Volume 4, Issue 6, June 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Effective Email Classification for Spam and Non-Spam

Savita Pundalik Teli*
Student in Dr. D. Y. Patil College, of Engineering, Ambi
. University of Pune, M.S, India

Santoshkumar Biradar
Professor in Dept of Computer Engineering, Dr. D. Y. Patil College of Engineering, Ambi, University of Pune, M.S, India

Abstract— Emails are used daily by number of users to communicate with each other through the world. Today large number of spam emails are causing serious problem for Internet user and Internet service. Spam is a bad email or unwanted email or unsolicited email. Email users daily receive more number of spam emails rather than legitimate emails, so it's necessary to have effective spam filtering technique. These filtering technique are based on data classification. Data classification is used to separate spam and legitimate emails. There are many classification techniques used for spam filtering. In this paper we propose an algorithm for email classification based on Naïve Bayesian theorem. The purpose is to automatically classify mails into spam and legitimate message. The mails are classified on the bases of email body. The proposed algorithm is effective and reasonable method for email classification

.

Keywords— Bayesian, Data mining, Email classification, Ham, Spam filtering.

I. INTRODUCTION

Email is the effective way of communicating with each other. Spam mails are unwanted emails or bad emails or unsolicited email which user receives. Spam mails are used for spreading virus or malicious code, for fraud in banking, for phishing, and for advertising. So it can cause serious problem for internet users such as loading traffic on the network, wasting searching time of user and energy of the user, and wastage of network bandwidth etc. According to investigation today user receives more spam emails then non spam emails. To avoid spam/irrelevant mails we need effective spam filtering methods. At the same time, a large part of email traffic consists of nonperson, non time critical information that should be filtered. Spam/Irrelevant emails greatly affect the efficiency and accuracy of the aimed processing work. As a result, there has recently been a growing interest in creating automated systems to help users filter the emails.

Below is the summary on how the filtering is performed. Filtering is nothing but arranging mails in specified order, such as removing spam, Deleting virus, and allowing non spam mails. Existing filtering techniques use classification.

Classification [6] is the technique of data mining. Data mining is defined as a discovering useful knowledge from data. Various applications of Data mining's are sales transactions, stock trading records, product descriptions, sales promotions, company profile and performance, medical and health industry, and customer feedback, reporting online analytical pro-cessing, business performance management and so on. Classification is the process of finding model that describe and distinguishes data classes or concepts. The models are derived based on the analysis of set of objects for which the class label is unknown. Classification is a type of data analysis that extracts models describing important data classes. Classification consists of two steps. First is process learning step: where a classification model is constructed and second classification step: In this step the model is used to predict class labels depending on the learning step for given data. Suppose we consider a scenario – You have just returned from a two week holiday. There has been no phone, no email for two weeks, and now you are back. You open your inbox and found there are 257 new messages! How could you manage to read all of them? Probably, you will spend the day endeavoring to sort out all this mails. Having finished this burdensome work, you seem like you need a holiday again. What is worse is that most of those messages are out of your interest/spam emails. Here comes the need for automatic email classification system that would sort spam and non spam emails. Thus keeping a much precious time of the users. In this paper we use the following terminology: Ham: legitimate (i.e. relevant or non-spam emails). Spam message: illegitimate message (i.e. irrelevant emails). False positive ratio: (number of false positives) / (number of ham messages) (Note that this ratio may be higher than the error rate).

The rest of the paper is organized as follows. The next section describes the literature Survey of existing systems. Section III introduces the Implementation Details or proposed algorithm for email classification. Section IV presents results. The final section consists of the conclusion and future scope, respectively.

II. LITERATURE SURVEY

We have performed survey literature on major three spam filtering technique namely Bayesian Theorem, SVM and K-NN respectively.

A. Bayesian Theorem

An attempt to email Spam filtering based on Naive Bayes classifier is done by S.Roy, A.Patra, S.Sau, K.Mandal and S.Kunar in [7]. Naive Bayes classifier uses the Baye's theorem of conditioned probability to recognize an email to be spam or not. Conditioned Probability is given as

$$P(cj|d) = P(d|cj)\,P(cj) / P(d) \quad (1)$$

Considering each attribute and class label as a random variable and given a record with attributes (A1, A2,… An), the goal is to predict class C. Specifically, we want to find the value of C that maximizes P(C| A1, A2,…An).

The approach taken is to compute the posterior probability P(C| A1, A2,…An) for all values of C using the Bayes theorem.

$$P(C|A1A2\ldots An) = \frac{P(A1A2\ldots..An|C)\,P(C)}{P(A1A2....An)} \quad (2)$$

So you choose the value of C that maximizes P(C| A1,A2,…An). This is equivalent to choosing the value of C that maximizes P(A1,A2,…An | C) P(C).

