Journal of Scientific & Industrial Research
Vol. 73, July 2014, pp. 437-442

Latent Semantic Indexing Based SVM Model for Email Spam Classification

Karthika Renuka D[1*] and Visalakshi P[2]

[1*]Department of Information Technology, PSG College of Technology, India
[2]Department of ECE, PSG College of Technology, India

Received 01 July 2013; revised 18 October 2013; accepted 26 February 2014

Internet plays a drastic role in part of communication nowadays but in e-mail, spam is the major problem. Email spam is unwanted, inappropriate or no longer wanted mails also known as junk email. To eliminate these spam mails, spam filtering methods are implemented using classification algorithms. Among various algorithms, Support Vector Machine (SVM) is used as an effective classifier for spam classification by various researchers. But, the accuracy level is not up to notable level so further. To improve the accuracy, Latent Semantic Indexing (LSI) is used as feature extraction method to select the suitable feature space. The hybrid model of spam mail classification can provide the effective results. The Ling spam email corpus is used as datasets for the experimentation. The performance of the system is evaluated using measures such as recall, precision and overall accuracy.

Keywords: Email, Ling spam dataset, Feature extraction, LSI, Spam, Spam classification, SVM

Introduction

Electronic mail is the simplest and most effective communication device for transmitting information between users. Electronic mail (E-mail) is an electronic message system which communicates messages throughout the computer network.

Email Spam: Email spam[1] or junk e-mail is one of the significant challenges facing the modern Internet, bringing monetary loss to companies and frustrating individual users. It is transmitting unnecessary email messages with commercial content to haphazard set of recipients. Spam is completely filling the Internet with several versions of the identical message, with the aim of thrusting the message on individuals who would not have received it otherwise. Spam is unnecessary and useless consumption of time, storage space and communication bandwidth.

Spam Classification: Spam classification is the function of filtering spam email from inbox and shifting them to the junk email folder. Classification is the task of disintegrating spam and ham mails. Classification faces the challenge of recognizing the set of categories a fresh inspection belongs, according to the training set of data having observations having category membership which is familiar. The term “classifier” also represents the mathematical function, executed by a classification algorithm, which maps input data to a category.

Machine Learning Algorithm: Knowledge engineering and machine learning[2, 3] are the two common methods employed in e-mail filtering. In knowledge engineering technique a set of rules has to be made specific on the basis of which emails are classified as spam or ham. Machine learning method is far superior to knowledge engineering technique as it does not necessitate demarcating any rules. On the other hand, the training samples is a set of pre classified e-mail messages. A definite algorithm is thereafter employed to learn the classification rules from these e-mail messages. Of late, Machine learning for spam classification is a significant research problem. Spam classification categorizes the mails into spam and non-spam mails by designing a spam classifier.

In this paper, we have proposed an email spam classification technique using Latent Semantic Indexing Based SVM Model. At first, the input dataset is given to pre-processing step which removes the stop words and punctuation marks so that keywords that are more relevant are obtained. The extracted keywords are then given to the feature extraction in which, Term Frequency (TF) and Inverse Document Frequency (IDF) are computed for the keywords taken from the pre-processing steps. After the feature matrix formation, the suitable dimension for the better classification is found out

*Author for correspondence
Email: karthirenu@gmail.com

using LSI model that maps the feature space to LSI space using correlation-based analysis. Finally, the LSI space is given to SVM algorithm which trains based on the patterns given in the training LSI space. In testing phase, the input mails represented in LSI space is classified as Spam or Ham based on the optimal hyperplane generated in SVM training [4]. By using these steps, classification of spam and ham mails has been done effectively.

The outline of this paper is follows: Section 2 describes objective and scope. The proposed latent semantic indexing based SVM model for email spam classification is detailed in Section 3. The experimental results and performance evaluation discussion is provided in Section 4. Finally, the conclusions are summed up in Section 5.

