Comparison of Neural Networks and Support Vector Machines using PCA and ICA for Feature Reduction

J. Sripriya[1] and E. S. Samundeeswari[2]
[1]Research Scholar, [2]Associate Professor
Department of Computer Science,
Vellalar College for Women, Erode,
Tamil Nadu, India

ABSTRACT
Web page classification provides an efficient information search to internet users. However, presently most of the web directories are still being classified manually or semi-automatically. This paper analyses the concept of the statistical analysis methods known as Principal Component Analysis (PCA) and Independent Component Analysis (ICA). The main purpose for using integration of PCA and ICA in Web News Classification is to perform feature separation and reduction. The feature vectors are applied to Neural Networks (NN) and Support Vector Machines (SVM) classifiers. F-measure is used to measure the classification effectiveness and found SVM is better than Neural Networks (NN). For the classification-ability experiment, sports news web page section was used.

General Terms
Dimensionality Reduction, Text Classification

Keywords
Independent Component Analysis, Neural Networks, Principal Component Analysis, Support Vector Machine

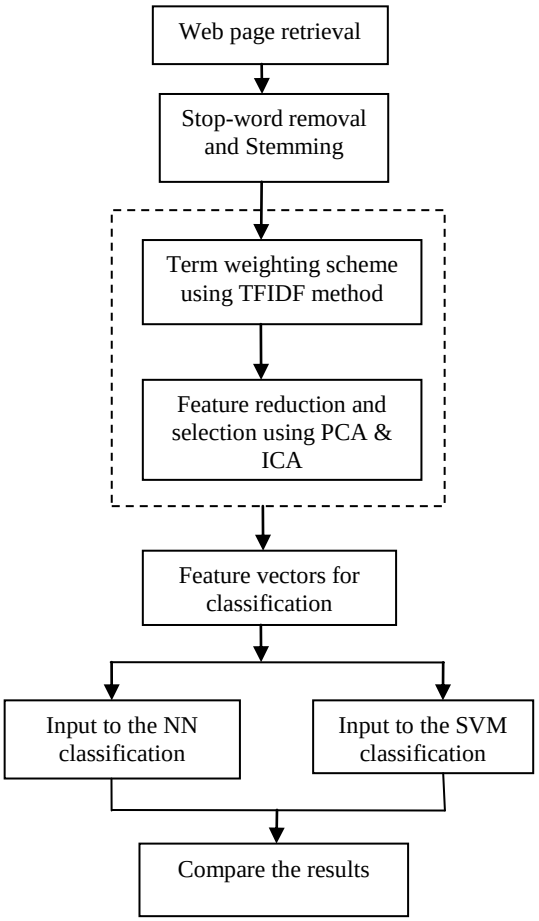
1. INTRODUCTION
Web page classification allows web visitors to navigate a web site quickly and efficiently. Presently, there are two approaches are commonly used by web users to find useful information on the web. The two approaches are using search directory such as *Yahoo, Altavista* or search engine like *Google*. Each approach has its own advantages. As an example internet users may find search directories as useful when browsing for general topics, while they may find search engines work well when searching for specific information. In web directories, web pages are classified into hierarchical categories according to their content and stored in database. This allows the web users to browse desired information according to its category. However at present, most of the web directories are still being classified manually or semi-automatically.

The news web pages have different characteristics as the text length for each of them is variable. The high dimensionality of the news web pages datasets has made the process of classification more difficult. Neural networks [1] have been widely applied by many researchers to classify the text documents with different types of feature vectors. The computational complexity of neural nets depends on the dimensionality of the input space and it suffers from multiple local minima.

The performance of machine learning process is dependent on its features. The fundamental challenge in machine learning is how to extract the features that best represent the original content. Integration of Principal Component Analysis (PCA) and Independent Component Analysis (ICA) in web news classification is to perform feature separation and reduction. PCA is used to reduce the data vectors into small numbers of relevant features before the data is input to the ICA. This paper proposes to use Support Vector Machine (SVM) classification algorithm [7]. The proposed method provides acceptable classification accuracy with the sports news datasets.

The composition of this research is as follows. In section 2, we observe the related research regarding the web page classification issues. Section 3 discussed the factors of choosing feature reduction and selection using PCA and ICA algorithm for this model. Section 4 presents the comparisons of web page classification algorithms NN classifier and SVM classifier and measures of the classification effectiveness. Section 5 provides the experimental results for comparison of the classification efficiency between the classifiers on sports web news dataset. Finally section 6 concludes the web page classification work.