Naïve Bayesian prediction requires each conditional probability be non zero. Otherwise, the predicted probability will be zero.

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i) \quad (3)$$

In order to overcome this, we use probability estimation from one of the following:

$$\text{Original} : P(A_i \mid C) = \frac{N_{ic}}{N_c}$$

$$\text{Laplace} : P(A_i \mid C) = \frac{N_{ic}+1}{N_c + c}$$

$$\text{m - estimate} : P(A_i \mid C) = \frac{N_{ic} + mp}{N_c + m}$$

c: number of classes

p: prior probability

m: parameter

(4)

The complete process of email classification can be divided into two phases:

1) Training Phase: In this each email is first individually categorized to a category (spam/ham). Remove html tags, stop words, special characters, articles, proverbs. Extract keywords calculate the frequency of keywords and then save it to database for the selected category.

2) **Classification Phase:** In this phase the newly arrived mail is first converting to lower case and stop words, html tags, special characters, articles, proverbs are removed. Extracts the keywords from mails and then calculate the probability of these words from the learning dataset. If the spam probability is higher than the mail is spam otherwise its no spam/legitimate mail.

It is observed that in order to classify, predict a spam email from a non spam one and to increase the accuracy of the above algorithm, the following techniques and assumptions are used [7]:

- Sorting spam or non spam according to the language, words and then count.
- If a word does not exist, consider to approximate P (word|class) using Laplacian.
- Threshold value.
- The Learning Dataset contains each word that filtering is used to determine if a message is spam or not. Beside each word, there are two numbers. The first number is the number of times that the word has occurred in legitimate emails. The second number is the number of times that the word has occurred in spam emails.

B. Support Vector Machine

Support vector machine model had been the most successful algorithm in the field of Text classification. It is mainly popular because of its ease of implementation and high accurate results. Originally it was presented to classify data into two fixed classes making it supervised non-probabilistic binary classifier. But with time it has been used by researchers for classifying data into N categories. One of the works in email classification using SVM discussed in this paper is by Fagbola Temitayo, Olabiyisi Stephen and Adigun Abimbola [8]. They have combined the SVM with the Genetic algorithm to enhance the performance of SVM. In its simplest form SVM can be used to represent a document in vector space where each feature (word) represents one dimension. Identical feature denotes same dimension. Two of the parameters namely Term Frequency (TF) and TF-Inverse Document Frequency (TF-IDF) add value to these vectors. TF— the number of times a word occurs in a document. Harris Drucker, Donghui Wu, and Vladimir N. Vapnik proposes a word is a feature only if it occurs in three or more documents which prevents misspelled words and words used rarely.

TF-IDF uses the above TF multiplied by the IDF (inverse document frequency). The document frequency (DF) is the number of times that word occurs in all the documents. The inverse document frequency (IDF) is defined as

$$IDF(W_i) = \log(|D|/DF(w_i)) \quad (5)$$

TF counts the number of times w_i feature occurs in a document D, where as DF counts the number of documents in which feature w_i occurred at least once. Using these two parameters we can assign priorities to the features used in creating vector in space. Performing the dot product of new vector with the vectors representing different classes in space a similarity factor of the new item to be classified is calculated and compared. SVM searches for the hyper plane which can separate the documents as category A and B (Binary SVM) with maximal margin. Fig 1 presents a rough sketch how SVM performs the classification.

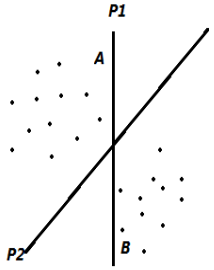

Fig. 1 SVM classification using two planes P1 and P2

Given some document D, comprising a set of n points of the form

$$D = \{(x_i, y_i) \mid x_i \in R, y \in \{-1,1\}\} \quad (6)$$

Where the yi is either 1 or −1, indicating the class A or B to which the point xi belongs. There can be multiple hyper plane which can divide a given set of points. Here we have given example of two such linear planes P1 and P2, where P2 divides the points with maximum margin. The equation of the plane can be given by the equation,

$$w.x - b = 0 \quad (7)$$

Where**.** denotes the dot product and w the normal vector to the hyper plane, and parameter b plays a role to find out the offset of plane along the vector w from origin. In the work [8], along with the SVM the authors introduce the Genetic Algorithm GA, which mainly serves the purpose of optimization of the feature set obtain in plain SVM using GA fitness function. Thus their approach is based on GA-SVM algorithm.

