Email Classification Using Data Reduction Method

Rafiqul Islam and Yang Xiang, *member IEEE*
School of Information Technology
Deakin University, Burwood 3125, Victoria, Australia

Abstract – **Classifying user emails correctly from penetration of spam is an important research issue for anti-spam researchers. This paper has presented an effective and efficient email classification technique based on data filtering method. In our testing we have introduced an innovative filtering technique using instance selection method (ISM) to reduce the pointless data instances from training model and then classify the test data. The objective of ISM is to identify which instances (examples, patterns) in email corpora should be selected as representatives of the entire dataset, without significant loss of information. We have used WEKA interface in our integrated classification model and tested diverse classification algorithms. Our empirical studies show significant performance in terms of classification accuracy with reduction of false positive instances.**

I. INTRODUCTION

The Internet is becoming an integral part of our everyday life and the email has treated a powerful tool intended to be an idea and information exchange, as well as for users' commercial and social lives. Due to the increasing volume of unwanted email called as spam, the users as well as Internet Service Providers (ISPs) are facing multifarious problems. Email spam also creates a major threat to the security of networked systems. Email classification is able to control the problem in a variety of ways. Detection and protection of spam emails from the e-mail delivery system allows end-users to regain a useful means of communication. Many researches on content based email classification have been centered on the more sophisticated classifier-related issues [10]. Currently, machine learning for email classification is an important research issue. The success of machine learning techniques in text categorization has led researchers to explore learning algorithms in spam filtering [1, 2, 3, 4, 10, 11, 13, and 14]. However, it is amazing that despite the increasing development of anti-spam services and technologies, the number of spam messages continues to increase rapidly.

Due to the rapid growth of email and spam over the time, the anti-spam engineers need to handle with large volume email database. When dealing with very large-scale datasets, it is often a practical necessity to seek to reduce the size of the dataset, acknowledging that in many cases the patterns that are in the data would still exist if a representative subset of instances were selected. Further, if the right instances are selected, the reduced dataset can often be less noisy than the original dataset, producing superior generalization performance of classifiers trained on the reduced dataset. The goal of instance selection is to select such a representative subset of instances, enabling the size of the new dataset to be significantly reduced Spam is defined as unsolicited commercial email or unsolicited balk email, has become one of the biggest worldwide problems facing the Internet today.

This paper proposes effective and efficient email classification techniques based on data filtering method into the training model. The main focus of this paper is to reduce the instance of the email corpora from training model using ISM, which is less significant in relation to the classification. Our empirical evidence shows that the proposed technique gives better accuracy with reduction of insignificant instances from email corpora. The rest of the paper is as follows: section 2 will describe the related work of classifiers; section 3 will describe the ISM approach; section 4 will present the proposed email classification architecture and its detail description. and section 5 will present the key findings. Finally, the paper ends with conclusion and references in section 6 and 7 respectively.

II. RELATED WORKS

In recent years, many researchers have turned their attention to classification of spam using many different approaches. According to the literature, classification method is considered one of the standard and commonly accepted methods to stop spam [10]. This method is effective for the currently encountered types of spam. The philosophy behind this method is to separate the spam from legitimate emails. The classification approaches can be broadly separated into two different categories. One is based on non-classification algorithms and other is based on classification algorithms.

A. Non-Classification algorithms

Non-classification based methods include heuristic or rule-based methods, white-listing, black-listing, hash-based lists and distributed black-lists. Non-classification based solutions work well because of their simplicity and relatively short processing time [15]. Another key attraction is that it does not require a training period. However, in the context of new filtering technologies and in the light of current spamming techniques, it has several drawbacks. Since these methods are based on standard rule sets, the

Rafiqul Islam is with the School of Information Technology, Deakin University, Burwood VIC 3125, Australia (e-mail: rislam@deakin.edu.au).
Yang Xiang is with the School of Information Technology, Deakin University, Burwood VIC 3125, Australia (e-mail: yang@deakin.edu.au).

system cannot adapt the filter to identify emerging rule changes because spammers can use various methods to defeat filters. Also, when using heuristic methods like black-listing and white-listing, the spammer can easily penetrate the user defenses [16] The sender's email address within an email can be faked, allowing spammers to easily bypass black-lists. According to the research in reference [10,16] the authors noted that although the non-classification based filtering can achieve substantial performance, this method often has a high rate of false positives, making it quite risky to use on its own, as a standard stand-alone filtering system.

