International Journal of Computer Engineering & Technology (IJCET)
Volume 9, Issue 3, May-June 2018, pp. 250–258, Article IJCET_09_03_030
Available online at http://www.iaeme.com/ijcet/issues.asp?JType=IJCET&VType=9&IType=3
Journal Impact Factor (2016): 9.3590(Calculated by GISI) www.jifactor.com
ISSN Print: 0976-6367 and ISSN Online: 0976–6375
© **IAEME** Publication

DOCUMENT CLASSIFICATION USING SVM COMBINED WITH OPTIMAL FEATURE SELECTION

Deepanshu
M. Tech Scholar, Kurukshetra University Kurukshetra, India

Dr. Ramesh Kait
Assistant Professor, Kurukshetra University Kurukshetra, India

ABSTRACT

Proper selection of features is more important and effective as compared to the initial feature set which is very large (may reach to hundreds or even thousands). Feature selection is one of the most important steps and plays a good role in the improvement of accuracy and time of processing in text classification. "Curse of dimensionality" is the phenomenon which might degrade the classification accuracy so to overcome the curse of dimensionality it is required to do feature reduction which is of two types- feature selection and feature extraction. In this paper, our approach is to use feature selection based on information gain(IG) on mini-newsgroups and after that comparison of the performance of three classifiers viz. Support Vector Machine(SVM), Naïve Bayes(NB), K-nearest neighbor(kNN) is made and finally, find out that which classifier outperformed the three classifiers. On the basis of classification accuracy, precision, F-measure, and recall, we also differentiated the SVM and SVMfs (with feature selection based on information gain).

Key words: Feature Selection Techniques, Information Gain, Naïve Bayes, Support Vector Machine, Text Classification.

Cite this Article: Deepanshu and Dr. Ramesh Kait, Document Classification Using SVM Combined With Optimal Feature Selection. *International Journal of Computer Engineering & Technology*, 9(3), 2018, pp. 250–258.
http://www.iaeme.com/ijcet/issues.asp?JType=IJCET&VType=9&IType=3

1. INTRODUCTION

The availability of electronic documents are increasing and exploding day by day and due to this it has become arduous for the users to quickly find the useful labeled information which is possible through text classification (TC) task [1]. For the addressing of this task, there are many learning algorithms [2]. Feature Selection is one of the most important step in the text classification (TC) for this work some dimensionality reduction techniques [3] are applied for improving the performance of classifiers [4][5][6].In general, feature selection is an optimization problem in which space is searched for the possible feature subsets and selection of the subset that is optimal or near-optimal with respect to an objective function which is base for improving certain criteria of classifiers. For instance, for feature selection, we have a set W and x_1, x_2, x_n are original dimensions, W $=\{x_1, x_2, x_3, x_4 \ldots\ldots\ldots\ldots x_n\}$ where n is the number of features. Pick a subset W' such that W' is a subset of W and W'$=\{x_1\ x_2\ x_3 \ldots\ldots\ldots x_m\}$where m is the number of features which differ from n.

1.1. Curse of dimensionality: If the number of training examples is fixed, the performance of the classifier usually degrade for a large number of features which can be understood with the help of graph as shown in Fig 1.1. The main reason behind this problem is irrelevant features which introduce noise (total number of possible subsets is 2^n) and it is impossible to enumerate each of these possible subsets. These noisy features fool the learning algorithms by making results wrong which results in degradation of performance of classifiers.

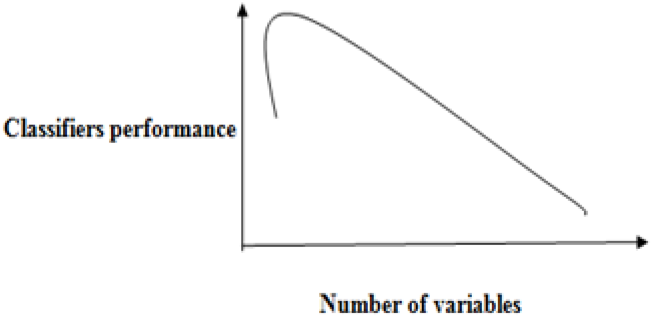


Fig 1.1 Curse of dimensionality

1.2. Information Gain: Another important term used for feature selection is information gain (IG). [7] It measures the no. of bit information taken for category prediction by knowing the presence and absence of a term in a document.

