I J C T A, 10(9), 2017, pp. 103-111 ISSN: 0974-5572

Particle Swarm Optimization based Feature Selection in Time Series Data classification

Tapas Ranjan Baitharu[1] **and Subhendu Kumar Pani**[2]*

ABSTRACT

Feature selection reduces the dimensions and simplifies the data. It leads to better performance of the classification models in terms of their predictive or descriptive accuracy, diminishing of computing time needed to build models as they learn faster, and better understanding of the models. Data classification is an important major task in KDD (knowledge discovery in databases) process. It has many potential applications. The performance of classifiers is robustly dependent on the data set used for learning. The feature selection is an important step in building an effective and efficient classifier. It is process that chooses an optimal subset of features according an objective function.. In this paper, a comparative analysis of data classification using feature selection is presented. Several search techniques are considered in the study for feature selection and are applied to pre-process the dataset. The predictive performances of popular classifiers are compared quantitatively. The Diabetes Time series dataset is available at UCI machine learning repository.

Keywords: Feature selection, knowledge discovery, classification, machine learning.

1. INTRODUCTION

Data and information have become major assets for most of the organizations. The success of any organisation depends largely on the extent to which the data acquired from business operations is utilised. In other words, the data serves as an input into a strategic decision making process, which could put the business ahead of its competitors. Also, in this era, where businesses are driven by the customers, having a customer database would enable management in any organisation to determine customer behaviour and preference in order to offer better services and to prevent losing them resulting better business. The data needed that will serve as an input to organizational decision-making process is generated and warehoused[1,2]. It is being collected via many sources, such as the point of sales transactions, surveys, through the internet logs – cookies, etc. This has resulted in huge databases which have valuable knowledge hidden in them and may be difficult to extract. Data mining has been identified as the technology that offers the possibilities of discovering the hidden knowledge from these accumulated databases. Techniques such as pattern recognition and classification are the most important in data mining [6].

The task of recognition and classification is one of the most frequently encountered decision making problems in daily activities. A classification problem occurs when an object needs to be assigned into a predefined group or class based on a number of observed attributes, or features, related to that object. Humans constantly receive information in the form of patterns of interrelated facts, and have to make decisions based on them. When confronted with a pattern recognition problem, stored knowledge and past experience can be used to assist in making the correct decision. Indeed, many problems in various domains such as financial, industrial, technological, and medical sectors, can be cast as classification problems.

[1] Associate Professor, Dept. of CSE, OEC, BPUT, Odisha, India, *Email: trbaitharu2001@yahoo.co.in*

[2] Associate Professor, Dept. of CSE, OEC, BPUT, Odisha, India, *Email: skpani.india@gmail.com*

* Corresponding Author: Dr. Subhendu kumar Pani

Examples include bankruptcy prediction, credit scoring, machine fault detection, medical diagnosis, quality control, handwritten character recognition, speech recognition etc[3,5]. Pattern recognition and classification has been studied extensively in the literature. In general, the problem of pattern recognition can be posed as a two-stage process:

- Feature selection which involves selecting the significant features from an input pattern[22].
- Classification which involves devising a procedure for discriminating the measurements taken from the selected features, and assigning the input pattern into one of the possible target classes according to some decision rule.

Research efforts dedicated to data mining, which focussed on improving the classification and prediction accuracy, have recently been undergoing a tremendous change [7], [8], [9]]. The continuous development of more and more sophisticated classification models through commercial and software packages have turned out to provide some benefits only in specific problem domains where some prior background knowledge or new evidence can be exploited to further improve classification performance. In general however, related research proves that no individual data mining technique has been shown to deal well with all kinds of classification problems. Awareness of these imperfections of individual classifiers has called for the emergence of careful development and evaluation strategies of data mining classification models.

The rest of the paper is structured as follows.

In the next section, different classifiers as a means of feature selection is described. Section 3 provides feature selevtion algorithm.. Section 4 presents the study and summarizes the results. Then the paper concludes in section 5.

2. CLASSIFIER SELECTION

We select five commonly used classifiers for prediction classification in our work based on their qualitative performance. These classifiers are described in this section and their WEKA names are given in Table-3.1.

- *K-Nearest Neighbour*: This classifier is considered as a statistical learning algorithm and it is extremely simple to implement and leaves itself open to a wide variety of variations. In brief, the training portion of nearest-neighbour does little more than store the data points presented to it. When asked to make a prediction about an unknown point, the nearest-neighbour classifier finds the closest training-point to the unknown point and predicts the category of that training point according to some distance metric. The distance metric used in nearest neighbour methods for numerical attributes can be simple Euclidean distance.
- *Decision Tree*: A decision tree partitions the input space of a dataset into mutually exclusive regions, each of which is assigned a label, a value or an action to characterize its data points. The decision tree mechanism is transparent and we can follow a tree structure easily to see how the decision is made. A decision tree is a tree structure consisting of internal and external nodes connected by branches. An internal node is a decision making unit that evaluates a decision function to determine which child node to visit next. The external node, on the other hand, has no child nodes and is associated with a label or value that characterizes the given data that leads to its being visited. However, many decision tree construction algorithms involve a two - step process. First, a very large decision tree is grown. Then, to reduce large size and over-fitting the data, in the second step, the given tree is pruned. The pruned decision tree that is used for classification purposes is called the classification tree. A popular decision tree algorithm is C4.5. It can help not only to make accurate predictions from the data but also to explain the patterns in it. It deals with the problems of