B. Classification Algorithms

Classification based algorithms are commonly use learning algorithms. Given that classification algorithms outperform other methods, when used in text classification (TC) [10] and other classification areas like biometric recognition and image classification [10,12,17], researchers are also drawn to its uses for spam filtering.

The email classification can be regarded as a special case of binary text categorization. The key concepts of classification using learning algorithms can be categorized into two classes, $yi \in p$, and there are N labeled training examples: {x1, y1),…,(xn, yn), $x \in \Re d$ where d is the dimensionality of the vector [13].

Many classification methods have been developed with the aid of learning algorithms such as Bayesian, Decision Tree, K-nn (K-nearest neighbour), Support Vector Machine (SVM) and boosting. All these classifiers are basically learning methods and adopt sets of rules. Bayesian classifiers are derived from Bayesian Decision Theory [2,8]. This is the simplest and most widely used classification method due to its manipulating capabilities of tokens and associated probabilities according to the user's classification decisions and empirical performance.

Support vector machine (SVM) is a powerful, state-of-the-art algorithm with strong theoretical foundations based on Vapnik's theory [18]. SVM has a strong data regularization property and can easily handle high dimensional feature spaces. SVM is based on the Structural Risk Minimization (SRM) principle, to find an optimal hyperplane by maximizing the margins that can guarantee the lowest true error due to increasing the generalization capabilities [2].

Random Forest (RF) is a classifier that is based on a combination of many decision tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. RF has excellent accuracy among current classifier algorithms. It also has an effective method for estimating missing data and it maintains accuracy when a large proportion of the data are missing [9].

The IB1 (Instance Based 1) algorithm is the simplest instance-based learning algorithm; it is a nearest neighbour algorithm which in addition normalizes its attributes' ranges, processes instances incrementally, and has a simple policy for tolerating missing values [19]. The performance of IB1 depends on the data structure of the input space.

The decision tree (DT) algorithm is a simple rule-based algorithm based on a set of rules which takes advantage of the sequential structure of decision tree branches so that the order of checking rules and corresponding actions is immediately noticeable. Those conditions and actions that are critical are connected directly to other conditions and actions, whereas those conditions that do not matter are ignored.

The Boosting method is a well established method for improving the performance of any particular classification algorithm. It is a relatively new framework for constructing a highly accurate classification rule by combining many simple and moderately accurate hypotheses (called weak classifiers) into a strong one. This method was initially presented in [19], but its performance is still being studied. Ongoing research introduced a new generation of Boosting method called AdaBoost (Adaptive Boosting) [12], where the mapping functions are themselves learnt from data from another learning algorithm.

Based on our literature we have chosen the above well known classification algorithms (such as SVM, NB, DT, RF, IB1 and Adaboost) in our experiment due to its simplicity and observing their empirical as well as analytical performance. However, it is very difficult to select a good classifier which can always focus on desired output space with ease of training model. The main reason for this is that the sensitivity to the feature selection method varies.

III. DATA FILTERING USING ISM

Instance selection method (ISM) is used for data summarization in [20]. According to the view on [20], the approaches of ISM are divided into three main groups:

i. Noise filters - which remove instances whose class labels do not agree with the majority of their neighbors.
ii. Condensation algorithms - that add instances from the training data to a new dataset if they add new information, but not if they have the same class label as their neighbors;
iii. Prototype construction methods do not focus on selecting which instances of the training data to include in the reduced dataset, but create new instances which are representative of the whole dataset via data squashing or clustering methods.

The first two types of instance selection methods (i and ii), can also be considered prototype selection methods (deciding which instances in the training data to include in the reduced dataset, using either incremental or decremental methods), while the third type are basically prototype construction approaches which seek to find new instances that can represent the whole dataset more compactly. Figure xx provides a taxonomic summary [20] of the related literature and the various approaches to ISM.