The information gain measure *I*(*w*) for a given word *w* is defined as follows:

$$I(w) = -\sum_{i=1}^{k} p_i . \log(p_i) + F(w). \sum_{i=1}^{k} p_i(w). \log(p_i(w)) + (1 - F(w)). \sum_{i=1}^{k} (1 - p_i). \log(1 - p_i(w))$$

Where pi(w) is the probability of the words, I(w) is the information gain of the words.

1.3. Entropy: What is the smallest possible number of bits, on average, per symbol, needed to transmit a stream of symbols drawn from X's distribution is known as entropy [7].

The Entropy of X: $H(X) = -\sum_{j=1}^{m} p_j \log p_j$

High Entopy: X is from a uniform distribution.

Low Entropy**:** X is from varied distribution.

In this study, the feature selection method based on information gain, which both provide the opportunity for making effective classification by improving the accuracy and simplify the classifiers complexity, taken into account.

The rest of the paper is organized as follows: Related work of the study is given in Section 2, proposed work is in Section 3, experimental results are provided in Section 4 and the conclusion part is included in Section 5.

2. LITERATURE SURVEY

M.Ikonomakis et al.[8] In this paper the automated text classification(TC) has been considered as a main and imperative method to manage & process a big amount of documents in electronic forms that are spread and continuously accelerating. In general, text classification plays a vital role in information retrieval and summarization, text extraction, and question & answering.

Also, This paper discussed the text classification steps using machine learning techniques. The performance of the classifiers depends on the training set if good training is provided then the performance of classification will be improved. Some points described in the paper are as follows: Dimensionality reduction is more efficient over large corpus i.e curse of dimensionality, Highly trained data give better performance in terms of accuracy, precision, recall etc. In other words, training text corpus optimizes the performance of the classifier.

Mehdad and Tetreault [9] Several experiments were performed in this paper. They tried to improve the f-score by simply using a character *n*-gram model and a support vector machine with Naive Bayes (NB) features as a classifier. In this paper, they used a Support Vector Machine (SVM) variant using NB log-count ratios as feature values which consistently performed better across different tasks and data sets. They released an implementation in Python.

Bo Tang et al. [10] This paper shows the Bayesian classification approach for TC using specific feature subset for each class. For the application of these class-specific features for classification, they follow baggenstoss's PDF projection theorem (PPT) to construct again the PDFs in raw data space from the class-specific PDFs in low-dimensional feature subspace and construct a classification rule related to Bayesian. In this paper due to class-specific features, it allows to get the most imperative features for classification and they derived a new naïve bayes rule that follows the PPT. The main importance of this approach is that the feature selection criteria like IG(Information Gain) and Maximum Discrimination(MD) can be easily corporated which leads to good performance. The evaluation of method's classification on several real-world benchmarks. Instead of using the combination operation to select a global feature subset for full classes, here is the selection of a specific feature subset for each class which is known as class-specific features, with the help of these class-specific features a Bayesian classification rule is made. Comparison of this approach using class-specific features with the other conventional approaches using non-class-specific features was made, where the global combination functions of the sum, the average, and the maximum are applied and with another one class-specific feature selection method using one vs. all scheme.

3. PROPOSED WORK

The methods which can be applied for feature selection can be optimum methods, heuristic methods, and randomized methods. However optimum methods can be applied if the hypothesis space or feature subset space has a structure otherwise we can use other methods. This method is an optimization algorithm which works in polynomial time. The methods which consider the different feature subsets will also have some mechanism to evaluate the subset. These methods are of two types: Supervised Method and Unsupervised method.

Supervised Method: The Evaluation of feature subset is done by using it on a learning algorithm. These are also called wrapper method in which trained using selected subset and estimate error on validation subset.

Unsupervised Method: Here the subsets is not evaluated over the training examples. Information content is evaluated in an unsupervised way. Feature selection is an optimization problem. The feature selection steps are as follows and as shown in Fig 1.2

Step 1 Search the space of possible feature subsets.

Step 2 Pick the subset that is optimal or near-optimal with respect to some objective function.