Objective and Scope of this Paper

Objective: The proposed system is mainly for classifying spam and ham mails. Commercial emails which are unwanted are also known as spam. To get rid of those spam mails or messages classifier is used. Machine learning techniques [5, 6] are used for this kind of classification but it's not up to cent percent so by combining with better feature extraction technique which presents more optimal solution it leads to more accuracy in classification.

Scope: Support Vector Machine constructs hyper-plane in high dimensional space where larger the margin will lower the generalization error. Support vector machines (SVM) are a set of related supervised learning methods used for classification and regression. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. Latent Semantic Indexing (LSI) [7, 8] is used for feature extraction with semantic structure as the conceptual searches. LSI tries to overcome the problems of lexical matching by using statistically derived conceptual indices instead of individual words for retrieval. By combining SVM with LSI yields better accuracy in classification. It reduces the misclassification of spam emails as non-spam and vice-versa.

Proposed Latent Semantic Indexing Based SVM Model for Email Spam Classification

This paper presents spam email classification system by effectively integrating SVM model with LSI [9]. The ultimate aim of this work is to improve the performance of the SVM model by finding the suitable feature space to learn the SVM in better way. The suitable and effective feature space can split the margin while the SVM is trained with the training dataset[10, 11]. With the intention of this, both the models are combined to provide an effective system for spam mail classification. This chapter contain three sub-sections in which the design, basics of SVM and LSI and the proposed module is explained in a detailed way.

Design of the Proposed Spam Classification Model

The design of proposed system is explained in figure 1. The objective of classifying the spam and ham mails was done with the subsequent steps, such as, dataset preparation, finding right tool and environment for implementation and the implementation of the proposed system. For this e-mail classification machine learning technique, Support Vector Machine is used in combination with feature extraction technique and Latent Semantic Indexing[12, 13]. The process of proposed email spam classification technique can be described as following significant stages:

Module Description

The detailed steps of the proposed system modules are explained in this sub-section. Initially, the input dataset is given to pre-processing step which removes the stop words and punctuation marks and then, root words are obtained. The extracted root words are then given to find TF and IDF for the root words. After the feature matrix formation, the suitable dimension is found out using LSI model because the feature space is large. Finally, SVM algorithm is trained with the

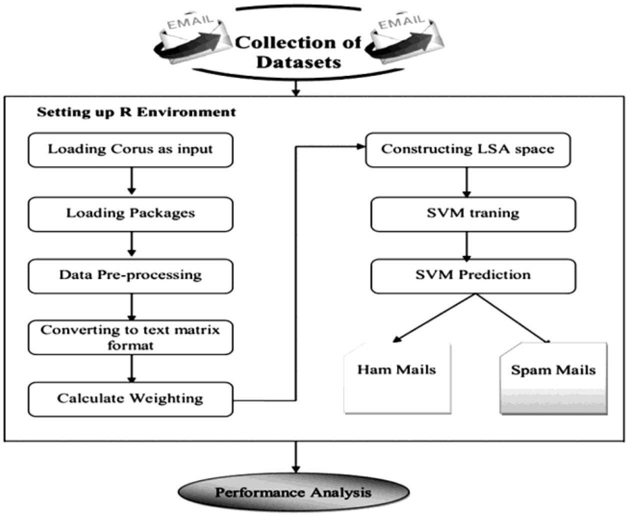

Fig. 1—Design of the proposed system

LSI space[14, 15, 16]. In testing phase, the input mails represented in LSI space is classified as Spam or Ham based on the trained SVM.

Data Pre-processing

Data pre-processing is an important step for classification which is one of the process in the data mining. If there is much irrelevant and redundant information present or noisy and unreliable data, then knowledge discovery during the training phase is more difficult. Data preparation and filtering steps can take considerable amount of processing time. Data pre-processing includes cleaning, normalization, transformation, feature extraction and selection, etc. The product of data pre-processing is the final training set.

In this proposed system, the lingspam dataset will be given as input and the preprocessing steps, like, stop words removal, punctuation and numbers removal are performed initially. Then, the root words are found out for all the terms presented in the text document.

