Effect of Various Kernels and Feature Selection Methods on SVM Performance for Detecting Email Spams

Shrawan Kumar Trivedi
Information Systems
Indian Institute of Management
Prabandh Shikhar,
Rau, Indore – 453 556, India

Shubhamoy Dey
Information Systems
Indian Institute of Management
Prabandh Shikhar
Rau, Indore –453 556, India

ABSTRACT

This Research presents the effects of interaction between various Kernel functions and different Feature Selection Techniques for improving the learning capability of Support Vector Machine (SVM) in detecting email spams. The interaction of four Kernel functions of SVM i.e. "Normalised Polynomial Kernel (NP)", "Polynomial Kernel (PK)", "Radial Basis Function Kernel (RBF)", and "Pearson VII Function-Based Universal Kernel (PUK)" with three feature selection techniques i.e. "Gain Ratio (GR)", "Chi-Squared (χ^2), and "Latent Semantic Indexing (LSI)" have been tested on the "Enron Email Data Set". The results reveal some interesting facts regarding the variation of the performance of Kernel functions with the number of features (or dimensions) in the data. NP performs the best across a wide range of dimensionality, for all the feature selection techniques tested. PUK kernel works well with low dimensional data and is the second best in performance (after NP), but shows poor performance for high dimensional data. Latent Semantic Indexing (LSI) appears to be the best amongst all the tested feature selection techniques. However, for high dimensional data, all the feature selection techniques perform almost equally well.

General Terms

Spam classification, Complex features, High and Low dimensional features.

Keywords

Support Vector Machine (SVM), Kernel functions, Feature selection methods.

1. INTRODUCTION

In today's automated world, Emails are a common and useful medium of communication. Conversely, spam, also called unsolicited bulk Email, is a nuisance in Email communication. A recent study indicates that more than 70% of commercial Emails are SPAM [1]. These unsolicited Emails can be transmitted as some money making advertisement or some content that may even conceal malicious code [2]. The growing volume of SPAM leads to serious hitches such as filling users' mailbox with unwanted Emails engulfing useful Emails, unnecessarily consuming storage space and bandwidth and wasting time in segregating them [3].

In recent years, SPAM classification has become a challenging area of research due to the complexity added by spammers in SPAM words. Complexity can be defined as attacks associated with SPAM features that make a word difficult to understand. Some spam attacks, like Tokenisation (Splitting or modifying the feature such as 'free' written as f r 3 3) and Obfuscation (hides feature from adding HTML or some other codes such as 'free' coded as fr&#101xe or FR3E), alter the information of particular feature [2, 4].

To tackle these problems, a number of Machine Learning (ML) approaches have been discussed in the literature. Some of these approaches have found a prominent place in the SPAM classification domain. Amongst all ML techniques, Support Vector Machines (SVM) and neural network based supervised learning models have proven their worth. SVM uses the concept of "Statistical Learning Theory" proposed by Vapnik [5], which propounds that, it is important to maintain right balance between achieved accuracy of the training set and the strength of classifier (i.e. ability of the classifier to learn training examples without error).

The foremost benefit of SVM is its strength in classifying high dimensional data with good accuracy. It works on the principle of finding a Maximum Margin Plane by dividing data from different classes. To find the Maximum Margin Plane by using this principle, a variety Kernel functions are used. The search of most appropriate Kernel function for implementation of SVM for a specific application is a challenging task because, to obtain the accurate classification, the parameters of the Kernel functions need to be 'fine-tuned'. A good choice of the Kernel function itself is also important for every specific application of SVM.

To improve the prediction accuracy of classifiers, various feature selection techniques have been proposed in literature. This research examines three feature selection techniques: Gain Ratio (GR), Chi-Squared (χ^2), and Latent Semantic Indexing (LSI), for extracting the most informative features.

The later sections are structured in the following way: Section 2 presents Related Work in this area, Section 3 describes Function Based Classifiers, Section 4 gives an overview of Feature Selection Techniques, Section 5 presents the Experiments and Evaluations, Section 6 carries out Comparative Analysis of the results, and last Section 7 Concludes the paper.