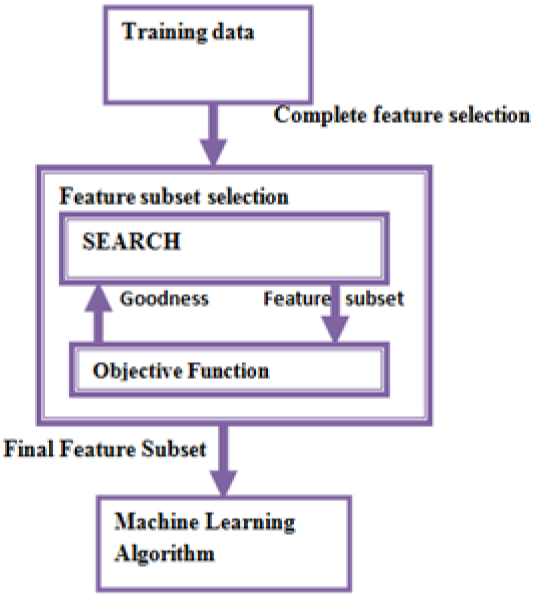
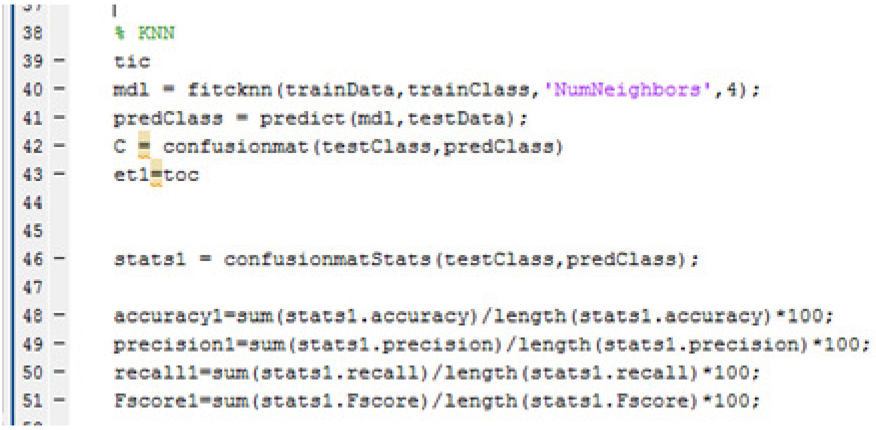


Fig.1.2 Feature selection steps

This work is implemented on tool MatLab (2015a). Comparison of classifiers viz Support Vector Machine, Naïve Bayes, k-NN on mini newsgroups (alt.atheism, comp.graphics, comp.os.ms-windows.misc) dataset by applying several steps on this. The program is so organized that 60% training phase and 40% testing phase in the model. Classifiers used here are k-NN, SVM and Naïve Bayes are as follows with respect to their code.

3.1. k-NN Classifiers

It is simple and generally used classification technique among all supervised machine learning algorithms mainly used for classification and retrieval. Based on the nearest training examples in feature space, it classifies the objects[11]. It is also known as instance-based/case based learning or lazy learning algorithm. The assumption in this algorithm is that documents can be classified in euclidian space as points and distance between two points can be calculated. It is not applicable for large repository where dynamic application is needed. Code for the k-NN classifier is as shown in Fig 1.3.

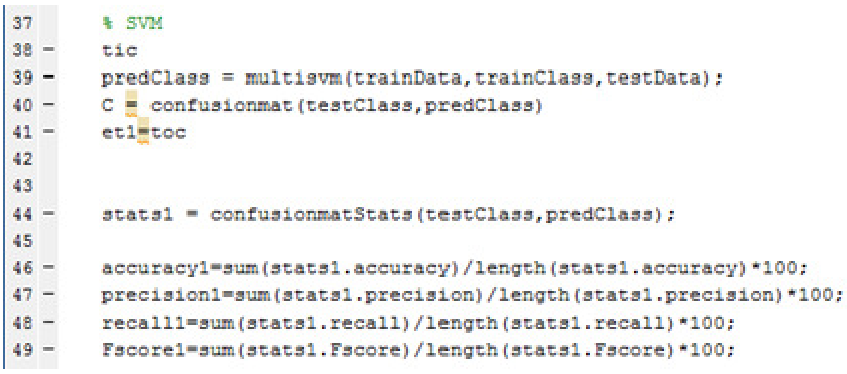


```
% KNN
tic
mdl = fitcknn(trainData,trainClass,'NumNeighbors',4);
predClass = predict(mdl,testData);
C = confusionmat(testClass,predClass)
et1=toc


stats1 = confusionmatStats(testClass,predClass);

accuracy1=sum(stats1.accuracy)/length(stats1.accuracy)*100;
precision1=sum(stats1.precision)/length(stats1.precision)*100;
recall1=sum(stats1.recall)/length(stats1.recall)*100;
Fscore1=sum(stats1.Fscore)/length(stats1.Fscore)*100;
```