Feature vector computation

Feature vector computation is the major step in all the classification system because the performance of the system is completely based on the features extracted from the input data. Accordingly, we have extracted TF and IDF for the terms extracted from the data preprocessing steps so that the feature vector is formed for all the mails utilized in the datasets.

Constructing LSA space

After extracting the feature vectors, the features matrix of input data is given to LSA model which will reduce the dimension of the features matrix by mapping the feature to LSA space. Latent Semantic Indexing (LSI) is a statistical method that obtains a statistical correlation between all terms and documents in a corpus, in an effort to tackle the issues present in lexical matching. Mainly, LSI method is used as dimensionality reduction method which can preserve the useful feature space by reducing unwanted feature space. The feature spaces that are further taken from LSI have good correlation with the predicted variable so that the performance in classifying the mails can be improved. The LSI model used in the proposed system model is taken from R environment.

For the execution of LSA algorithm in R environment, first, a textmatrix is constructed from the input corpus and textmatrix can weighted using one of the various weighting schemes provided. Then, the singular-value decomposition is executed over this textmatrix and the resulting partial matrices are truncated and returned by lsa(). The number of dimension to keep can be set using various recommender routines (e.g., dimcalc_kaiser ()). The resulting latent-semantic space can be converted back to text matrix format using as.textmatrix ().In case that additional documents are to be folded into the existing latent-semantic space, again a new textmatrix is constructed using textmatrix() re-using the vocabulary of the first. Again the resulting textmatrix can be weighted (eventually re-using the global weights of the first textmatrix). Using fold_in(), the resulting textmatrix can be mapped into the existing latent-semantic space, thereby re-using the truncated left-sided and the diagonal partial matrices of the SVD.

SVM Processing

Once the input data is converted to LSI space, the reduced matrix is given to Support Vector Machine (SVM) for training that finds the optimal hyperplane to further separate the input data into two different sets. A Support Vector Machine (SVM) is one of the data mining algorithms which perform classification by constructing an N-dimensional hyperplane that optimally separates the data into two categories. Here, the aim of SVM modeling is to find the optimal hyperplane that separates group of vectors in such a way that cases with one category of the predicted variable are on one side of the plane and cases with the other category are on the other size of the plane. The vectors near the hyperplane are the support vectors. The figure 2 shows the explanation of the SVM process.

In R environment, the implementation of SVM is done by using e1071 package. The package or the interface to the package has to be imported and installed in the zipped format. Radial Basis Function (RBF) can be denoted as kernel="rbfdot". There are many kernel functions according to the usage it can be used. Then predict () is the function which performs prediction from the model build on the training dataset.

Training of SVM

SVM training is the first phase in classification process. When the particular file has been given as input with numeric values, it has to be separated for

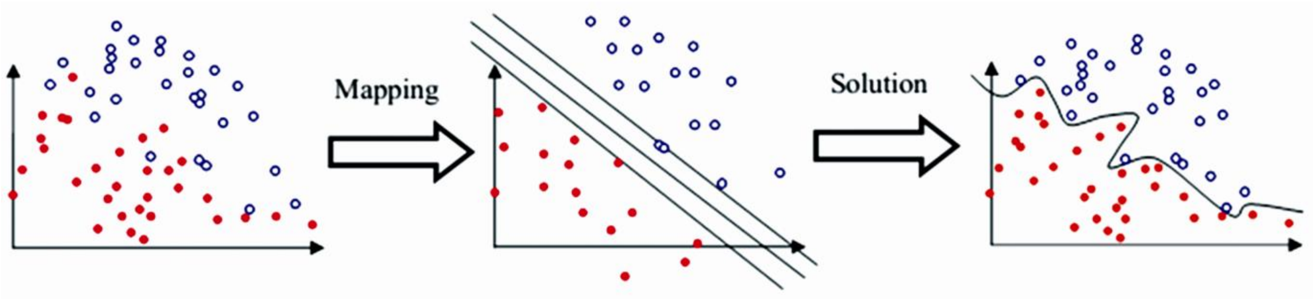

Fig. 2—Process of SVM algorithm

both training and testing dataset. Separate subset has been created for training dataset and classes. Here, the classes are two levels: spam and non-spam. First of all, the extracted feature values have been selected without their classes into particular subset. Then, the classes alone selected as the labels for that particular subset. Then, for training, we have to select data and classes from that subset to build SVM model which helps to predict classes for other similar datasets. svm() is the function used to build the model. To display the model or results, print() function is used.