2. RELATED WORK

A great deal of significant work has been reported in the area of classification. However, this work restricts its attention to the SPAM classification domain which has become a challenging area research in recent years. This paper presents in Table I, a summary of some existing work where classifiers have been tested on spam datasets.

TABLE I. LITERATURE REVIEW

Author(s) & Year	Model Used	Data Source / Data Set	Accuracy (%)
Harris Drucker, Donghui Wu, and Vladimir N. Vapnik (1999) [6]	SVM, Boosting Tree	AT&T staff member & AT&T technical staff Data Set	NA
Matthew Woitaszek, and Muhammad Shaaban Roy Czernikowski (2003) [7]	SVM	RIT's ITSs help desk for spam mix with Publically available Non Spam Emails	96.69
Le Zhang, Jingbo Zhu, and Tianshun Yao (2004) [8]	SVM, Boosting, ME, Memory based, NB	PU1,LingSpam, SA, and ZH1	F-value – 95 to 97.5
Victor Cheng, and C.H. Li (2006) [9]	SVM, NB	Different user from public domain	73-96
D. Sculley, and Gabriel M. Wachman (2007) [10]	Relaxed Online SVMs	Spam-Assassin	93.1-94.9
Ming-wei Chang, Wen-tau Yih, and Christopher Meek (2008) [11]	partitioned logistic regression	A non-public Hotmail dataset, and 2005 and 2006 TREC Spam	AUC value, 58.8-96.2
W.A. Awad, and S.M. Elseuofi (2011) [12]	Bayesian, KNN, ANN, SVM, AIS and RS	Spam-Assassin	97.42-99.46
R. Kishore Kumar, G. Poonkuzhali, and P. Sudhakar (2012) [13]	Several classifiers including SVM	UCI machine learning and created in. Hewlett-Packard Labs. repository	Up to 99

3. FUNCTION BASED CLASSIFIERS:

3.1. Support Vector Machine (SVM)

Support Vector Machine (SVM) is popular amongst various machine learning classifiers. It uses the concept of Statistical Learning Theory and Structural Minimization Principle [14]. Due to its appealing capability to handle high dimensional data by the use of various Kernel functions, it is one of the most popular techniques in the literature.

The basic idea behind SVM algorithm is to separate classes (i.e. positive and negative) with maximum margin created by hyper planes. Let us consider a training sample $X = \{x^i, y^i\}$, where $x^i \in R^n$ and $y^i \in \{+1, -1\}$ which is defined as the corresponding class for i^{th} training sample. Here $+1$ represents SPAM Emails and -1 represents legitimate (HAM) Emails. Output of the classifier is given as:-

$$y = w.x - b \qquad (1)$$

Where, y is the final result of classifier, w represents normal weights corresponding to those in the feature vector x, and b is the bias parameter that will be determined by training process. The margin between classes can be maximized by following optimization function:-

$$\text{minimize} \quad \tfrac{1}{2}\|w\|^2 \qquad (2)$$

$$\text{subject to} \quad y^i\left(w.x - b\right) \geq 1, \forall i \qquad (3)$$

3.2. Kernel functions

On many occasions, SVMs are unable to find a linear hyper-plane that can separate the input data into classes. This problem can be tackled by transforming the input data that exists in high dimensional space by using some non-linear transformation function. By this process, the input data can be separated out in such a way that linear separable hyper planes can be found in that transformed space. However, due to the high dimensionality of the feature space, computation of inner products of two transformed data vectors would be practically unfeasible. This problem is tackled by the use of "Kernel Functions" that can be used in place of the inner product of two transformed data vectors in feature space. Effective use of Kernel functions can significantly reduce computational effort to making the operation feasible.

3.3. Kernel Selection

A good choice of Kernel function is very important for effective SVM based classification. An appropriate Kernel function provides learning capability to Support Vector Machine (SVM). A number of Kernels have been proposed in the literature. In this paper, four types of Kernel Functions will be used for experimentation. These are displayed in Table II.