Fig.1.3 k-NN classifier code in matlab

3.2. SVM Classifier

This is the most efficient supervised machine learning algorithm [12][13]. It can be applied only to the binary classification and independent of the dimensionality of feature space. This algorithm finds a hyperplane which lies between the positive and negative examples of the training set. Code for the SVM classifier is as shown in fig 1.4.

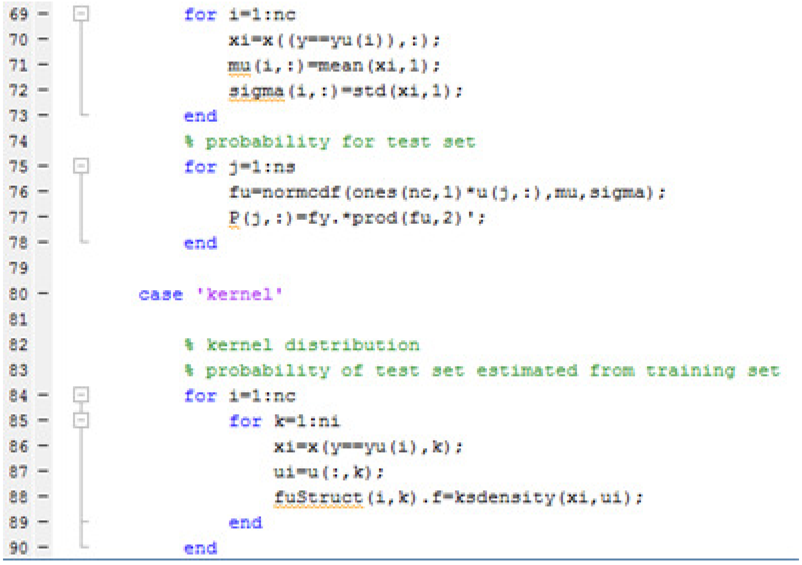


```
% SVM
tic
predClass = multisvm(trainData,trainClass,testData);
C = confusionmat(testClass,predClass)
et1=toc


stats1 = confusionmatStats(testClass,predClass);

accuracy1=sum(stats1.accuracy)/length(stats1.accuracy)*100;
precision1=sum(stats1.precision)/length(stats1.precision)*100;
recall1=sum(stats1.recall)/length(stats1.recall)*100;
Fscore1=sum(stats1.Fscore)/length(stats1.Fscore)*100;
```

Fig.1.4 SVM classifier code in Matlab

3.3. Naïve Bayes classifier

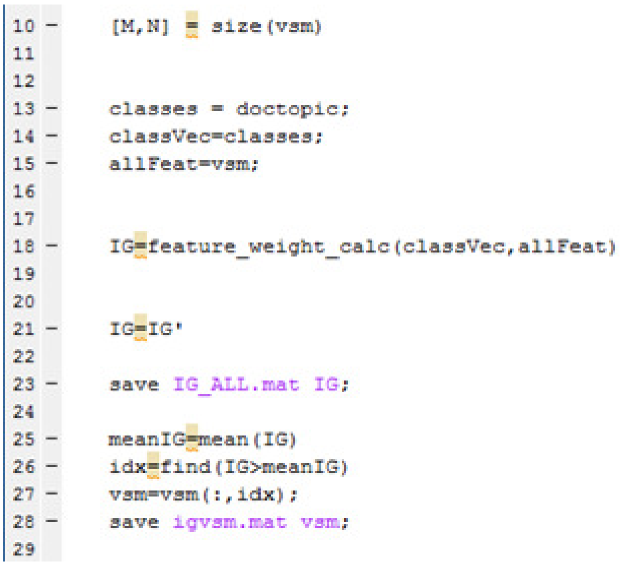
It is probabilistic classifier which is based on Bayes theorem [14] and consists of strong and naïve independence assumptions. Comparatively, it is quite efficient as it is less computationally intensive and it required a less amount of training data [15][16]. Code for the naïve Bayes classifier is as shown in fig 1.5.



```
        for i=1:nc
            xi=x((y==yu(i)),:);
            mu(i,:)=mean(xi,1);
            sigma(i,:)=std(xi,1);
        end
        % probability for test set
        for j=1:ns
            fu=normcdf(ones(nc,1)*u(j,:),mu,sigma);
            P(j,:)=fy.*prod(fu,2)';
        end

    case 'kernel'

        % kernel distribution
        % probability of test set estimated from training set
        for i=1:nc
            for k=1:ni
                xi=x(y==yu(i),k);
                ui=u(:,k);
                fuStruct(i,k).f=ksdensity(xi,ui);
            end
        end
```

