International Journal of Scientific Research in Computer Science and Engineering
Vol.6, Issue.2, pp.33-37, April (2018)

Research Paper

E-ISSN: 2320-7639

An Efficient Spam Filtering using Supervised Machine Learning Techniques

Deepika Mallampati

Department of Computer Science, Sreyas Institute of Engineering and Technology, Jawaharlal Nehru Technological University, Hyderabad, India

Corresponding Author: mokshhyd@gmail.com, Tel.: +91-9912538433

Available online at: www.isroset.org

Received: 07/Apr/2018, Revised: 12/Apr/2018, Accepted: 26/Apr/2018, Online: 30/Apr/ 2018

Abstract— Email spam or junk e-mail (unsolicited e-mail "usually of a commercial nature sent out in bulk") is one of the major problem of the today's Internet, carrying financial damage to companies and annoying individual users. Among the approaches developed to stop spam, filtering is an important and popular one. Common uses for mail filters comprise organizing incoming email and removal of spam and computer viruses. In proposed work, we employed supervised machine learning techniques to filter the email spam messages. Extensively used supervised machine learning techniques namely C 4.5 Decision tree classifier, Multilayer Perceptron, Naïve Bayes Classifier are used for learning the features of spam emails and the model is built by training with known spam emails and legitimate emails.

Keywords-Spam,SpamFilter,SupervisedMachineLearning

I. INTRODUCTION

In recent years, e-mails have become a common and important medium of communication for most Internet users. However, spam, also known as unsolicited commercial/ bulk e-mail, is a bane of e-mail communication. Spam is commonly compared to paper junk mail. However the difference is that junk mailers pay a fee to distribute their materials, whereas with spam the recipient or ISP pays in the form of additional bandwidth, disk space, server resources, and lost productivity. If spam continues to grow at the current rate, the spam problem may become unmanageable in the near future. A study estimated that over 70% of today's business emails are spam [1]; therefore, there are many serious problems associated with growing volumes of spam such as filling users' mailboxes, engulfing important personal mail, wasting storage space and communication bandwidth, and consuming users' time to delete all spam mails. Spam mails vary significantly in content and they roughly belong to the following categories: money making scams, fat loss, improve business, sexually explicit, make friends, service provider advertisement, etc.[2]

This paper is organized as follows Section I contains the introduction of Spam , Section II contain the related work of existing spam filtering techniques, Section III presents an effective supervised machine learning techniques with methodology. In section IV showing comparative experimental results for proposed strategies with performance and classification accuracy in a tabular form. Finally Section V concludes research work.

II. RELATED WORK

Nosseir , Khaled Nagati and Islam Taj-Eddin performed a work," Intelligent Word-Based Spam Filter Detection Using Multi-Neural Networks". They proposed an approach which is character-based technique. This approach uses a multi-neural networks classifier. Each neural network is trained based on a normalized weight obtained from the ASCII value of the word characters. Results of the experiment show high false positive and low true negative percentages. [3]. R. Kishore Kumar, G. Poonkuzhali, P. Sudhakar provides the analysis of email spam classifier through data mining techniques. In their work," Comparative Study on Email Spam Classifier using Data Mining Techniques " spam dataset is analyzed using TANAGRA data mining tool to explore the efficient classifier for email spam classification. Initially, feature construction and feature selection is done to extract the relevant features. Then various classification algorithms are applied over this dataset and cross validation is done for each of these classifiers.
Finally, best classifier for email spam is identified based on the error rate, precision and recall. [4]. Rafiqul Islam and Yang Xiang performed classification of user emails form penetration of spam. In their paper," Email Classification Using Data Reduction Method" an effective and efficient

email classification technique based on data filtering method is presented. They 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. They have used WEKA interface in our integrated classification model and tested diverse classification algorithms. Their empirical studies show significant performance in terms of classification accuracy with reduction of false positive instances. [5].

Asmeeta mali performed a work," Spam Detection using Bayesian with Pattern Discovery". In her paper she presents an effective technique to improve the effectiveness of using and updating discovered patterns for finding relevant and interesting information. Using Bayesian filtering algorithm and effective pattern Discovery technique we can detect the spam mails from the email dataset with good correctness of term. [6].