2. WEB PAGE CLASSIFICATION
This model consists of several modules such as web page retrieval process that includes stop-word filtering and stemming, feature reduction and selection, classification and evaluation. To classify the web pages after the preprocessing, the PCA algorithm is used to reduce the original data vectors to small number of relevant features [10]. The ICA is used for data separation. After the integration of PCA and ICA, the extracted features are input to NN and SVM classifiers. Finally the performance accuracy of the above two classifiers are compared. Fig.1 shows the process of web page classification.



PCA – Principal Component Analysis
ICA – Independent Component Analysis
NN – Neural Network
SVM – Support Vector Machine

Fig. 1 - The process of web news classification

2.1 Web Page Retrieval and Pre-processing

Web page retrieval [6] is a process to retrieve collections of web documents to the database from internet with the help of web crawler or web browser. These retrieved web pages are pre-processed to transform them as text documents. The pre-processing steps are removing stop-words and stemming.

2.1.1 Stop-word removal

Many web news classification systems use stop-word list to delete "noise" words before going to classification algorithm [12]. Examples of noisy words are 'and', 'are', 'have', 'is, 'the', etc that appear very frequently in all the documents, but almost always carry no useful information. Stop-word removal is a process, which filters these common words that exist in the web document by using the stop-word list.

2.1.2 Stemming

A stem is a natural group of words with equal or very similar meaning. After the stemming process, every word is represented by its stem. In this web page classification method, the Porter Stemmer algorithm is used for stemming process.

2.2 Term Weighting Scheme

After the stemming and stopping process the data will be represented as the document-term frequency matrix (Doc_j x TF_{jk}). The tf–idf [7] is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus.

Calculation based on the term frequency - inverse document frequency (*tfidf*) will be defined before feature reduction and feature selection.

$$\text{Term Weight} = x_{jk} = tf_{jk} * idf_k \qquad (1)$$

Term Frequency (TF):

$$tf_{jk} = n_{jk} / \Sigma_k n_{j,k} \qquad (2)$$

Inverse Document Frequency (IDF):

$$idf_k = \log(n / df_k) \qquad (3)$$

Table 1- Explanation of terms for TF-IDF calculation

Index	Explanation
n	Total number of documents in the database
k	Number of words
x_{jk}	Terms weight of k^{th} word in j^{th} document
Doc_j	Each web page document that exists in local database
Tf_{jk}	How many times number of the distinct word w_m occurs in document Doc_j
n_{jk}	Number of occurrences of the considered term (t_k) in the document (d_j)
$\sum_k n_{j,k}$	Size of the document
df_k	Total number of documents in the database that contains a word w_k

3. DIMENSIONALITY REDUCTION

Dimensionality reduction (DR) [11] is beneficial in that it tends to reduce the problem of over-fitting. Recently, a lot of techniques for viewing DR can be classified as global vs. local and it depends on whether the task is approached locally or globally. Both methods eventually give a global coordination of the intrinsic dimension of the manifold.

3.1 Feature Reduction using PCA

The dimensionality reduction process to reduce the original data vectors into small number of relevant features using PCA and it calculates the Eigen-vectors of the covariance matrix, and projects the original data onto a lower dimensional feature space, which is defined by Eigen-vectors with large Eigen-values[4].

Let M to be the matrix document-terms weight as below:

$$M = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1k} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2k} & \cdots & x_{2m} \\ x_{31} & x_{32} & \cdots & x_{3k} & \cdots & x_{3m} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \cdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} & \cdots & x_{nm} \end{bmatrix}$$