A brief algorithm can be presented as follows [8]

- Convert each mail from training set to an xlsx format.
- Each row contains a mail in form of features stored in columns.
- A label is associated with each feature as (1,-1) to indicate a spam or not.
- Around 7000 total most frequent words are recognised with a unique id in keyword set.
- Dataset is passed to GA-SVM algorithm for classification.

The GA-SVM algorithms show an improvement from the SVM algorithm for spam detection as obtained from the results. Both in terms of accuracy and computational time the GA-SVM has shown an improvement from SVM [8].

Table 1. SVM V/s GA-SVM results [8]

Classifier	Accuracy (%)	Computational time(s)
SVM	90	149.9844
GA-SVM	93.5	119.562017

C. K- Nearest Neighbour (K-NN)

Another important contribution to email classification using machine learning approach is done by M. Chang and C.K. Poon in [9]. They have compared the performance of three algorithms namely K-NN, K-NN with TF-IDF and cosine measure and Naive Bayes classifier of which our focus of discussion is K-NN.

K-NN is a classification algorithm which classifies objects based on K objects having closest pattern in the training sample. To find the closest patter objects a number of similarity measures are used among which the most popular is Euclidean distance calculated as:

$$D(p1, p2) = \sqrt{(x2 - x1)2 + (y2 - y1)2} \quad (8)$$

Where p1 and p2 represents the points or objects in space having coordinates (x1, y1) and (x2, y2) respectively. Using such a model a training set can be generated for different classes associated with each point. Consider a New Object N is to be classified as one of the predefined classes A or B. The Euclidean distance of that objects will be calculated with the other objects in space and the class of K nearest neighbor is assigned to N. Fig represents the 4-NN model for classification where N will be assigned class A since out of 4 nearest objects 3 have Class A. In the paper [9], the authors

have introduced the use of phrases from email as a feature for K-NN classification. Unlike earlier approaches where most algorithms uses words as features for email classification. Phrases represent a sequence of words separated by white space if it occurs at least a minimum number of times. Such phrases are called Shingle by the authors.

Considered a document D containing the text, "Believe and act as if it were impossible to fail".

If such a document is used to collect features in terms of shingles of length 3 then the set of features S is given as:

S3 (D) = {"Believe and act", "and act as", "act as if", "as if it", "if it were", "it were impossible", "were impossible to", "impossible to fail".}The total number of shingles N of length 3 in D is 8, given by formula:

$$N = n-w+1 \quad (9)$$

Where n is the total number of words in D and w is the length of each shingle. The authors used the concept of resemblance as their similarity measure defined as in [9],

$$r_w(A, B) = \frac{|S_w(A) \cap S_w(B)|}{|S_w(A) \cup S_w(B)|} \quad (10)$$

Where S_w (A) and S_w (B) are w shingles of document A and B. Using the above resemblance formula K nearest neighbor can be find out for a new email, thus classifying it to the class shared by the most number of mails in K neighbours.

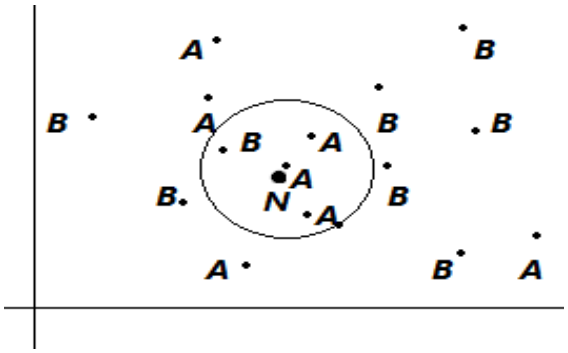

Fig. 2 K-NN algorithm for 4 nearest neighbour

Table 2 shows the summary of above mentioned algorithms.

Table 2. Summary of Algorithms

Algorithm	Dataset	Type of Learning	Method	Result
Bayesian Theorem	5175 emails of Author Account	Supervised Learning	Extracted Keywords	94.2%
Support Vector machine	6000 mails from Spam Assassin dataset	Supervised Learning	Bag of words optimized by GA algorithm	93%
K –Nearest Neighbour	2903 emails grouped into 27 folders.	Supervised Learning	Shingles-phrases of words	94% precision for spam detection

In our Dissertation, we focus on Naive Bayesian Theorem for analysis and enchantment in email classification.

III. IMPLEMENTATION DETAILS

The email consists of a message header and message body. Header includes sender and receiver address, subject, date and server address etc. Message body includes text data. In our implementation of the classification we will focus on message body only.

The Naïve Bayes classification is based on Baye's rule of condition probability. Entire email classification process is divided into three phases or steps.