There are other ISM's which combine elements of clustering and prototype selection. We adopt in this paper is

similar to cluster classifier approach [20], related to leader sampling, but quite simpler. Prototype points (leaders) are identified through the k-means clustering algorithm [20]. The prototypes are not used for constructing new instances, but form the basis of a prototype selection process. From each cluster we select the closest (100- β .% of the cluster size measured as the Euclidean distance from the cluster centroid. This is a form of stratified sampling based on the similarity of the instances, rather than the class labels, and thus is quite naïve since apriori knowledge about class probabilities is not being used. Of course, this strategy means that it is not being used as a noise filter based on class membership, and so it is closer to a condensation algorithm. The data reduction achieved is β %. We vary the value of β to explore the effectiveness of a classification algorithm on the reduced dataset, compared to the performance of the classification algorithm on the original dataset.

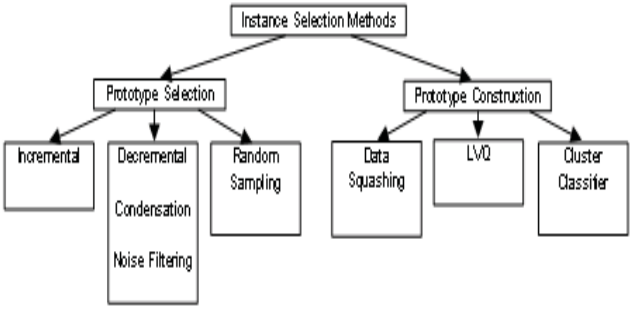

Fig. 1. Taxonomic summary of various ISM approaches

There is no doubt that many of the more sophisticated ISM's would yield improved accuracy for the classifier, but the point here is to explore how the performance on a given ISM varies with instance characteristics. The methodology is broadly applicable and extendable to other ISM's and classification algorithms.

IV. PROPOSED CLASSIFICATION MODEL

This section presents the integrated email classification model based on machine learning algorithms using weka interface. The model includes initial transformation or pre-processing of email sets, feature extraction and selection, classification and finally the evaluation of the classification result.

A. Architecture of the Model

The general approach of our architecture is to extract a broad set of features from each email that can be passed to our classification system. The process is to first initial transformation of incoming email samples, extract a feature and split the entire corpora into different sets. Finally build training and test set and call the weka library for classification and validation. Figure 2 shows the architecture of our classification system.

The objective of the email transformation is to pre-process the email messages into a homogeneous format that can be recognized by the classifier. The initial transformation is to collect the feature of the incoming email contents and convert it into a vector space where each dimension of the space corresponds to a feature of whole corpus of the email message. The initial transformation is often a null step that has the output text as just the input text. Character-set-folding,case-folding and MIME (Multipurpose Internet Mail Extensions) normalization are used for initial transformation in many cases. A corpus of emails is used in our system are collected from PUDA 123, public data sets [10].

Feature extraction which is an important part for data classification. Our general approach to the classification problem is to extract a broad set of features from each email sample that can be passed to our classification engine.