The definition of x, j, k, m and n have been explained in Table 1. The mean of m variables in data matrix M will be calculated by

$$\bar{x} = \frac{1}{n}\sum_{j=1}^{n} x_{jk} \qquad (4)$$

Then the covariance matrix, C = $\{c_{jk}\}$ is calculated. The variance C^2_{kk} is given by

$$C^2_{kk} = \frac{1}{n}\sum_{j=1}^{n}(x_{jk} - \bar{x}_k)^2 \qquad (5)$$

The covariance is given by

$$C^2_{ik} = \frac{1}{n}\sum_{j=1}^{n}(x_{ji} - \bar{x}i)(x_{jk} - \bar{x}_k) \qquad (6)$$

where i=1,2,…,m. An eigen-value **λ** and eigen-vector **e** is found by C**e** = **λe** where C is covariance matrix. If C is an m * m matrix of full rank, m eigen-values and all corresponding eigen-vectors are found by using

$$(C - \lambda_i I)e_i = 0 \qquad (7)$$

The eigen-values and eigen-vectors are sorting by $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m$. A square matrix E can be constructed from the eigen-vector columns where E=[$\mathbf{e}_1$ $\mathbf{e}_2$ ... $\mathbf{e}_m$]. Let matrix B be denoted as

$$B = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \cdots \\ 0 & 0 & 0 & \lambda_m \end{bmatrix}$$

Now the eigen-value decomposition is performing to get the principal component of matrix C by using

$$E^T C E = B \qquad (8)$$

The first d≤m eigen-vectors are selecting, where d is the desired value such as 300, 400, etc. The set of principal components is represented as $Z_1 = \mathbf{e}_1^T C$, $Z_2 = \mathbf{e}_2^T C$, …, $Z_p = \mathbf{e}_p^T C$. An n*p matrix R is represented as

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1p} \\ r_{21} & r_{22} & \cdots & r_{2p} \\ \vdots & \vdots & \ddots & \cdots \\ r_{n1} & r_{n2} & \cdots & r_{np} \end{bmatrix}$$

where R is a reduced feature vectors from the original size m * m to n * p size.

3.2 Feature Reduction using ICA

Independent component analysis (ICA) [13] is the most common method for generating spatially localized features and is used to produce basis vectors that are statistically independent. In matrix form, the ICA framework is usually defined as linear noise-free generative model

$$X=AS \qquad (9)$$

where S be the vector of source signals and the latent independent component is represented as random variable vector $S=(s_1,s_2,..,s_n)^T$. X is the observed mixture signals which $X=(x_1,x_2,\ldots,x_n)^T$ that are generated by multiplying A where matrix $A=(a_1,a_2,..,a_n)$ is a constant n x n mixing square matrix. It can be expressed by

$$X=a_1s_1+a_2s_2+\ldots+a_ns_n \qquad (10)$$

which is written as

$$X = \sum_{k=1}^{n} a_k s_k \qquad (11)$$

where column a_k of the mixing matrix A, give the basis where the observations are represented. The goal of ICA is given X, find S and A where both A and S are statistically independent.

Independent component analysis relies on the concept of statistical independence [10]. Statistical independence is expressed as follows:

Random variables A and B are independent if the conditional probability of A with respect to A is just the probability of A. This means, knowing the value of B tells us nothing about A. Following is the equation:

$$P(A|B)=P(A) \qquad (12)$$

Since P(A|B)= P(A,B), where P(A,B) is the joint density function of A and B. Then

$$P(A,B) = P(A)P(B) \qquad (13)$$

The whitened data have the form of

$$X = D^{-1/2}E^T V \qquad (14)$$

where X is whitened data vector, D is a diagonal matrix containing the eigen-values of the correlation matrix and E contains the corresponding eigen-vectors of the correlation matrix as its columns.

$$w \leftarrow E\{Xg(w^T X)\} - E\{g'(w^T X)\}w \qquad (15)$$

where w is one of the rows of the un-mixing matrix W. X is the data vectors that have been centered and whitened in equation (14). The nonlinear function g is chosen so that it is the derivative of the non-quadratic contrast function g. The choice of is important to optimize the performance of the algorithm. An initial unit norm vector w is chosen randomly. w is again normalized to have unit norm after each iteration step (15). The iteration is continued until the direction of w does not change significantly.

4. CLASSIFICATION

4.1 Neural Network Classifiers

The main objective to use neural network (NN) [1] is its learning and generalization characteristics. Learning is the ability to approximate the underlying behaviour adaptively from given data, while generalization is the ability to predict efficiently beyond the trained data.

In this model the output of the feature vector is given as input of the NN classifier. The feed forward-back propagation neural network [11] is adapted as the classifiers. For classifying a test document d_j, its term weight w_{jk} is loaded into the input units. The activation of these units is propagating forward through the network, and finally the value of the output unit(s) determines the categorization decision(s). The back propagation neural network is used because, if a misclassification occurs, the error is "back propagated" so as to change the parameters of the network and minimize or eliminate the error.