TABLE II. KERNEL FUNCTIONS

Type of Kernels	**Full Name**	**Functions**
NP	Normalised Polynomial Kernel	$K_r\left(x^i, y^j\right) = \frac{\left(x_T^i . x^j + 1\right)^P}{sqrt\left(x_{T+1}^i + \mathrm{x}_{T+1}^j\right)}$
PK	Polynomial Kernel	$K_r\left(x^i, y^j\right) = \left(x_T^i . x^j + 1\right)^P$
PUK	Pearson VII function-based universal kernel	$K_r\left(x^i, y^j\right) = \frac{1}{\left[1 + \left(\frac{2\sqrt{\left\|x^i - x^j\right\|^2}\sqrt{2^{\left(\frac{1}{\omega}\right)} - 1}}{\sigma}\right)^2\right]^{\omega}}$
RBF	Radial Basis Function kernel	$K_r\left(x^i, y^j\right) = \exp\left(-\gamma\left\|x^i - x^j\right\|^2\right)$

4. FEATURE SELECTION METHODS

Various feature selection techniques have been proposed in the literature of Machine Learning. These techniques help to select relevant / most informative features from the feature sets and discard those features that seem irrelevant or redundant. In this research, three feature selection techniques have been considered.

4.1. Gain Ratio (GR)

This technique is an extension of Information Gain (IG). The weakness of IG was bias towards selection of features that contain higher numerical value even when they carry very little information.

$$IG = H(Y) - H\left(\tfrac{Y}{X}\right) = H(X) - H\left(\tfrac{Y}{X}\right) \qquad (4)$$

To compensate the bias of IG, Gain Ratio (GR) is used which is a type of non-symmetrical measurement [15].

$$GR = \tfrac{IG}{H(X)} \qquad (5)$$

From "(5)", when variable Y is to be predicted, IG will be normalised by dividing by the entropy of X . The normalisation process makes GR values lie between 0 and 1. When the GR value is 1, the information in X will completely predict Y , and if this is 0, then X and Y will have no relation with each other. Gain Ratio (GR) differs from IG by favouring features with lower numerical value.

4.2. Chi-Squared (χ^2)

This is a well known and commonly used technique for selecting features [16]. Chi-Squared (χ^2) method provides valuable features from the feature space with respect to the class by analysing value of chi-square statistics. This method tests initial hypothesis H_0 which makes an assumption, that "two features will be dissimilar".

$$\chi^2 = \sum\nolimits_{i=1}^{r}\sum\nolimits_{j=1}^{c}\left(\tfrac{O^{ij}-E^{ij}}{E^{ij}}\right)^2 \qquad (6)$$

Where, the notations O^{ij} is the observed frequency and E^{ij} is expected frequency, justified by the Null hypothesis. Higher value of the χ^2 will give significant evidence against the initial hypothesis H_0.

4.3. Latent Semantic Indexing (LSI)

This method attempts to investigate a low dimensional subspace by using association of words and documents. It uses Singular Value Decomposition (SVD) method for investigating such subspaces. Let us consider X_W^D is Word-Document Matrix corresponding to a vector space model. This matrix X_W^D can be decomposed by the product of three matrices.

$$X_W^D = WSD^W \qquad (7)$$

Where W and D are orthogonal matrices, and S is diagonal matrix that have diagonal components corresponds to decreasing order of singular value. This technique approximates the Word-Document Matrix (X_W^D) by use of bases which will correspond to higher value. Let us consider S' is a matrix whose all higher diagonal elements are put to zero. Hence, Term Document Matrix (X_W^D) can be represented

$$X_W^{D'} = WS'D^W \qquad (8)$$

From equation (8), matrix $X_W^{D'}$ indicates an optimal approximated value of matrix X_W^D, which is defined in terms of the mean-square error. LSI assumes that word of documents carries an essential Latent Semantic structure. It uses the idea that, such structure can be revealed by finding less singular value in S . It is proved by previous studies, that this method has shown performance improvement in Information Retrieval (IR) applications [17, 18, 19].

5. EXPERIMENTS AND EVALUATIONS

5.1. The Data Set

The Enron Email dataset was being selected for this research. From all the versions of the Enron dataset, this research included Enron versions 3, 4, 5 and 6 and then constructed dataset containing 6000 HAM and 6000 SPAM files by random selection [2]. The rationale of selecting these versions is that attacks (such as Tokenisation, Obfuscation etc.) present in the words of these Email files make them relatively more complex from the point of view of classification.