A. First phase

In First phase the user creates the rule for classification. Rules are nothing, but the keywords/phrases that occur in mails for respective legitimate or spam mails. Then this rule is stored in database as a set of tokens for email classification.

B. Second Phase

In Second phase can be called as training phase. Here the classifier will be trained using a spam and legitimate emails manually by the user. Then with the help of algorithm the keywords are extracted from classified mails, and this keyword is stored as the set of token in database.

C. Third Phase

One the first and second phase are completed, we now move to next phase for classifying the emails by given algorithm, using this knowledge of tokens, the filter classifies every new incoming email. Here the probability of maximum keyword match is calculated and the status of a new email is confirmed as spam or ham mail, all its tokens are also calculated and then updated in the database. This self-learning functionality of our filter makes it unique among all other available spam filters. Even if the filter misclassifies any message, the user can rectify it and the spam filter would update its database accordingly. Below Figure 3 shows the block diagram.

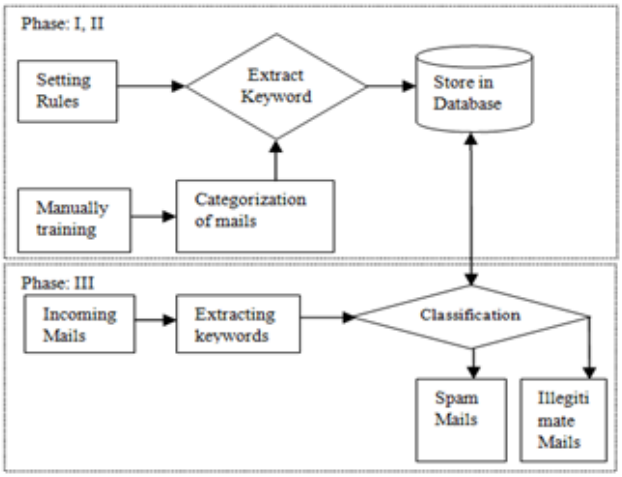

Fig. 3 Proposed spam filtering system with three phase technique.

D. Platform

To implement a system, we have used Visual Studio 2010 as front end, SQL Server 2008 as database for storing data and supported Operating System are WINDOWS XP & its above versions.

IV. RESULTS

A. Data Set

For our project we used total 12600 emails out of which 1250 legitimate messages and 11350 spam messages. The dataset message collection is, available on http://www.csmining.org/index.php/data.html.

B. Result

The accuracy of classifier dependent on several parameters such as number of mails considered during training phase, threshold value of each spam/legitimate emails, size of email etc. The evaluation measures which used for testing process in our research work could be defined as follows:

- True Positive (TP): no. of spam emails correctly classified
- True Negative (TN): no. of ham emails correctly classified.
- False Positive (FP): no. of spam emails classified as ham.
- False Negative (FN): no. of ham email classified as spam.
- Overall Accuracy (TP+TN) / (TP+TN+FP+FN): is percentage of prediction that is correct.
- Recall Rate TP/ (TP+FN): is percentage of positive label instance is predicted as positive.
- Precision TP/ (TP+FP): is percentage of positive prediction is correct.

Table 3. Performance results of Email classifier based on proposed algorithm

Email Dataset		Recall	Precision	Accuracy
Spam	Ham			
100	50	0.84	0.8	0.77
200	180	0.82	0.9	0.84
500	350	0.91	0.95	0.92
1000	600	0.97	0.95	0.95

The email dataset is the number of spam and ham/ legitimate email manually classified during the learning phase. Note the accuracy is also dependent on properly training the emails during learning phase as the spam and legitimate mails may differ from organization to organization or person to person. We also observed that when two different categories (spam/ legitimate) have many keywords in common, the classifier accuracy is low. With distinct keywords for each categories (spam/ legitimate) the overlapping is minimum which lead to increase in classifier accuracy above 0.9. The results also show that, if we have large training data the accuracy increases.

V. CONCLUSIONS

In this Dissertation, we consider the requirement of improving the efficiency of filtering techniques based on Naïve Bayesian, which is a good machine learning algorithm. The project we concentrate on text words from subject and

message body. The results show that our approach to classify emails is a reasonable and effective one. However, there are lots of enhancements to be done in future. The future enhancement includes working with other languages than English, classifying the emails depending on their header, the type of legitimate emails such as important, social network, personal etc, depending on attachments, check for malicious code in email, understanding the email text and so on.