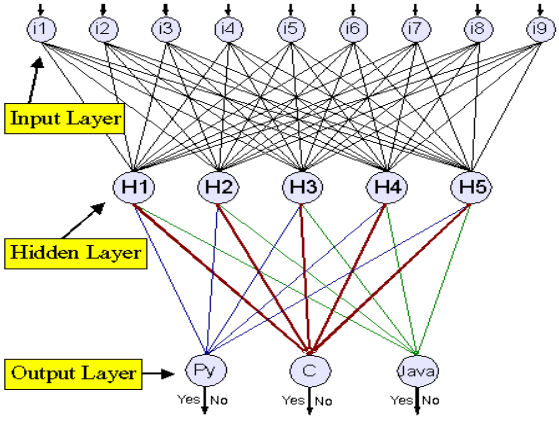


Fig. 2 Back-propagation neural network

The *Back-propagation* algorithm extends the analysis that underpins the delta rule to neural nets with hidden nodes which is shown in Fig.2.

4.2 SVM classifiers

Support Vector Machine (SVM) [7] is a supervised classification algorithm. During training, this algorithm constructs a hyper-plane that maximally separates the positive and negative instances in the training set.

In this model, a document *d* is represented by a possibly weighted vector ($t_{d1},\ldots,t_{dN}$) of the counts of its words. A single SVM can only separate two classes: a) positive class L_1 (indicated by $y = +1$) and b) negative class L_2 (indicated by $y=-1$). In the space of input vectors a hyper-plane may be defined by setting $y=0$ in the following linear equation.

$$y = f(t_d) = b_0 + \sum_{j=1}^{N} b_j t_{dj} \qquad (16)$$

The parameters b_j are adapted in such a way that the distance (margin) between the hyper-plane and the closest positive and negative example documents are maximized as shown in Fig. 3.

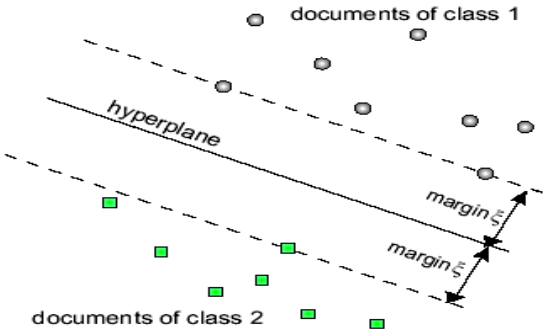


Fig.3 Hyper-plane with maximal distance (margin) to examples of positive and negative classes constructed by the support vector machine

In documents, distance from the hyper-plane is called *support vectors* and determine the actual location of the hyper-plane. A new document with term vector *td* is classified in $L1$ if the value $f(td) > 0$ and into $L2$ otherwise. In case that the document vectors of the two classes are not linearly separable a hyper-plane is selected such that as few as possible document vectors are located on the "wrong" side. SVM can be used with non-linear predictors by transforming the usual input features in a non-linear way, e.g. by defining a *feature map* [8].

$$\phi(t_1,\ldots,t_N) = (t_1,\ldots t_N, t_1^2, t_1t_2,\ldots, t_Nt_{N-1}, t_N^2) \qquad (17)$$

Substantially a hyper-plane defined in the expanded input space. The most important property of SVM is that learning nearly independent of the dimensionality of the feature space [2]. It requires feature selection as it inherently selects data points (the support vectors) required for classification. This allows generalization even in the presence of a large number of features and makes SVM suitable for the classification of texts.

Advantages of SVM:

- To avoid over-learning by structural risk minimization.
- To deal with higher-dimensional data, need not much more calculation amount and memory capacity in learning.

4.3 Measures of Classification Effectiveness

The classification effectiveness of this model is measured using standard information retrieval measurement that are precision, recall, and F-measure. Precision (P) is the fraction of retrieved documents that are relevant to the search and it measures the exactness of a classifier. Recall (R) is the fraction of the documents that are relevant to the query that are successfully retrieved and it measures the completeness or sensitivity of a classifier.