Fig. 1.5 Naïve Bayes code in MatLab

Feature selection based on information gain is used for improving the performance of classifiers. Here as shown in Fig. 1.6 threshold is selected meanIG and values are taken only which are greater than meanIG. It directly reduces the dimensionality of the space.



```
[M,N] = size(vsm)


classes = doctopic;
classVec=classes;
allFeat=vsm;


IG=feature_weight_calc(classVec,allFeat)


IG=IG'

save IG_ALL.mat IG;

meanIG=mean(IG)
idx=find(IG>meanIG)
vsm=vsm(:,idx);
save igvsm.mat vsm;

```

Fig.1.6 Feature selection based on Information Gain

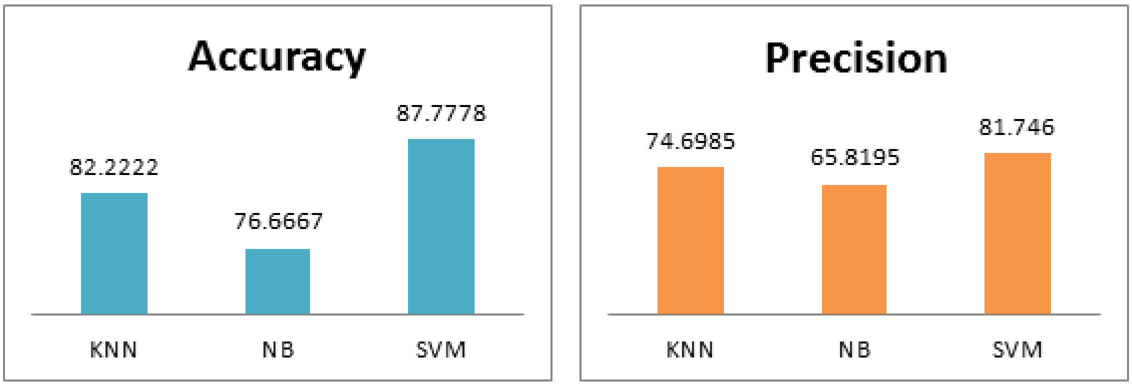
4. EXPERIMENTAL RESULTS AND ANALYSIS

Significant changes can be seen from the graphical representation of the graphs of the classifiers with the feature selection in case of SVM and k-NN after applying feature selection before the classifiers, but in case of Naïve Bayes classifier very least change occur. Comparison for the classifiers on the basis of classification performance metrics is as shown in Table 1.1

Table 1.1.

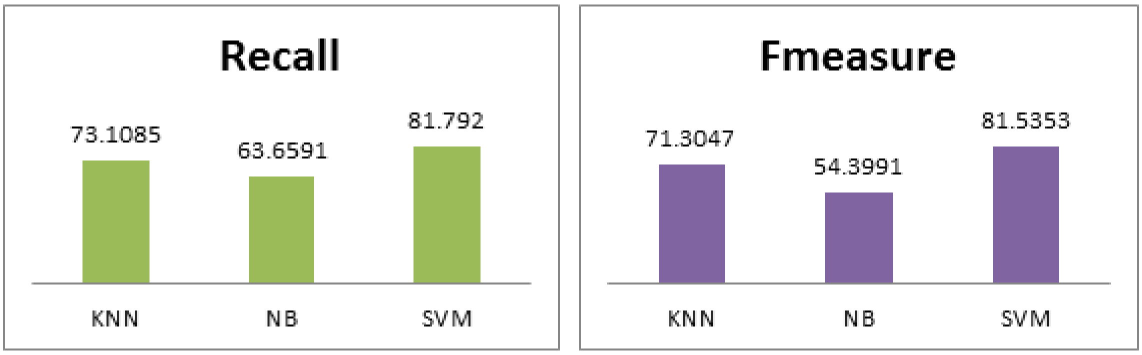
Classifiers	Accuracy	Precision	Recall	Fmeasure
KNN	82.2222	74.6985	73.1085	71.3047
NB	76.6667	65.8195	63.6591	54.3991
SVM	87.7778	81.746	81.792	81.5353