ACKNOWLEDGMENT

The Author Savita Pundalik Teli would like to acknowledge the contribution of Mr. Prof. Santoshkumar Birdar for his valuable suggestions and guidance. His extremely successful professional life has been a strong motivating factors in pursuit my PG. He is not only a great advisor but also a caring mentor during my PG. His strong dedication, amazing energy, and generosity will continue to be the source of inspiration to me.

REFERENCES

[1] A. Dasgupta, P. Drineas, and B. Harb, "Feature Selection Methods for Text Classification", KDD'07, ACM, 2007.

[2] F. Sebastiani, "Text categorization", Alessandro Zanasi(ed.) Text Mining and its Applications, WIT Press, Southampton, UK, pp. 109-129, 2005.

[3] A. Khan, B. Baharudin, L. H. Lee, and K. Khan, "A Review of Machine Learning Algorithms for Text-Documents Classification", Journal of Advances Information Technology, vol. 1, 2010.

[4] F. Sebastiani, "Machine Learning in Automated Text Categorization", ACM 2002.

[5] D. Vira, P. Raja, and S. Gada, "An Approach to Email Classification using Bayesian Theorem," GJCST. (USA), vol. 12, Issue 13, ver. 1.0, 2012.

[6] J. Han, M. Kamber, and J. Pei, (2011, July 6) "Data Mining Concepts and Techniques" (3rd ed.). The Morgan Kaufmann Series in Data Management Systems.

[7] A. Patra, K.Mandal, S.Roy, S.Sau and S. Kunar, "An Efficient Spam Filtering Techniques for Email Account," American Journal of Engineering Research, vol. 02, Issue 10, pp. 63-73, 2013

[8] F. Temitayo, O. Stephen, and A. Abimbola, "Hybrid GA-SVM for Efficient Feature Selection in Email Classification," IISTE, vol. 3, no. 3, 2012.

[9] M. Chang, C.K Poon, "Using Phrases as Features In Email Classification," ELSEVIER,vol.82, 2009, pp. 1036-1045.

[10] V. P. Deshpande, R. F. Erbacher, and C. Harris, "An Evaluation of Naive Bayesian Anti-Spam Filtering Techniques" Proc. of IEEE, 2007.

[11] S. Youn and D. McLeod, "Efficient Spam e-mail Filtering using Adaptive Ontology," International Conference on Information Technology (ITNG'07), pp.249-254, 2007.

[12] S. Hershkop, and J. Stolfo, "Combining e-mail models for false positive reduction," Proc of KDD'05 of ACM. Chicago. [s.n.], pp. 98—107, 2005.

[13] S. J. Delany, P. Cunningham, and B. Smyth, "ECUE: A spam filter that uses machine learning to track concept drift," Proceedings of the 17th European Conference on Artificial Intelligence (PAIS stream), pp.627-631, 2006.

[14] Y. Yang, "An Evolution of statistical Approaches to Text Categorization," Information Retrieval 1, 69-90 1999.

[15] Li, Y. H, and A. K. Jain, "Classification of text documents". The Computer Journal, 537–546. 1998.

[16] L. S. Larkey, and W. B. Croft, "Combining classifiers in text categorization". In Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (Zurich, CH, 1996), pp. 289–297 1996.

[17] O. Zaiane, and M. Antonie, "Text Document Categorization by Term Association", Proceedings of ICDM 2002, IEEE,, pp.19-26 2002.

[18] S. Buddeewong1, and W. Kreesuradej, "A New Association Rule-Based Text Classifier Algorithm", Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence, 2005.

[19] S.M.Kamruzzaman ,and M.R.Chowdhury,"Text Categorization using Association Rule and Naive Bayes Classifier" CoRR, 2010.

[20] K. Aas, and L. Eikvil, "Text Categorization: A Survey" Report No. 941. ISBN 82-539-0425-8. , June, 1999.

[21] K. Tretyakov, Machine Learning Techniques in Spam Filtering, Technical report, Institute of Computer Science, University of Tartu, 2004.

[22] K. A. Vidhya, and G. Aghila, "A Survey of Naive Bayes Machine Learning approach in Text Document Classification", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, 2010.

[23] A. McCallum, and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification". AAAI/ ICML -98 Workshop on Learning for Text Categorization.

[24] K. Sang- Bum, et al, "Some Effective Techniques for Naive Bayes Text Classification "IEEE Transactions on Knowledge and Data Engineering, Vol. 18, November 2006.

[25] S. Yirong, and J. Jing," Improving the Performance of Naive Bayes for Text Classification"CS224N Spring 2003.

[26] M. J. Pazzani, "Searching for dependencies in Bayesian classifiers" Proceedings of the Fifth Int. workshop on AI and, Statistics. Pearl, 1988.