Precision (P) and Recall (R) are defined as:

$$P = \left\{ \frac{TP}{TP + FP} \right.$$

$$R = \left\{ \frac{TP}{TP + FN} \right.$$

Table 2 – Explanation of parameters

Predicted class	Actual class	
	TP (True Positive)	**FP** (False Positive)
	FN(False Negative)	**TN**(True Negative)

The F-measure (F) combines precision and recall with equal importance into a single parameter for optimization.

$$F = \frac{2PR}{P+R}$$

Among the various classifiers studied, the neural networks and support vector machines are applied for web news classification. Stop-word removal and stemming for preprocessing and dimensionality reduction is done using PCA and ICA. The resultant feature vector is applied to the selected classifiers and the results are discussed in the following chapter.

5. RESULTS AND DISCUSSION

The experimental evaluation of the classification performance on Sports News dataset is used. The news types are Boxing, Cricket, Cycling, Football, Golf, Swimming, Tennis and other various sports. Neural Network and Support Vector Machine algorithms are used in this work. The classification accuracy is done using F-measure for the sports news data set.

Classification performance of NN and SVM classifiers are shown in the Table3:

Table 3 - F-measure value for Sports News dataset

Datasets	Classification Algorithms	
	NN	**SVM**
Boxing	0.4286	0.8010
Cricket	0.4615	0.8235
Cycling	0.5143	0.6957
Football	0.6471	0.6960
Golf	0.3407	0.6897
Swimming	0.4516	0.7501
Tennis	0.6061	0.7755

Comparison of F-measure values of NN with SVM classification for boxing and tennis datasets are shown in Fig.4 and Fig.5 respectively.

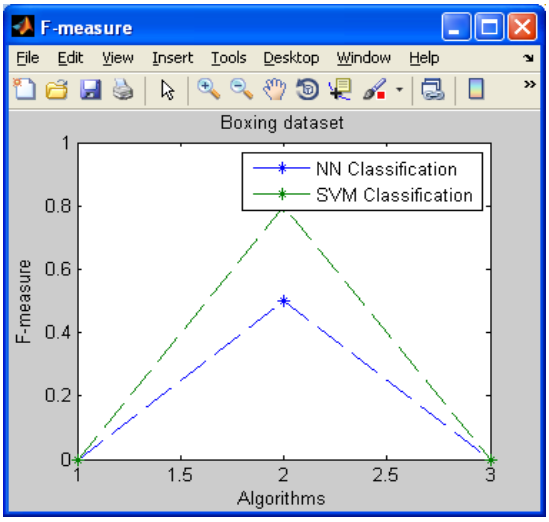


Fig.4 F-Measure for boxing dataset

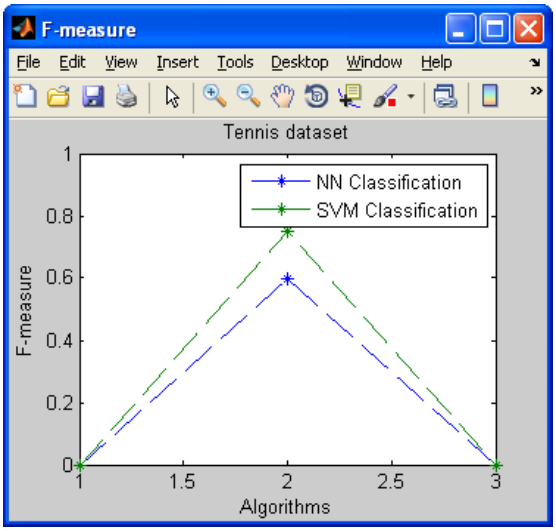


Fig.5 F-Measure for tennis dataset

Fig.6 shows the comparison of NN with SVM classification for different sports news datasets.