5.2. Pre-Processing of Data

Email files can be represented as vectors of features a_i^k i.e. the weight of a word i belongs to document k [20]. These vectors will be combined for a collection of text documents to generate Word-Document matrix. Resultant matrix will be large and sparse in nature due to the large number of documents used for classification. This problem can be well tackled by "Dimensionality Reduction" technique employed before the classification step and are done by the "Feature Selection" or "Feature Extraction" procedure. Dimensionality of the matrix is also reduced by "STOP WORD" (some words carry no information such as pronouns, prepositions, and conjunction) elimination [20] and "LEMMATISATION" (grouping words that carry same information such as "Improve, Improved and Improving").

5.3. Feature Selection

Feature selection is performed just after dimensionality reduction of the word-document matrix. Three feature selection methods, i.e. "Gain Ratio (GR)", "Chi-Squared (χ^2), and "Latent Semantic Indexing (LSI)", have been used in this research. For the purpose of the experiments, 1358 top ranked features were generated using each technique. The classifiers were then tested on those features.

5.4. Classifiers

Java and Matlab software on Window 7 operating system were used for testing the concerned classifiers. This study tests function based classifiers i.e. "Support Vector Machine (SVM)" with the use of four different Kernel function, i.e. "Normalise Polynomial Kernel (NP)", "Polynomial Kernel (PK)", "Radial Basis Function kernel (RBF)", and "Pearson VII function-based universal kernel (PUK)" on features selected by above feature selection techniques. Thereafter, the results were compared for performance evaluation.

5.5. Evaluation

In this research, 1358 best features are selected by three different techniques. This study starts with low dimensional (158) features and progressively moves to high dimensional (1358) features for classifying 12000 Emails (6000 HAM + 6000 SPAM) files. Thereafter, data splitting is performed where 66% data is split for training of classifiers and remaining 34% data will be taken for our analysis. Percentage "($A_{ccuracy}$)", and "(F_{value})" were used for evaluation.

Accuracy: The ratio of total correctly classified text Email to total number of text Email and defined as

$$A_{ccuracy} = \frac{S^{C}_{PAM} + H^{C}_{AM}}{E^{T}_{MAIL}} \quad (9)$$

Where S^{C}_{PAM} is correctly classified SPAM Emails, H^{C}_{AM} is correctly classified HAM Emails, and E^{T}_{MAIL} is the total emails.

F-value: This value is also used for testing the strength of classifiers. It is calculated by taking the harmonic mean of "($P_{recision}$)" and "(R_{ecall})", and defined as

$$F_{value} = \frac{2 * P_{recision} * R_{ecall}}{P_{recision} + R_{ecall}} \quad (10)$$

These values will help to predict accuracy and strength of classifiers to classify unsolicited Emails.

6. COMPARATIVE ANALYSIS

The Results of the empirical evaluations are displayed in the Tables III, and IV and the Figures 1, 2 and 3. The analysis is divided into three parts: first part incorporates the Analysis of Kernel function, second part takes the Analysis of Feature Selection Mechanisms and last part shows the combined effect. $A_{ccuracy}$ and F_{value} are adopted as measures of performance.

6.1. Analysis of Kernel Functions

In this research, Support Vector Machine (SVM) classifier has been tested by taking four different Kernel Functions on high dimensional 1358 data features and thereafter, low dimensional 158 data features. From Table III and IV and Figure 1, 2 and 3, it is clear that "Normalised Polynomial Kernel (NP)" is best for improving the performance of SVM among other Kernel functions. This Kernel is giving 98.5% accuracy for 1358 high dimensional data features and 78.1% to 85.2% accuracy for 158 low dimensional data features.