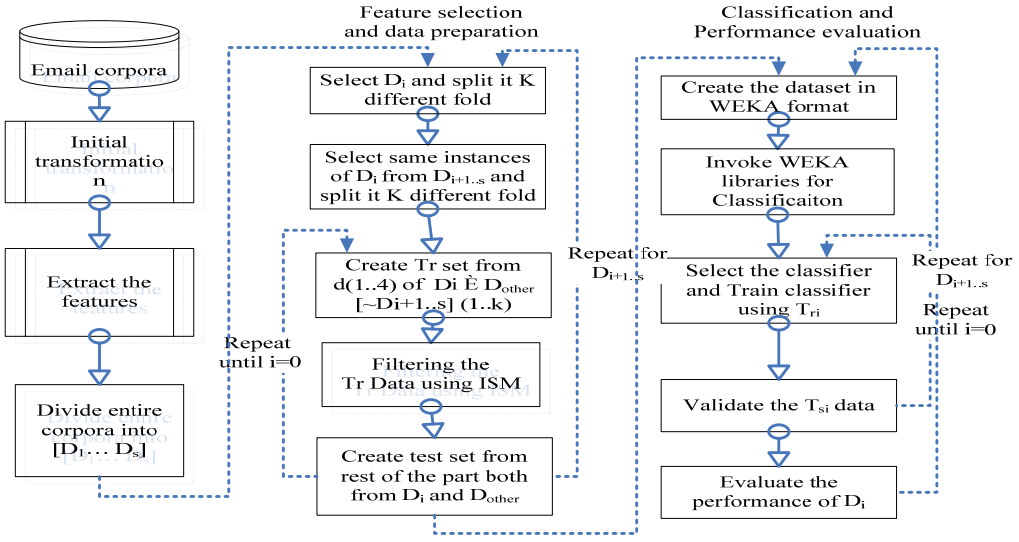

Fig 2. Block diagram of our proposed model.

In our model, tokenisation and domain specific feature selection methods [10] are used for feature extraction. The behavioural features are also included for improving performance, especially for reducing false positive (FP) problems. The behavioural features include the frequency of sending/receiving emails, email attachment, type of attachment, size of attachment and length of the email.

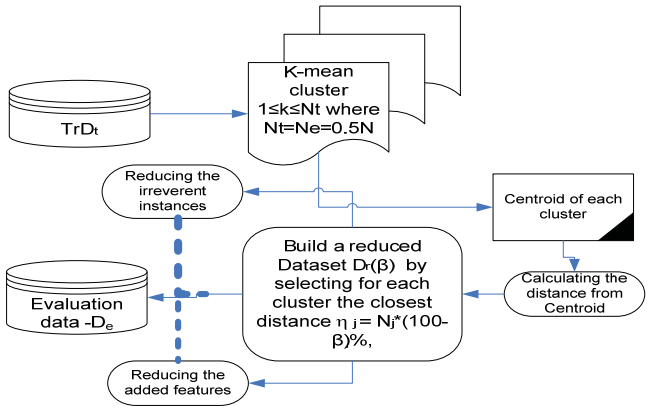

Fig. 3 : Block diagram of data filtering process

B. Filtering the Data

We used an innovative technique to filter the instances of training model of our email classification. The algorithm used in our model to filter the training data is a combination of the naïve ISM to build a reduced dataset coupled with the Bayes classifier. Figure 3 shows out data filtering process used in our architecture.

The methodology used for the algorithm is as follows:

a) Cluster the training data (T_rD_t) using the k-means algorithm, for a given value of k, selecting $1 \le k \le N_t$ where $N_t = N_e = 0.5N$. Cluster j contains Nj instances, for $1 \le j \le k$;
b) Identify each of the k cluster cancroids as a "leader" of the cluster.
c) Add extra feature F_{ex} for measuring the distance of closest η_j instances around leader j, where $\eta_j = N_j*(100-\beta)\%$, for a selected value of β .
d) Build a new dataset by selecting the closest ηj instances around leader j, where $\eta_j = N_j*(100-\beta)\%$, for a selected value of β . The new reduced dataset $D_r(\beta)$ is β % smaller than the original training data D_t;
e) Remove the Fex from the build dataset and build an reduced dataset $D_r(\beta)$ which we called and evaluation dataset D_e
f) The D_e will be applied to the classifier to classify the TsData.

C. Learning and Classification

Machine learning techniques are applied in our model to distinguish between spam and legitimate emails. The basic concept of the training model of a classifier is that it employs a set of training examples of the form $\{(x_1, y_1), ..., (x_k, y_k)\}$ for the projection of a function $f(x)$. Here the values of x are typically of the form $< x_i,1, x_i,2, ..., x_i,n >$, and are composed of either real or discrete values. The y values represent the expected outputs for the given x values, and are usually drawn from a discrete set of classes. Consequently, the task of a learning model involves the approximation of the function $f(x)$ to produce a classifier.