The R interface to libsvm in package e1071, svm(), was designed to be as intuitive as possible. Models are fitted and new data are predicted as usual, and both the vector/matrix and the formula interface are implemented. As expected for R's statistical functions, the engine tries to be smart about the mode to be chosen, using the dependent variable's type (y): if y is a factor, the engine switches to classification mode, otherwise, it behaves as a regression machine; if y is omitted, the engine assumes a novelty detection task.

Testing of SVM

For testing the SVM, LSI space of testing data can be given to the trained SVM which find the space in the feature plane based on the optimal hyerplane generated from the training data. The space obtained provide the whether the test data belongs to Spam or Ham. In R environment, from the model build from svm() function, we can use it for predictions among other dataset. Predict() is the function used for predicting the classes for testing dataset. It can be either spam or non-spam (i.e., ham).

Results and Discussion

This section presents the results obtained from the proposed experimentation and its detailed discussion about the results. The proposed approach of email spam classification is experimented with the publically available datasets and the result is evaluated with the precision, recall and accuracy.

Dataset Description

Here, we use Ling-spam dataset for the experiment, and there are two classes in this dataset including spam and legitimate mail. Ling-Spam is composed of 481 spam mail and 2412 legitimate mail from the collection of researcher Androutsopoulos[17]. In the Ling-spam dataset, 10 sub directories (part1 ...part10) have been separated randomly, and each one of the 10 subdirectories contains both spam and legitimate mail. In each repetition, 2 parts were reserved for testing and the other 8 were used for training. Files whose names have the form spmsg*.txt are spam mail and all other files are legitimate mail.

Evaluation Metrics

The results are evaluated using important evaluation matrices called precision, recall and accuracy. The evaluation of proposed email spam classification technique[18, 19, 20] in lingspam corpus are carried out using the following metrics as suggested by below equations,

$$\Pr ecision = \frac{TP}{TP + FP}$$

$$\mathrm{Re}\,call = \frac{TP}{TP + FN}$$

$$Accuracy = \frac{TN + TP}{TN + FP + FN + FP}$$

Where, TP stands for True Positive, TN stands for True Negative, FN stands for False Negative and FP stands for False Positive . *Precision* is the proportion of true positives that are correctly identified by a proposed method. It shows how good the test is at detecting a road. *Recall* is the proportion of the true negatives correctly identified by proposed method. It suggests how good the test is at identifying

Table 1—Input File for SVM

Terms Docs	Access	Cancel	Claim	Entity	Related	Yield	Classes
D1	898.13	561.43	131.89	290.12	543.56	439.17	Spam
D2	289.30	579.12	339.00	135.56	147.53	593.05	Ham
D3	353.34	559.06	611.33	451.11	635.41	103.87	Ham
D4	526.02	213.00	316.10	324.06	150.09	604.63	Spam
D5	168.63	412.22	834.53	590.43	313.45	821.33	Ham
D6	322.54	944.53	641.23	274.78	189.31	178.95	Ham

normal (negative) condition. *Accuracy* is the proportion of true results, either true positive or true negative, in a population.

