities exceeds a certain limit, the filter will mark the e-mail to either category. Here, only two categories are necessary: spam or ham. Almost all the statistic-based spam filters use Bayesian probability calculation to combine individual token's statistics to an overall score [1], and make filtering decision based on the score.

The statistic we are mostly interested for a token T is its spamminess (spam rating) [10], calculated as follows:

$$S\,[T] = \frac{C_{Spam}(T)}{C_{Spam}(T)\ +\ C_{Ham}(T)}$$

Where $\mathbf{C}_{Spam}\mathbf{(T)}$ and $\mathbf{C}_{Ham}\mathbf{(T)}$ are the number of spam or ham messages containing token T, respectively.

The above description is used in the following algorithm [10]:

Stage1. Training :

Parse each email into its constituent tokens Generate a probability for each token W

$S[W] = C_{spam}(W) / (C_{ham}(W) + C_{spam}(W))$ store spamminess values to a database

Stage2. Filtering :

For each message M while (M not end) do scan message for the next token T_i query the database for spamminess $S(T_i)$ calculate accumulated message probabilities S[M] and H[M] Calculate the overall message filtering indication by:

$$I\,(M) = f(S[M], H[M])$$

f is a filter dependent function,

$$I[M] = 1 + (S[M] - H[M]))\frac{1}{2}$$

Probability that [M] can be calculated for the threshold and the decision on the treatment of a spam message as spam or not can be made based on this threshold value.

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

$$errorrate\ = \frac{noofincorrectsamples}{totalnumberofsamples}$$

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 is represented graphically.

$$accuracy = \frac{noofcorrectlyclassifiedsamples}{totalnumberofsamples}$$

Recall:

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

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

Precision:

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

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

IV CONCLUSION

In this paper we review some of the most popular machine learning methods and of their applicability to the problem of spam e-mail classification. In term of accuracy we can find that the Naïve bayes and rough sets methods has a very satisfying performance among the other methods, more research has to be done to escalate the performance of the Naïve bayes and hybrid system or by resolve the feature dependence issue in the naïve bayes classifier, or hybrid the Immune by rough sets. Finally hybrid systems look to be the most efficient way to generate a successful anti spam filter nowadays.

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provided me the possibility to complete this paper. A special gratitude I give to the Prof. B.K.Patil whose contribution in stimulating suggestions and encouragement helped me to coordinate my paper especially in writing this paper.

REFERENCES

[1] M. N. Marsono, M. W. El-Kharashi, and F. Gebali, "Binary LNS-based naïve Bayes inference engine for spam control: Noise analysis and FPGA synthesis", IET Computers & Digital Techniques, 2008

[2] Muhammad N. Marsono, M. Watheq El-Kharashi, Fayez Gebali "Targeting spam control on middleboxes: Spam detection based on layer-3 e-mail content classification" Elsevier Computer Networks, 2009

[3] Yuchun Tang, Sven Krasser, Yuanchen He, Weilai Yang, Dmitri Alperovitch "Support Vector Machines and Random Forests Modeling for Spam Senders Behavior Analysis" IEEE GLOBECOM, 2008

[4] Guzella, T. S. and Caminhas, W. M. "A review of machine learning approaches to Spam filtering." Expert Syst. Appl., 2009

[5] Wu, C. "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks" Expert Syst., 2009

[6] Khorsi. "An overview of content-based spam filtering techniques", Informatica, 2007

[7]Hao Zhang, Alexander C. Berg, Michael Maire, and Jitendra Malic. "SVM-KNN: Discriminative nearest neighbour classification for visual category recognition", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006

[8] Carpinteiro, O. A. S., Lima, I., Assis, J. M. C., de Souza, A. C. Z., Moreira, E. M., & Pinheiro, C. A. M. "A neural model in anti-spam systems.", Lecture notes in computer science.Berlin, Springer, 2006

[9] El-Sayed M. El-Alfy, Radwan E. Abdel-Aal "Using GMDH-based networks for improved spam detection and email feature analysis"Applied Soft Computing, Volume 11, Issue 1, January 2011

[10] Li, K. and Zhong, Z., "Fast statistical spam filter by approximate classifications", In Proceedings of the Joint international Conference on Measurement and Modeling of Computer Systems. Saint Malo, France, 2006

[11] Cormack, Gordon. Smucker, Mark. Clarke, Charles " Efficient and effective spam filtering and re-ranking for large web datasets" Information Retrieval, Springer Netherlands. January 2011

[12] Almeida,tiago. Almeida, Jurandy.Yamakami, Akebo " Spam filtering: how the dimensionality reduction affects the accuracy of Naive Bayes classifiers" Journal of Internet Services and Applications, Springer London , February 2011

[13] Yoo, S., Yang, Y., Lin, F., and Moon, I. "Mining social networks for personalized email prioritization". In Proceedings of the 15th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining (Paris, France), June 28 - July 01, 2009