Vandana Jaswal proposes an image spam detection system that uses detect spam words. In her work," Spam Detection System Using Hidden Markov Model" filtering method are used to detect stemming words of spam images and then use Hidden Markov Model of spam filters to detect all the spam images. [7].

In year 2011, Saadat Nazirova performed a work," Survey on Spam Filtering Techniques". In this paper the overview of existing e-mail spam filtering methods is given. The classification, evaluation, and comparison of traditional and learning-based methods are provided. Some personal anti-spam products are tested and compared. The statement for new approach in spam filtering technique is considered. [8]. As we are working on the approach that gives better result than other approaches to identify spam mail we need danger theory and dendritic cell algorithm. Here some other work on DCA is defined in literature.

Neha Singh performed a work,"Dendritic Cell algorithm and Dempster Belief Theory Using Improved Intrusion Detection System". To minimize false alarm rate she proposed novel dual detection of IDS based on Artificial Immune System that integrating the Dendrite Cell Algorithm and Dempster Belief theory in her work. [9]. In year 2007 Green smith submitted his work,"The Dendritic Cell Algorithm". This is a novel immune inspired algorithm based on the function of the dendritic cells of the human immune system. In nature, dendritic cells function as natural anomaly detection agents, instructing the immune system to respond if stress or damage is detected. Dendritic cells are a crucial cell in the detection and combination of 'signals' which provide the immune system with a sense of context. The dendritic Cell Algorithm is based on an abstract model of dendritic cell behaviour, with the abstraction process performed in close collaboration with immunologists. This algorithm consists of components based on the key properties of dendritic cell behavior, which involves data fusion and correlation components.[10]

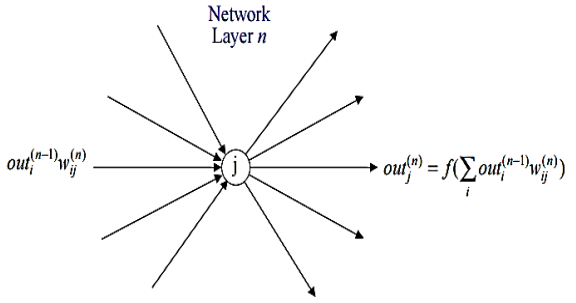
III. PROPOSED METHODOLGY

Most of the unsolicited mail filtering techniques is based on text categorization strategies. Thus filtering spam activates a classification trouble. In our work, rules are framed to extract a function vector from e-mail. As the traits of discrimination are not nicely described, it's far extra convenient to follow device getting to know strategies. Three system studying algorithms, C4.5 Decision tree classifier, Multilayer perceptron and Naïve Bayes classifier are used for studying the category model.

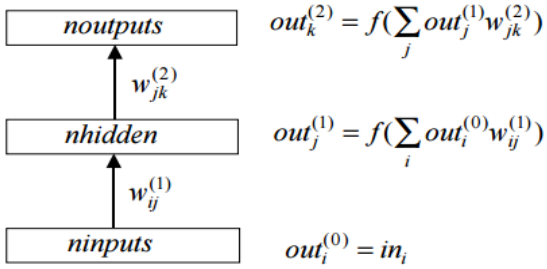
1). Multilayer Perceptron (MLP):

It is most widely used neural network classifier. MLP networks are standard-motive, bendy, nonlinear model along with a number of devices organized into multiple layers. The complexity of the MLP network may be changed by varying the range of layers and the range of devices in each layer. Given sufficient hidden units and sufficient records, it has been shown that MLPs can approximate genuinely any characteristic to any desired accuracy. In different sense, MLPs are generic approximators. MLPs are precious gear in problems while one has little or no understanding approximately the shape of the connection among enter vectors and their corresponding outputs.

Dealing with multi-layer networks is easy if we use a sensible notation. We simply need another label *(n)* to tell us which layer in the network we are dealing with.