TABLE III. ACCURACY & F-VALUE (1358 FEATURES)

Kernels	High Dimensional					
	1358 features					
	CS		GR		LSI	
	Acc (%)	F value (%)	Acc (%)	F value (%)	Acc (%)	F value (%)
NP	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**
PK	97.3	97.3	97.3	97.3	97.3	97.3
RBF	97.9	97.9	97.9	97.9	98	97.9
PUK	94.1	94.1	94.1	94.1	94.1	94.1

TABLE IV. ACCURACY & F-VALUE (158 FEATURES)

Kernels	Low Dimensional					
	158 features					
	CS		GR		LSI	
	Acc (%)	F value (%)	Acc (%)	F value (%)	Acc (%)	F value (%)
NP	**78.9**	**79**	**78.1**	**78.2**	**85.2**	**85.1**
PK	55.2	54.6	53.5	52.7	82.9	82.8
RBF	59.5	58.3	62.8	62.3	79.9	79.5
PUK	**77.3**	**77.3**	**77.4**	**77.3**	**83.5**	**83.4**

Observations show an interesting fact that the movement of features from high to low dimension affect the performance of all Kernels. For high dimensional 1358 data features, PUK Kernel shows poor performance with 94.1% accuracy whereas RBF and PK kernel give second and third best result after NP with 97.9% and 97.3% accuracy respectively. Contradictorily, for low dimensional 158 data features, PF and RBF gives poor performance with 55.2% to 82.8% and 58.3% to 79.9% accuracy respectively whereas PUK Kernel gives second best performance after NP with 77.3% to 83.4% accuracy.

6.2. Analysis of Feature Selection Mechanisms

This study takes three Feature Selection Mechanisms i.e. "Gain Ratio (GR)", "Chi-Squared (χ^2), and "Latent Semantic Indexing (LSI)", for testing our classifiers Table I & II and Figure 1, 2 and 3 demonstrate performance comparison of Feature Selection Mechanisms. Results show that for high dimensional 1358 data features, all techniques give more or less same performance for respective Kernels with 94.1% to 98.5% accuracy. Observations confirm that when we move along with high to low dimension, performance of these mechanisms varies significantly. Among all the mechanisms, Latent Semantic Indexing (LSI) gives best results i.e. for low dimensional 158 data features it gives 83.4% to 85.1% accuracy.

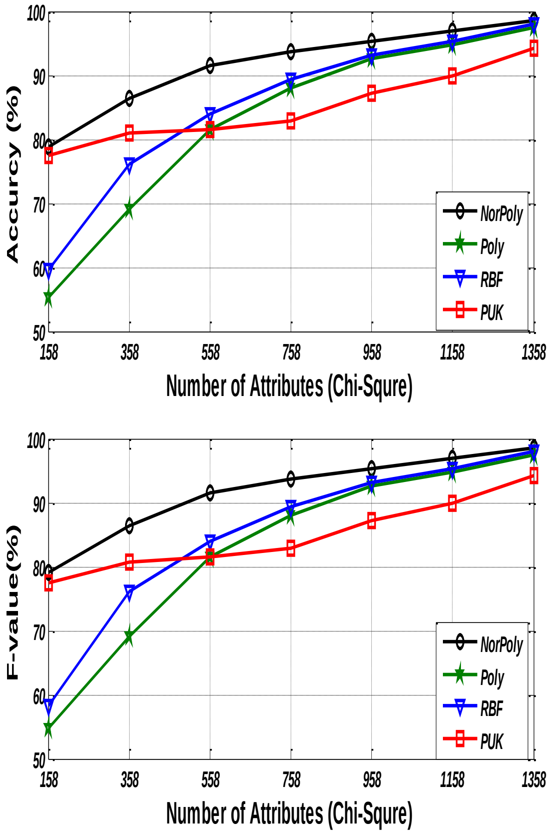

Figure 1. Accuracy & F-Value (Chi Squired)