In the training model, we use a particular set and the same number of instances by randomly selected from other sets. Then we split the data in K different fold and use the feature selection technique, only for training set of k-fold data, to reduce the number of features. In our process we use $(.8T_r:.2T_s)$ technique for making k-fold. Then use these training set to train the classifier and evaluate the classification result by using the test data. The same rotating process applies for all other sets.

D. WEKA Interface

In our system, we built an interface program with WEKA for our data classification. Weka is a data mining tools used mainly for classification and clustering. It is a collection of machine learning algorithms to perform data mining tasks. The interface will link with email database to collect the data for pre-processing, as mentioned in previous section. After pre-processing we generate the training and test data sets and then we convert both sets into WEKA format. We pass training set to the WEKA library to train the classifier and then test the effectiveness with test set. Our program is designed in such a way that the system can select the data sets and the corresponding classifiers according to our requirements rather than the default in WEKA.

V. Empirical Evidence

This section presents the classification outcome of different algorithms. In our experiment, we have tested five base classifiers Naive Bayes, SVM, IB1, Decision Table and Random Forest. We also tested adaptive boosting (AdaboostM1) as meta-classifier on top of base classifiers.

Table 1. Average classification results of base classifier

Algorithm	FP	FN	Pr	Re	Acc
NB	0.01	0.19	0.92	0.9	0.923
SMO	0.02	0.07	0.96	0.96	0.964
IB1	0.02	0.08	0.96	0.95	0.958
DT	0.02	0.2	0.95	0.94	0.959
RF	0.02	0.07	0.97	0.96	0.961

We used 6 different data sets from public data [10] in our experiment. The table 1 presents the average of experimental results according to the classifiers. It has been shown that all the classifier accuracy is almost similar except the naïve bayes algorithm which is worst compared to others. The SMO and RF are shows best performance among the

classifiers. Table 2 shows the performance of Meta-classifier (AdaboostM1). It has been shown that the meta-classifier outperforms compared to base classifier, except the SMO.

Table 2. Average classification results of meta-classifier

Algorithm	FP	FN	Pr	Re	Acc
NB	0.01	0.12	0.94	0.92	0.934
SMO	0.03	0.12	0.96	0.95	0.965
IB1	0.01	0.06	0.96	0.95	0.965
DT	0.02	0.06	0.96	0.95	0.965
RF	0.01	0.07	0.97	0.96	0.972

Figure 4 shows the comparison of classifiers accuracy with and without meta-classifiers. It is clear from figure 3 that there is huge variation in Naïve bayes classier if we apply meta-classifier on top of base classifier. The comparison of SMO is almost same and some slide variation with IB1 and RF. There is also considerable variation of meta-classifier where decision tree is the base algorithm. It is to be noted that the performance of the classifier is depend on the data set characteristics, as mention earlier. It is therefore comparing the above recent works our performance is significant.

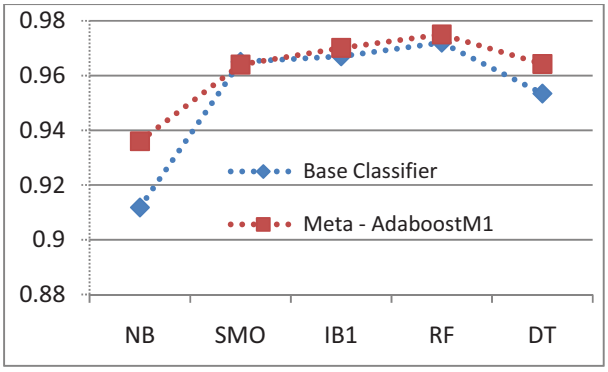

Fig 4. Comparison of accuracy (with and without boosting)

VI. Conclusion and Future Work

This paper presents and effective email classification technique based on an innovative data filtering technique into the training model. In our data filtering process, we have used cluster classifier technique to reduce the insignificant instances from our training model. After investigation of different classification algorithms, we have chosen five classifiers based on our simulation performance and we have used meta-learning technique (Adaboost) on top of every classifier. Our empirical performance shows that, we achieved overall classification accuracy above 97%, which is significant. In our future work we have a plan to consider the features from dynamic information from regular incoming emails and pass to our classification method to achieve better performance.