Each unit *j* in layer *n* receives activations $out_i^{(n-1)} w_i^j$ from the previous layer of processing units and sends activations $out_j^{(n)}$ to the next layer of units. Conventionally, the input layer is layer 0, and when we talk of an *N* layer network we mean there are N layers of weights and *N* non-input layers of processing units. Thus a two layer Multi-Layer Perceptron takes the form:



It is clear how we can add in further layers, though for most practical purposes two layers will be sufficient. Note that there is nothing stopping us from having different activation functions *f(x)* for different layers, or even different units within a layer.

We can use the same ideas as before to train our *N*-layer neural networks. We want to adjust the network weights w_i^{jn} in order to minimise the sum-squared error function.

$$E(W_{ij}^{(n)}) = \frac{1}{2}\sum_p \sum_j (targ_j^p - out_j^{(N)}(in_i^p))^2$$

and again we can do this by a series of gradient descent weight updates

$$\Delta w_{kl}^{(m)} \dot{\eta} = -\eta \frac{\partial E(w_{ij}^{(n)})}{\partial w_{kl}^{(m)}}$$

Note that it is only the outputs out_n^j of the final layer that appear in the error function. However, the final layer outputs will depend on all the earlier layers of weights, and the learning algorithm will adjust them all. The learning algorithm automatically adjusts the outputs out_n^j of the earlier (hidden) layers so that they form appropriate intermediate (hidden) representations.

A network as a whole will usually learn most efficiently if all its neurons are learning at roughly the same speed. So may be different parts of the network should have different learning rates η. There are a number of factors that may affect the choices:
1. The later network layers (nearer the outputs) will tend to have larger local gradients (deltas) than the earlier layers (nearer the inputs).
2. The activations of units with many connections feeding in or out of them tend to change faster than units with fewer connections.
3. Activations required for linear units will be different for sigmoid units.
4. There is empirical evidence that it helps to have different learning rates η for the thresholds/biases compared with the real connection weights.
In practice, it is often quicker to just use the same rates η for all the weights and thresholds, rather than spending time trying to work out appropriate differences. A very powerful approach is to use evolutionary strategies to determine good learning rates.

2)C 4.5 Decision Tree Induction:

Decision Tree Classification generates the output as a binary tree like structure called a decision tree, in which each branch node represents a choice between a number of alternatives, and each leaf node represents a classification or decision. A Decision Tree model contains rules to predict the target variable. This algorithm scales well, even where there are varying numbers of training examples and considerable numbers of attributes in large databases.

One limitation of ID3 is that it is overly sensitive to features with large numbers of values. This must be overcome if you are going to use ID3 as an Internet search agent. We address this difficulty by borrowing from the C4.5 algorithm, an ID3 extension. ID3's sensitivity to features with large numbers of values is illustrated by Social Security numbers. Since Social Security numbers are unique for every individual, testing on its value will always yield low conditional entropy values. However, this is not a useful test. To overcome this problem, C4.5 uses "Information gain," This computation does not, in itself, produce anything new. However, it allows to measure a gain ratio.
Gain ratio, is defined as follows:

$$GainRatio(p,T) =$$

Where Split Info is:

$$SplitInfo(p,test) = -\sum_{j=1}^{n} P'\left(\frac{j}{p}\right) Xlog(P')$$

P' (j/p) is the proportion of elements present at the position p, taking the value of j-th test. Note that, unlike the entropy, the foregoing definition is independent of the distribution of examples inside the different classes. Like ID3 the data is sorted at every node of the tree in order to determine the best splitting attribute. It uses gain ratio impurity method to evaluate the splitting attribute (Quinlan, 1993). [10] Decision trees are built in C4.5 by using a set of training data or data sets as in ID3. At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. Its criterion is the normalized information gain (difference in entropy) that results from choosing an attribute for splitting the data. The attribute with the highest normalized information gain is chosen to make the decision.

A. Attributes of unknown value: During the construction of the decision tree, it is possible to manage data for which

some attributes have an unknown value by evaluating the gain or the gain ratio for such an attribute considering only the records for which this attribute is defined. [2] Using a decision tree, it is possible to classify the records that have unknown values by estimating the probabilities of different outcomes.