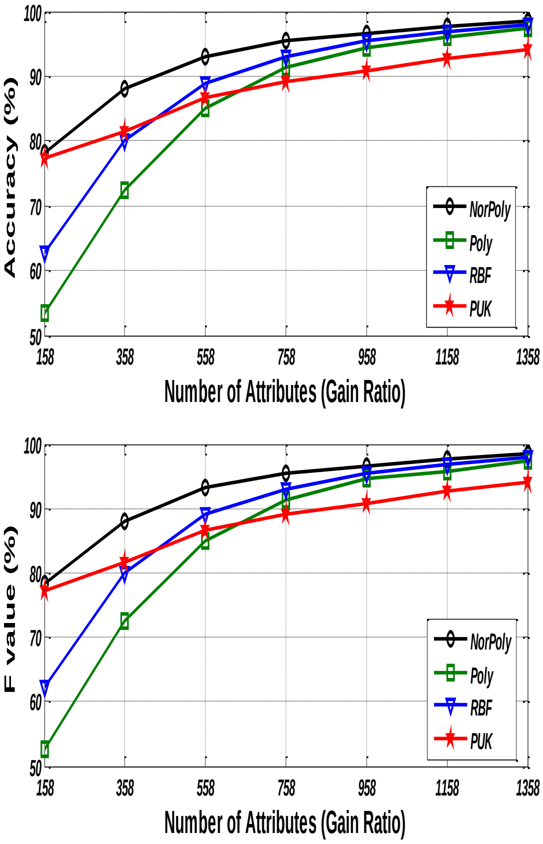

Figure 2. Accuracy & F-Value (Gain Ratio)

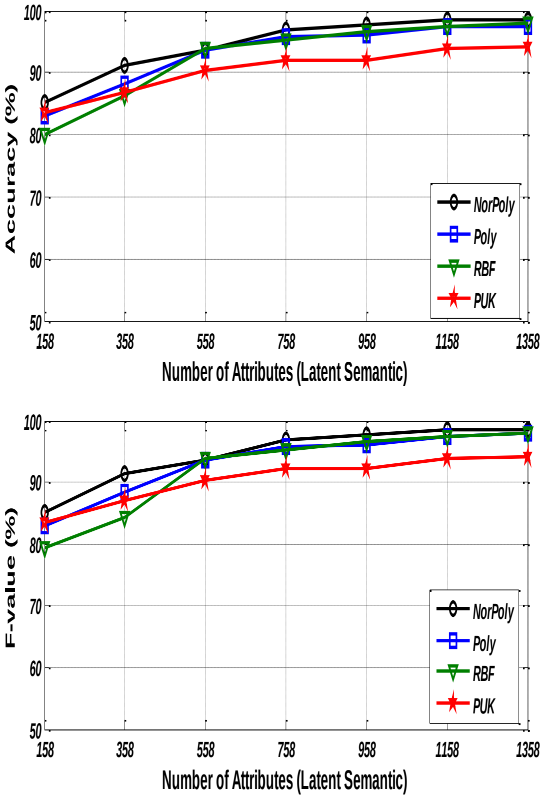

Figure 3. Accuracy & F-Value (LSI)

6.3. Analysis of the Combined Effect

The combined effect of Kernel Functions and Feature selection Mechanisms on SVM is given some interesting facts. For all Kernels, LSI gives best results on high dimensional 1358 data features as well as low dimensional 158 data features. Observation of low dimension shows interesting finding for Kernel PK and RBF. In the case of Gain Ratio (GR), and Chi-Squared (χ^2) feature selection, RBF gives third best result after NP and PUK with 58.3% to 62.8% accuracy and PK gives poor performance among all with 52.7% to 55.2% accuracy. Contradictorily, in the case of Latent Semantic Indexing (LSI), PK gives third best performance after NP and PUK with 82.9% and RBF give poor results among all with 79.5% to 79.9%.

7. CONCLUSION

This research demonstrates that depending on the dimensionality of the data set, the effect of the Kernel function varies significantly. Amongst the Kernel functions tested in this work, NP has shown best SVM performance across a wide range of dimensions when used with a variety of feature selection techniques. The PUK kernel shows variable performance depending on dimensionality, it is good for low dimensional data but poor for high dimensional data. Latent Semantic Indexing (LSI) has been found to be the most effective among the feature selection techniques considered.

The results show that the choices of the Kernel function and feature selection technique have a profound effect on the performance of SVM for SPAM email detection. In future, some other Kernel functions and feature selection techniques can be tested on the same lines. The same classifiers and feature selection techniques can also be tested on other datasets.

8. REFERENCES

[1]. Aladdin Knowledge Systems, Anti-spam white paper, <http://www.eAladdin.com>.