Table 1.1 Comparison of classifiers based on classification performance metrics From the graphs, it is clearly seen that SVM outperformed among all classifiers in performance metrics. The order of performance of the classifiers is SVM> KNN>NB.



(a)

(b)



(c) (d)

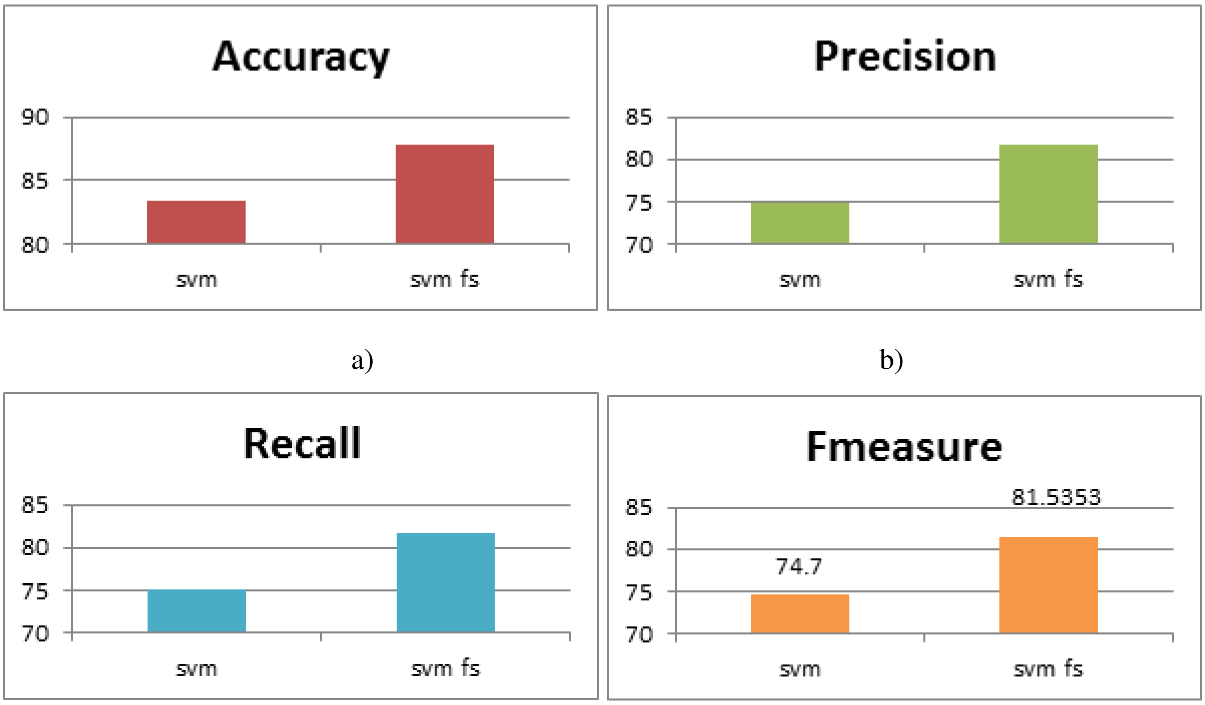
Fig. 1.7 Improvement in different performance metrics a) accuracy b) precision c) Recall d) F measure using feature selection

A comparative analysis of svm and svmfs (with feature selection) is shown in Table 1.2. It is clearly seen that using feature selection improvement occurs in all metrics. Dimensionality reduces up to certain extent which makes the improvement possible. In Fig.2.1 all the graphs are shown for the comparison of svm and svmfs.