B. Attributes value on continuous interval: C4.5 also manages the cases of attributes with values in continuous intervals as follows. Let us say that Ci attribute a continuous interval of values. Examine the values of this attribute in the training data. Let that these values are in ascending order, A_1, $A_{2....}$, A_m .Then for each of these values, the partitioned between records those that have values of C, less than or equal to A_j and those which have a value larger then A_j values. For each of these partitions gain is calculated, or the gain ratio and the partition that maximizes the gain is selected.

C. Pruning: Generating a decision to function best with a given of training data set often creates a tree that over-fits the data and is too sensitive on the sample noise. Such decision trees do not perform well with new unseen samples.

3). Naïve Bayes Classification

The naive bayes classifier (NB) is a simple but effective classifier which has been used in numerous applications of information processing including, natural language processing, information retrieval, etc. The Naive Bayes Classifier technique is based on Bayesian theorem and is particularly suited when the dimensionality of the inputs is high. Naïve Bayes classifiers assume that the effect of a variable value on a given class is independent of the values of other variable. The Naive-Bayes inducer computes conditional probabilities of the classes given the instance and picks the class with the highest posterior. Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting.

The work is based on rules and uses a score-based system. The rules are framed by analyzing the mail header information, keyword matching and the body of the message. And a relative score is assigned to each rule. There are number of rules framed by considering the various features that will aid to identify the spam messages effectively. Each rule performs a test on the email, and each rule has a score. When an email is processed, it is tested against each rule. For each rule found to be true for an email, the score associated with the rule is added to the overall score for that email. Once all the rules have been used, the total score for the email is compared to a threshold value. If the score exceeds the threshold, then the email is marked as spam and the others are classified as legitimate mail. The following are the rules used:

Table .1 Scheme of rules assigned to each spam feature

From name meaningful
From domain name
Blocked IP
Apostrophe in From name
From name in Auto Whitelist (AWL)
From address in User's Block list
From address in User's White list
Content Type
Content Boundary exists
To name meaningful
To address Undisclosed recipients
To header original
From address and To address same
Is subject present
Subject content has obfuscate words
Is forwarded message
Is reply message
Subject Reply without reference header
Is message body exists
Sensual message
Repeated double quotes in body
Character set includes foreign language
More blank lines in body

In these rules, some are simple and some are associated with one another. A simple rule could search for a word 'Viagra' in subject line of an email, while a complex rule may involve comparing an email against an online database of spam. Each rule adds to the overall score, so an email that triggers only one rule due to the use of the word 'Viagra' will not necessarily mark an email as spam. However, if an email triggers several rules, it will have a combined score that could be over the threshold and the mail could be marked as spam.

The email spam filtering has been carried out using WEKA. The Weka, Open Source, Portable, GUI-based workbench is a collection of state-of-the-art machine learning algorithms and data pre processing tools. The machine learning techniques Naïve Bayes Classifier, C4.5 Decision tree classifier, Multilayer Perceptron are used for training the dataset in WEKA environment.

IV RESULTS

Table.2 Comparative results of the classifiers

Evaluation Criteria	Naïve Bayes	J48	MLP
Training time (secs)	0.15	0.20	138.05

Correctly Classified Instances	1479	1449	1490
Prediction Accuracy (%)	98.6	96.6	99.3
False Positive (%)	5	4	1

The performance of the classifiers are summarized in Table II and shown in Fig.1 and Fig.2.

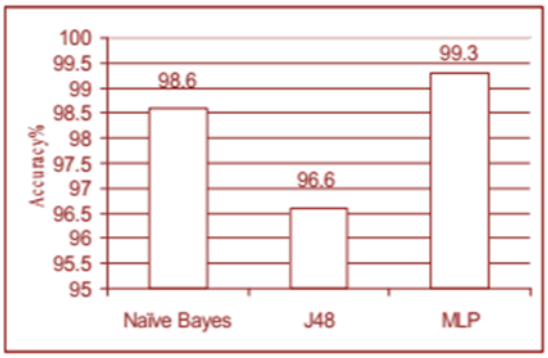


Fig. 1 Classification Accuracy

The performance of the three models was evaluated based on the three criteria, the prediction accuracy, learning time and false positive rate. Multilayer perceptron predicts better than other algorithms.