Experimental Results

The lingspam corpus will be given as the input for data pre-processing. The sample output for the data after pre-processing. It has to be converted to the textmatrix format and then it will calculate weighting using term frequency and inverse document frequency. Then the particular result will be converted and will be used for constructing the LSA space [21, 22]. The textmatrix format is nothing but the term-by-document which has to be arranged into rows and columns representing the numeric values of features extracted. Each feature from the documents has to be extracted and for all those features it has weighting. In this pre-process stage numbers, punctuation and all other stop words has been removed. Then the output of semantic space constructed. The semantic structure gives similarity or association of each term in all documents. While using tf-idf also we can compute the cosine similarity but it is not much effective than the cosine similarity computed using LSI. It provides the conceptual searches among the documents. It contains numeric values which gives similarity between the terms in the whole corpus. Then the LSA space will be converted to the textmatrix format and then written to the file to a specific format. It will be given as input for SVM in following Table 1.

After training SVM, the particular model will be build. Then the prediction will be done using the model in the SVM training phase. From the experimental validation, we interpret that the performance level of SVM using LSI has been increased when comparing to SVM using tf-idf. In the first method, we have to calculate the term frequency by extracting the terms from the documents and then calculating the inverse document frequency among the documents then by constructing the tf-idf matrix we are classifying the mails using SVM classifier[23]. In the second method, we have to remove stop words and then creating LSA spaces which are also known as semantic structure by using that values in combination with SVM classifier the mails has been classified. By implementing both methods from the results we yield, the second method performs better than the other and gives better accuracy.

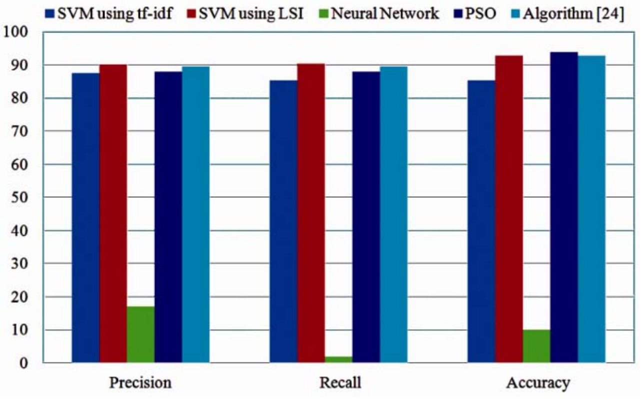

Fig. 3—Comparative Evaluation

Comparative Analysis

The comparative analysis of the proposed system model for email spam classification is discussed in this section. At first, the performance is compared with and without dimensionality reduction technique. From the performance, the reduced dimension with the help of LSI shows the better results as compared with TF and IDF-based scheme (without LSI). The SVM (TF-IDF) achieve 85% accuracy while the SVM (LSI) achieved 93% accuracy. The performance improvement of the proposed system model over the SVM (TF-IDF) is 8%. Figure 3 shows the performance of the proposed system with respect to other classification system. From the figure 3, the existing system[24] and the PSO-based system achieved 92.8 %, 92.9 % as accuracy. As compared with PSO-based system, the performance improvement of the proposed system model over the PSO-based system is 0.1%, As compared with existing system[24], the performance improvement of the proposed system model over the SVM (TF-IDF) is 0.1%.

Conclusion

In this paper, we have proposed an email spam classification technique using Latent Semantic Indexing Based SVM Model. Here, TF and IDF-based features were extracted and feature selection was done with LSI as feature reduction technique. The proposed hybrid system classifies the spam and ham mails using SVM classifier. The experimentation was performed with the lingspam database and the results are compared with the existing systems like, neural network, PSO as reduction technique and the TF-IDF-based system. The performance metrics such as precision, recall and f-measure are used as evaluation metrics to compare the performance and, the proposed hybrid model ensures the better performance as compared with the existing algorithm from the results. In future, the optimization algorithm like, Group Search Optimizer and the hybrid optimization algorithm are used for the selecting the features effectively.

References

1 Saad O, Darwish A & Faraj R, A Survey of Machine Learning techniques for Spam Filtering, *Int J Comp Sci and Network Security(IJCSNS),* 12 (2012).

2 Parimala R & Nallaswamy R, A Study on Enhancing Classification Accuracy of Spam E-mail Dataset, *Int J Comp Trends and Tech*, 2 (2011).