[2]. Trivedi, S., Dey, S., 2013, "Interplay between Probabilistic Classifiers and Boosting Algorithms for Detecting complex unsolicited Emails," selected in International Conference of Information and Network Security, Bangkok, Thailand (ICINS 2013) and for publication in Journal of Advances in Computer Networks (JACN) ISSN: 1793-8244.

[3]. Lai, C. C. 2007, "An empirical study of three machine learning methods for spam filtering,"Journal of Knowledge-Based Systems archive, Volume 20, Issue 3, PP. 249-254.

[4]. Goodman, J., Cormack, G.V., and Heckerman, D. , 2007, “Spam and the ongoing battle for the inbox,” Communications of the ACM, vol.50, issue 2, pp. 24-33.

[5]. Vapnik, V.N., 1999. “An Overview of Statistical Learning Theory”, IEEE Trans.on Neural Network, Vol. 10, No. 5, pp.988-998.

[6]. Drucker, H., Wu, D., and Vapnik, V. N., 1999 “Support Vector Machines for Spam Categorization,” IEEE Transaction of Neural Networks, Vol. 10, No. 5.

[7]. Woitaszek, M., Shaaban, M., and Czernikowski, R., 2003, “Identifying Junk Electronic Mail in Microsoft Outlook with a Support Vector Machine,” conf.

Proceedings, 2003 Symposium on Applications and the Internet, PP 166 – 169, 27-31.

[8]. Zhang, L., Zhu, J., and Yao, T., 2004, “An Evaluation of Statistical Spam Filtering Techniques,” Journal ACM, Transactions on Asian Language Information Processing (TALIP), PP 243 – 269, Volume 3 Issue 4.

[9]. Cheng, V., Li, C.H., 2006, “Personalized Spam Filtering with Semi-supervised Classifier Ensemble,” WI '06 Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, ISBN: 0-7695-2747-7, Pages 195-201.

[10]. Sculley, D., Wachman, G. M., “Relaxed Online SVMs for Spam Filtering” SIGIR '07 Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval, PP 415-422, ISBN: 978-1-59593-597-7, July 2007.

[11]. Chang, M., Yih, W., and Meek, C., 2004, “Partitioned Logistic Regression for Spam Filtering,” KDD '08 Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, PP 97-105 ISBN: 978-1-60558-193-4.

[12]. Awad, W.A., and ELseuofi, S.M., 2012 “Machine Learning Methods for Spam Classification,” International Journal of Computer Science & Information Technology (IJCSIT), PP 173-184, Vol 3, No 1.

[13]. Kumar, R. K., Poonkuzhali, G., Sudhakar, P., 2012, “Comparative Study on Email Spam Classifier using Data Mining Techniques,” Proceedings of International Multi Conference on Engineers and Computer Scientist (IMECS) , Hong Kong, Vol. 1, ISBN: 978-988-19251-1-4.

[14]. Vapnik, V., 1995, “The Nature of Statistical Learning Theory”, Springer, AT&T Bell Labs, Holmdel, NJ.

[15]. Hall, M.A., and Smith, L.A., 1998, “Practical feature subset selection for machine learning”, Proceedings of the 21st Australian Computer Science Conference, pp 181–191.

[16]. Liu, H., and Setiono, R., 1995, “Chi2: Feature selection and discretization of numeric attributes”, Proc. IEEE 7th International Conference on Tools with Artificial Intelligence pp, 338-391.

[17]. Deerwester, S., Dumais, S.T., Furnas, G.W., and. Landauer, T.K., 1990 “Indexing by latent semantic analysis,” J. Amer. Soc. Inform. Sci , pp 391–407.

[18]. Dumais, S.T., 1995, “Using LSI for information filtering,” In: Harman, D., (Eds.), The Third Text REtrieval Conference (TREC3). National Institute of Standards and Technology Special Publications 500-215, pp. 219-230,

[19]. Story, R.E., 1996, “An explanation of the effectiveness of latent semantic indexing by means of a Bayesian regression model,” Inform. Process. Manage. Vol 32, pp 329–344.

[20]. Aas, K. and Eikvil L, 1999, “Text categorisation: A survey, Technical report,” Norwegian Computing Centre.