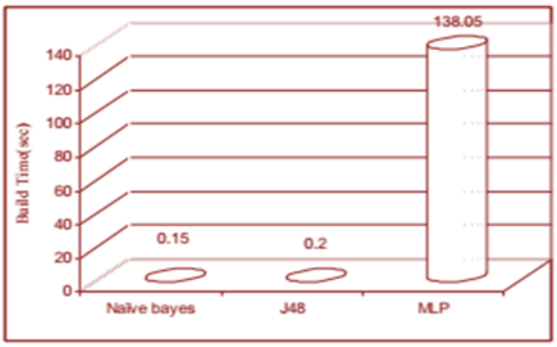


Fig. 2 Learning Time of the Model

Multilayer perceptron, the neural network classifier consumes more time to build the model. The naivebayes, the probabilistic classifier and decision tree model tends to learn more rapidly for the given data set.

V CONCLUSION

This paper evaluates an effective supervised machine learning techniques for email spam classification. In our work, we produced spam and legitimate message amount from the most recent mails and employed machine learning techniques to build the model. The performance of the model is assessed using 10-fold cross validation and observed that Multilayer Perceptron classifier out performs other classifiers and the false positive rate also very low compared to other algorithms'. Email spam filters using this approach can be adopted either at mail server or at client side to reduce the amount of spam messages and to reduce the risk of productivity loss, bandwidth and storage usage.

REFERENCES

[1] Aladdin Knowledge Systems, "Anti-spam white paper, www.csisoft.com/security/aladdin/esafe_antispam", Retrieved December 28, 2011.

[2] F. Smadja, H. Tumblin, "Automatic spam detection as a text classification task", in: Proc. of Workshop on Operational Text Classification Systems, 2002.

[3] Ann Nosseir , Khaled Nagati and Islam Taj-Eddin, "Intelligent Word-Based Spam Filter Detection Using Multi-Neural Networks", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 2, No 1, March 2013 ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784.

[4] R. Kishore Kumar, G. Poonkuzhali, P. Sudhakar," Comparative Study on Email Spam Classifier using Data Mining Techniques", Proceedings of the International MultiConference of Engineers and Computer Scientists 2012 Vol I, IMEC2012, March 14-16,2012, Hong Kong, ISBN: 977-988-19251-1-4.

[5] Rafiqul Islam and Yang Xiang, member IEEE, "Email Classification Using Data Reduction Method" created June 16, 2010.

[6] Asmeeta Mali, "Spam Detection Using Baysian with Pattren Discovery", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-2, Issue-3, July 2013.

[7] Vandana Jaswal, " Spam Detection System Using Hidden Markov Model", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 7, July 2013 ISSN: 2277 128X.

[8] Saadat Nazirova, "Survey on Spam Filtering Techniques", Communications and Network, 2011, 3, 153 160, doi:10.42 36/cn.2011.33019 Published Online August 2011 (http: //www.SciRP.org /journal/cn).

[9] Neha Singh,"Dendritic Cell Algorithm and Dempster Belief Theory Using Improved Intrusion Detection System ", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 7, July 2013 ISSN: 2277 128X.

[10] Julie Greensmith, "The Dendritic Cell Algorithm", Thesis submitted to the University of Nottingham for the degree of Doctor of Philosophy October 2007.

Author Profile

Deepika Mallampati is currently the Assistant Professor,Department of Computer Science & Engineering, SREYAS INSTITUTE OF ENGINEERING & TECHNOLOGY, Nagole, Hyderabad.She obtained her Bachelor of Engineering Degree and her Masters degree in Computer Science and Engineering from *Jawaharlal Nehru Technological University, Hyderabad*. Her main research work focuses on Big Data Analytics and Data Mining. She has 9 years of teaching experience .