VII. References

1. Zhang, J., et. al,. A. Modified logistic regression: An approximation to SVM and its applications in large-scale text categorization. In Proceedings of the 20th International Conference on Machine Learning. AAAI Press, pp.888–895,2003.
2. M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. A bayesian approach to filtering junk e-mail. In Learning for Text Categorization: Papers from the Workshop, Madison, Wisconsin, AAAI Technical Report WS-98-05, 1998.
3. Androutsopoulos, I., et.,al,. Learning to filter spam e-mail: A comparison of a Naive Bayesian and a memory-based approach. In Proceedings of the Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases. Lyon, France, 1–13, 2000.
4. H. Drucker, B. Shahrary and D. C. Gibbon, “Support vector machines: relevance feedback and information retrieval,” Inform. Process. Manag. 38, 3, 305–323, 2003.
5. Islam, R, Chowdhury, M. Zhou, W, “An Innovative Spam Filtering Model Based on Support Vector Machine”, Proceedings of the IEEE International Conference on Intelligent Agents, Web Technologies and Internet Commerce, Volume 2, 28-30 Page(s):348 – 353, 2005
6. Cohen, W. and Singer, Y. Context-sensitive learning methods for text categorization. ACM Transactions on Information Systems 17, 2, pp. 141–173, 1999.
7. Cristianini, N. and Shawe-Taylor, J.. An introduction to Support Vector Machines and other kernel-based learning methods. Cambridge University Press, 2000.
8. Kaitarai, H., Filtering Junk e-mail: A performance comparison between genetic programming and naïve bayes, Tech. Report, Department of Electrical and Computer Engineering, University of Waterloo, November 1999
9. Liu Huan and Yu Lei, Toward Integrating Feature Selection Algorithms for Classification and Clustering, IEEE Transaction on Knowledge and Data Engg. Vol. 17, No. 4, 2005.
10. Islam , R., and Wanlei Zhou. (2008) An innovative analyser for multi-classifier email classification based on grey list analysis, The Journal of Network and Computer Applications, Elsevier, Vol 32, Issue 2, pp. 357-366.
11. Islam, R, Chowdhury, “Spam filtering using ML Algorithms” Proceedings of the WWW/Internet conference, Portugal, 2005
12. Islam, M. and Zhou, W. (2007) Architecture of Adaptive Spam Filtering Based on Machine Learning Algorithms, Accepted for publication (Springer Verlag in its Lecture Notes in Computer Science series) on ICA3PP 07, June 11-14, China.
13. Islam, M., Chowdhury, M. and Zhou, W. (2007) Dynamic feature selection for spam filtering using support vector machine Support Vector Machine, 6th IEEE International Conference on Computer and Information Science (ICIS 2007), July 11-13, 2007, Melbourne, Australia
14. Chong, P.H.J, et. Al. “Design and Implementation of User Interface for Mobile Devices” IEEE Transactions on Consumer Electronics, Vol. 50, No. 4, 2004.
15. Hunt, R. and J. Carpinter (2006). Current and New Developments in Spam Filtering. IEEE International Conference on Networks, ICON '06, vol. (2), pp.(1-6).
16. Wang, X.-L. and I. Cloete (2005). Learning to classify email: a survey. IEEE ICMLC 05, IEEE, vol. (9), pp.(5716-5719), Isbn:0-7803-9091-1.
17. Eleyan, A. and H. Demirel (2007). Face Recognition using Multiresolution PCA. IEEE International Symposium on Signal Processing and Information Technology, 2007, pp.(52-55)
18. V.N. Vapnik, Statistical Learning Theory, John Wiley, New York, 1999.
19. A. Kapoor and J. Spurlock. Binary feature extraction and comparison. In Presentation at AVAR 2006, Auckland, December 35, 2006.
20. Miles, S. Kate and Rafiqul Islam (2009), Meta-learning of instance selection for data summarization. Book chapter in Meta-learning in Computational Intelligence, Publisher: Springer Verlag.