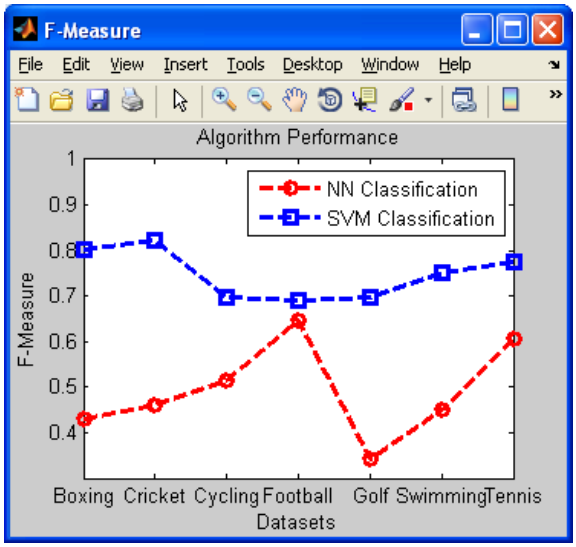


Fig.6 Comparison of NN and SVM on Sports News dataset

This work classifies web news documents using the two classification algorithms Neural Networks and Support Vector Machines. The support vector machines are capable of efficiently processing the high dimensional feature vectors. Based on the classification accuracy, Fig.6 shows the Support Vector Machine is found to be better than the Neural Networks.

6. CONCLUSION

Text mining has proved to be a useful tool in exploring and discovering interesting and useful knowledge in unstructured text. This work studied various classifiers and preprocessing methods. Different preprocessing techniques such as Term Frequency (TF), Inverse Document Frequency (IDF), and Principal Component Analysis (PCA), Independent Component Analysis (ICA) for dimensionality reduction techniques are discussed. After preprocessing, each preprocessed dataset is classified using Neural Networks and Support Vector Machines classifiers. Finally the performance of the Neural Network and Support Vector Machine algorithms are compared using the F-measure value. The experimental results show that the SVM classifier is more accurate than the neural network for sports news real time dataset.

The work can be extended to identify the impact of Information Gain (IG), Mutual Information (MI) and Chi-Square Tests for feature selection. This work applied SVM on the tuned feature vector. Further work can use SVM without any parameter tuning and to find good parameter settings automatically.

7. REFERENCES

[1] S. Ali and O. Sigeru, Web page feature selection and classification using neural networks, Inf. Sci. Inf. Comput. Sci., vol. 158, pp. 69-88, 2004.

[2] J. Brank & M. Grobelnik, N. Milic-Frayling, D. Mladenic, Training text classifiers with SVM on very few positive examples, Microsoft Research Technical Report MSR-TR-2003-34, 2003.

[3] Burges, C.J.C, A Tutorial on Support Vector Machines for Pattern Recognition, Data Mining and Knowledge Discovery Vol.2 No.2 (1998) 121-167.

[4] R. A. Calvo, M. Partridge, and M. A. Jabri, A Comparative Study of Principal Component Analysis Techniques, presented at In Proc. Ninth Australian Conf. on Neural Networks, Brisbane, 1998.

[5] G. G. Chowdhury, Introduction to modern information retrieval. London: Library Association Publishing, 1999.

[6] S. T. Dumais, Improving the retrieval of information from extemal sources. Behavior Research Methods, Instruments and Computers, 23(2), 229-236, 1991.

[7] T. Joachims, Probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization, Proceedings of International Conference on Machine Learning (ICML), 1997.

[8] Joachims, T. Advances in Kernal Methods-Support Vector Learning, chapter Making Large-Scale SVM Learning Practical MIT-Press, 1999.

[9] T. Joachims, Learning to Classify Text using Support Vector Machines, Kluwer, 2002.

[10] G. Karypis and E.H. Sam, Concept indexing: a fast dimensionality reduction algorithm with applications IO document retrieval & categorization, ClKM 2000 (2000).

[11] S. L.Y. Lam and D. L. Lee, Feature reduction for neural network based text categorization, Proceedings of the 6th International Conference on Database Systems for Advanced Applications 19 - 22 April Hsinchu, Taiwan, 1999.

[12] Lee Zhi Sam, Mohd Aizaini bin Maarof, Ali Selamat, Feature Extraction for Illicit We Pages Identifications Using Independent Component Analysis, International Conference on Intelligence and Advanced Systems, 2007.

[13] Prof.Dr.Markus Borschbach, A Hierarchical ICA-based Text Classifier, Institut fur Informatik, 2010.

[14] H. Mase and H. Tsuji. Experiments on automatic web page categorization for information retrieval system, Joumal of Information Processing, IPSJ Journal, Feb. 2001, pg. 334-347, 2001.

[15] Miao Zhang, De-xian Zhang, Trained SVMs based rules extraction method for text classification, IEEE International Symposium on IT in Medicine and Education, 2008, ITME 2008, 12-14 Dec. 2008.

[16] F. Sebastini, Machine learning in automated text categorization, ACM Computing Surveys, 34(1), 2002.