the numeric attributes, missing values, pruning, estimating error rates, complexity of decision tree induction, and generating rules from trees [11]. In terms of predictive accuracy, C4.5 performs slightly better than CART and ID3 [10]. C4.5's successor, C5.0, shows marginal improvements to decision tree induction but not enough to justify its use. The learning and classification steps of C4.5 are generally fast [12]. However, scalability and efficiency problems, such as the substantial decrease in performance and poor use of available system resources, can occur when C4.5 is applied to large data sets.

- *Bayesian Networks*: This classifier is a powerful probabilistic representation, and its use for classification has received considerable attention. This classifier learns from training data the conditional probability of each attribute Ai given the class label C. Classification is then done by applying Bayes rule to compute the probability of C given the particular instances of A1…..An and then predicting the class with the highest posterior probability. The goal of classification is to correctly predict the value of a designated discrete class variable given a vector of predictors or attributes. In particular, the Naive Bayes classifier is a Bayesian network where the class has no parents and each attribute has the class as its sole parent. Although the naive Bayesian (NB) algorithm is simple, it is very effective in many real world datasets because it can give better predictive accuracy than well-known methods like C4.5 and BP [13],[14] and is extremely efficient in that it learns in a linear fashion using ensemble mechanisms, such as bagging and boosting, to combine classifier predictions [15]. However, when attributes are redundant and not normally distributed, the predictive accuracy is reduced [16].
- *Neural Network*: Back-Propagation (BP) Neural Networks can process a very large number of instances; have a high tolerance to noisy data; and has the ability to classify patterns which they have not been trained [37]. They are an appropriate choice if the results of the model are more important than understanding how it works [17]. However, the BP algorithm requires long training times and extensive testing and retraining of parameters, such as the number of hidden neurons, learning rate and momentum, to determine the best performance [18].
- *Support Vector Machine*: Support vector machines exist in different forms, linear and non-linear. A support vector machine is a supervised classifier. What is usual in this context, two different datasets are involved with SVM, training and a test set. In the ideal situation the classes are linearly separable. In such situation a line can be found, which splits the two classes perfectly. However not only one line splits the dataset perfectly, but a whole bunch of lines do. From these lines the best is selected as the "separating line". The best line is found by maximizing the distance to the nearest points of both classes in the training set. The maximization of this distance can be converted to an equivalent minimization problem, which is easier to solve. The data points on the maximal margin lines are called the support vectors. Most often datasets are not nicely distributed such that the classes can be separated by a line or higher order function. Real datasets contain random errors or noise which creates a less clean dataset. Although it is possible to create a model that perfectly separates the data, it is not desirable, because such models are over-fitting on the training data. Over-fitting is caused by incorporating the random errors or noise in the model. Therefore the model is not generic, and makes significantly more errors on other datasets. Creating simpler models keeps the model from over-fitting. The complexity of the model has to be balanced between fitting on the training data and being generic. This can be achieved by allowing models which can make errors. A SVM can make some errors to avoid over-fitting. It tries to minimize the number of errors that will be made. Support vector machines classifiers are applied in many applications. They are very popular in recent research. This popularity is due to the good overall empirical performance. Comparing the naive Bayes and the SVM classifier, the SVM has been applied the most. WEKA names of selected classifiers is shown in Table 1.

Table 1
WEKA names of selected classifiers

Generic Name	WEKA Name
Bayesian Network	Naïve Bayes (NB)
Neural Network (NN)	Multilayer Perceptron
Support Vector Machine	SMO
C4.5 Decision Tree	J48
K-Nearest Neighbour	1Bk

3. FEATURE SELECTION ALGORITHMS IN WEKA

WEKA provides several feature selection algorithms which are described in Table 2.

Table 2: Feature Section Algorithms

Name (in WEKA)	*Description*
Cfs Subset Eval	It evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy between them. Subsets of features that are highly correlated with the class while having low inter-correlation are preferred It identifies locally predictive attributes. Iteratively adds attributes with the highest correlation with the class as long as there is no attribute in the subset that has a higher correlation with the attribute in question.
Chi Squared Attribute Eval	Evaluates the worth of an attribute by computing the value of the chi-squared statistic with respect to the class.
Consistency Subset Eval	Evaluates the worth of a subset of attributes by the level of consistency in the class values when the training instances are projected onto the subset of attributes. Consistency of any subset can never be lower than that of the full set of attributes; hence the usual practice is to use this subset evaluator in conjunction with a random or exhaustive search which looks for the smallest subset with consistency equal to that of the full set of attributes.
Info Gain Attribute Eval	Evaluates the worth of an attribute by measuring the information gain with respect to the class. InfoGain(Class, Attribute) = H(Class) - H(Class | Attribute).