Table 1.2

Classifier	Accuracy	Precision	Recall	Fmeasure
svm	83.333	74.909	75.2	74.7
svm fs	87.7778	81.746	81.792	81.5353

Table 1.2 Comparison of svm and svmfs (with feature selection) on the basis of performance metrics.



a) b)

c) d)

Fig 2.1 Comparison of svm without feature selection and svm fs(with feature selection) on the basis of different metrics a)accuracy b)Precision c) Recall d) Fmeasure

5. CONCLUSION AND FUTURE SCOPE

Feature selection can be extremely useful in reducing the dimensionality of the data to be processed by the classifier It is valuable for the performance of the classifier to reduce the dimension of the feature set. In this paper performance of three classifiers SVM, Naïve Bayes, k-NN are measured and compared them on the basis of classification performance metrics like accuracy, f-measure, recall and precision. SVM classifier performed well according to the input datasets and give good results. In another comparison, svmfs gives better results as compared to the svm.

For future work, this input dataset can be used for the other classifiers and feature selection based on information gain can be applied to these other classifiers for the improvement of performance. Also, different classifiers can be applied by taking alternatives metrics in feature selection processes. A hybrid combination of the classifiers can be used for the input dataset and comparison can be made on that.

REFERENCES

[1] W.Lam, M.Ruiz and P.Srinivasan, Automatic text categorization and its application to text retrieval, IEEE Transaction Knowledge Data Eng., 11(6), 1999, 865-879.

[2] F.Sebastiani, Machine Learning in automated text categorization, ACM Comput. Surveys, 34(1), 2002, 1-47.

[3] Mohomed K. Elhadad, Khaled M. Badran, and Gouda I. Salama, A Novel Approach for Ontology-based Dimensionality Reduction for Web Text Document Classification, International Conference on Computer and Information Science(ICIS), 2017.

[4] S.M.G. Swati Kaur, A Survey on Dimension Reduction Techniques for Classification of multidimensional Data, International journal of Science Technology & Engineering(IJSTE), 2(12), 2016 ,31-37.

[5] Haozhe Xie, Jie Li, Qiaosheng Jhang, Yadong Wang, Comparison among dimensionality reduction techniques based on Random Projection for cancer classification, Computational Biology and Chemistry, vol.65, 2016, 165-172.

[6] D.A.Said, Dimensionality Reduction Techniques For Enhancing Automatic Text Categorization, Master Thesis, Cairo University, 2007.

[7] S.Niharika, V.Sneha Latha, D.R.Lavanya, A Survey On Text Categorization, International Journal of Computer Trends and Technology, 2012, 3(1).

[8] M.Ikonomakis, S.Kotsiantis,V.Tampakas, Text Classification Using Machine Learning Techniques, Wseas Transactions on Computers, Volume 4(8), 2005, 966-974.

[9] Mehdad, Y. and Tetreault, J., Do characters abuse more than words?, In Proceedings of the SIGdial 2016 Conference: The 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2016, 299–303.

[10] Bo Tang, Haibo He, Paul M. Baggenstoss and Steven Kay, A Bayesian Classification Approach Using Class-Specific Features for Text Categorization, IEEE Transactions On Knowledge And Data Engineering, 28(6), June 2016.

[11] Bo. Tang, H. He, ENN: Extended nearest neighbor method for pattern recognition [research frontier], IEEE Comput.Intell. Mag.,10(3), 52-60, 2015.

[12] A.Basu, C.Waters and M.Shephard, Support Vector Machines for Text Categorization, Proceeding of the 36th Annual Hawaii International Conference on System Sciences, 2003.

[13] T.Joachins, Text categorization with support vector machines: Learning with many relevant features, In Proc.10th European Conference Machine Learning, 137-142, 1998.

[14] D.Lewis, Naïve Bayes at Forty: The Independence Assumption in Information Retrieval, Proceedings of the 10th European Conference on Machine Learning (ECML-98), 1998.

[15] Y.H.Li and A.K. jain, Classification of text documents, The Compt. J., 41(8), 1998, 537-546.

[16] Deepanshu, Ramesh Kait, "A Technical Review: Text Classifications and Related approaches" paper published in the preceding of International Conference on Science & Technology: Trends and Challenges (ICSTTC-2018).