3 Drucker H, Wu D & Vapnik V N, Support Vector Machines for Spam Categorization, *IEEE trans Neural Net*, 10 (1999).

4 Yang Q & Li F, Support Vector Machine for Customized Email Filtering based on Improving Latent Semantic Indexing, *Int Conf ML and Cybernetics*, 6 (2005) 3787 - 3791.

5 Awad W A & ELseuofi S M, Machine Learning Methods for Spam E-mail Classification, *Int J Comp Sci & Info Tech (IJCSIT)*, 3 (2011).

6 Santos I, Laorden C et al, Enhanced Topic-based Vector Space Model for Semantics-Aware Spam Filtering, *Expert Sys with App*, 39 (2012) 437–444

7 Landauer T & Psotka J, Simulating Text Understanding for Educational Applications with Latent Semantic Analysis, *Int Conf Interactive Learning Envi*, 8 (2000) 73-86.

8 Deerwester S, Dumais S et al, Indexing by Latent Semantic Analysis, *J American Soc for Info Sci*, 41 (1990) 391-407.

9 Yang J, Liu Y et al, A New Feature Selection Algorithm based on Binomial Hypothesis Testing for Spam Filtering, *Knowledge-Based Systems,* 24 (2011) 904–914.

10 Pérez-Díaz N, Ruano-Ordás D et al, Rough sets for Spam Fltering: Selecting Appropriate Decision Rules for Boundary E-mail Classification, *Applied Soft Comp*, 12 (2012) 3671–3682.

11 Ying K, Lin S et al, An Ensemble Approach Applied to Classify Spam E-mails, *Expert Sys with App*, 37 (2010) 2197–2201.

12 Hsu C, Chang C & Lin C, A Practical Guide to Support Vector Classification, *Research article from CiteSeerX rep*, (2008).

13 Wild F & Stahl C, Investigating Unstructured Texts with Latent Semantic Analysis, *Lenz, Decker (Eds.): Adv in Data Analysis*, Springer, Part V (2007).

14 Smadja F & Tumblin H, Automatic Spam Detection as a Text Classification Task, *Workshop on Operational Text Classification Sys*, (2002).

15 Gee K, Text Classification Using Latent Semantic Indexing, *Master's thesis:Univ of Texas at Arlington*, (2001).

16 Jiang J, Using Latent Semantic Indexing for Data Mining, *Dept of Comp Sci*, *Univ of Tennessee, (1997).*

17 Androutsopoulos I, Koutsias J et al, An evaluation of Naïve Bayesian anti-spam filtering, *Workshop on ML in the New Info Age,* 11th ECML 2000, Barcelona, Spain, (2000) 9-17.

18 Islam M & Zhou W, Architecture of Adaptive Spam Filtering based on ML Algorithms, *ICA3PP 2007, Lec Notes in Comp Sci*, 4494 (2007) 458–69.

19 Islam R, Chowdhury M & Zhou W, An Analysis of Spam and its Classification Techniques based on Statistical Learning Algorithms, *Tech report TRC 05/06, Deakin Univ, Aus; 2005a.*

20 Islam R, Chowdhury M & Zhou W, An Innovative Spam Filtering Model based on Support Vector Machine, *IEEE Int conf on Intelligent agents, Web tech and Internet comm,* 2, 28–30 (2005) 348–53.

21 Islam M & Chowdhury M, Spam filtering using ML algorithms,*IADIS Int conf WWW/Internet. USA: Int Asso for Dev of the Info Society Press; (2005) 419–26.*

22 Baskaran S, Content based email classification system by applying conceptual maps, *Int Conf Intelligent Agent & Multi-Agent Sys*, (2009).

23 Zhang W & Gao F, An Improvement to Naive Bayes for Text Classification, *Procedia Engg 15 (2011) 2160-2164.*

24 Renuka D, Karthika & Visalakshi P, Blending Firefly and Bayes Classifier for Email Spam Classification, *Int Review on Comp & Software*, 8 (2013) 2168- 2177.