4. EXPERIMENT DESIGN AND RESULT ANALYSIS

We follow a methodical approach to conduct the study which is described in the following subsections.

4.1. WEKA Tool

We use WEKA (www.cs.waikato.ac.nz/ml/weka/), an open source data mining tool for our experiment. WEKA is developed by the University of Waikato in New Zealand that implements data mining algorithms using the JAVA language. WEKA is a state-of-the-art tool for developing machine learning (ML) techniques and their application to real-world data mining problems. It is a collection of machine learning algorithms for data mining tasks. The algorithms are applied directly to a dataset. WEKA implements algorithms for data pre-processing, feature reduction, classification, regression, clustering, and association rules. It also includes visualization tools. The new machine learning algorithms can be used with it and existing algorithms can also be extended with this tool.

4.2. Performance Measure

We use different metrics for comparing the classifiers' predictive performance in our experiment. These are presented below:

Confusion Matrix: The columns of the confusion matrix represent the predictions, and the rows represent the actual class. Correct predictions always lie on the diagonal of the matrix. Given below is the general structure of confusion matrix.

TP FN

FP TN

where in, True Positives (TP) indicate the number of instances of the minority that were correctly predicted, True Negatives (TN) indicate the number of instances of the majority that were correctly predicted. False Positives (FP) indicate the number of instances of the majority that were incorrectly predicted as minority class instances and False Negatives (FN) indicate the number of the minority that were incorrectly predicted as majority class instances. Though the confusion matrix gives a better outlook on how the classifier performed than accuracy, a more detailed analysis is preferable which are provided by the further metrics.

Recall: Recall is a metric that gives a percentage of how many of the actual minority class members the classifier correctly identified. (TP + FN) represent a total of all minority members. Recall is given below

$$\text{Recall} = TP / (TP + FN)$$

Precision: It gives us the total the percentage of how many of minority class instances as determined by the model or classifier actually belong to the minority class. (TP + FP) represents the total of positive predictions by the classifier.

Precision is given by

$$\text{Precision} = TP / (TP + FP)$$

Thus in general it is said that Recall is a Completeness Measure and Precision is an Exactness Measure. The ideal classifier would give value as 1 for both Recall and Precision but if the classifier gives higher (closer to one) for one of the above metrics and lower for the other metrics in that case choosing the classifier is difficult task. In such cases some other metrics as discussed further are suggested in the literature.

F-Measure: It is a harmonic mean of Precision & Recall. We can say that it is essentially an average between the two percentages. It really simplifies the comparison between the classifiers. It is given by

$$\text{F-Measure} = 2 / (1/\text{Recall} + 1/\text{Precision})$$

4.3. Dataset Description

We performed computer simulation on a Diabetes dataset available UCI Machine Learning Repository [4]. It contains 768 instances and 8 input features as well as 1 output feature. Features in the Dataset in presented by table 3.

Table 3
Features in the Dataset

Feature No.	*Description*
1	Number of times pregnant
2	Plasma glucose concentration a 2 hours in an oral glucose tolerance test
3	Diastolic blood pressure (mm Hg)
4	Triceps skin fold thickness (mm)
5	2-Hour serum insulin (mu U/ml)
6	Body mass index (weight in kg/(height in m)^2)
7	Diabetes pedigree function
8	Age (years)
9	Output variable(0 or 1)

4.4. Experiment Design

In the study, we use Weka data mining tool to conduct the experiment. We compared the classification performance of the chosen models employing feature selection using particle swarm optimization against feature selection using genetic algorithm. Genetic Algorithm based search method is bundled with Weka tool while Particle Swarm Optimization based search method is not bundled with Weka. We integrated a publicly available PSO based search algorithm [19, 20] with Weka in order to perform the study.

We used CfsSubSetEval, as the attribute evaluator. This algorithm in Weka, evaluates the worth of a subset of attributes by considering the individual predictive ability of each feature along with the degree of redundancy among them. Subsets of features that are highly correlated with the class while having low inter-correlation are preferred. The parameters used in this attribute evaluator as well as search methods are kept at its default values.

We use 10-fold cross validation as the test mode to record classification accuracy. This approach is suitable to avoid biased results and provide robustness to the classification. Also, the parameters of a classification algorithm are chosen to their default values[21].

The following steps have been applied to generate experimental data in order to draw inference:

1. Find classification performance of the classifiers with original features in the dataset.
2. Find classification performance after selecting feature subset using GA based search.
3. Find classification performance after selecting feature subset using PSO based search.

4.5. RESULTS ANALYSIS

Following the experimental procedures described in the previous section, we performed several runs in Weka tool and gathered the data for the inference. Table 4 summarizes the classification accuracy in

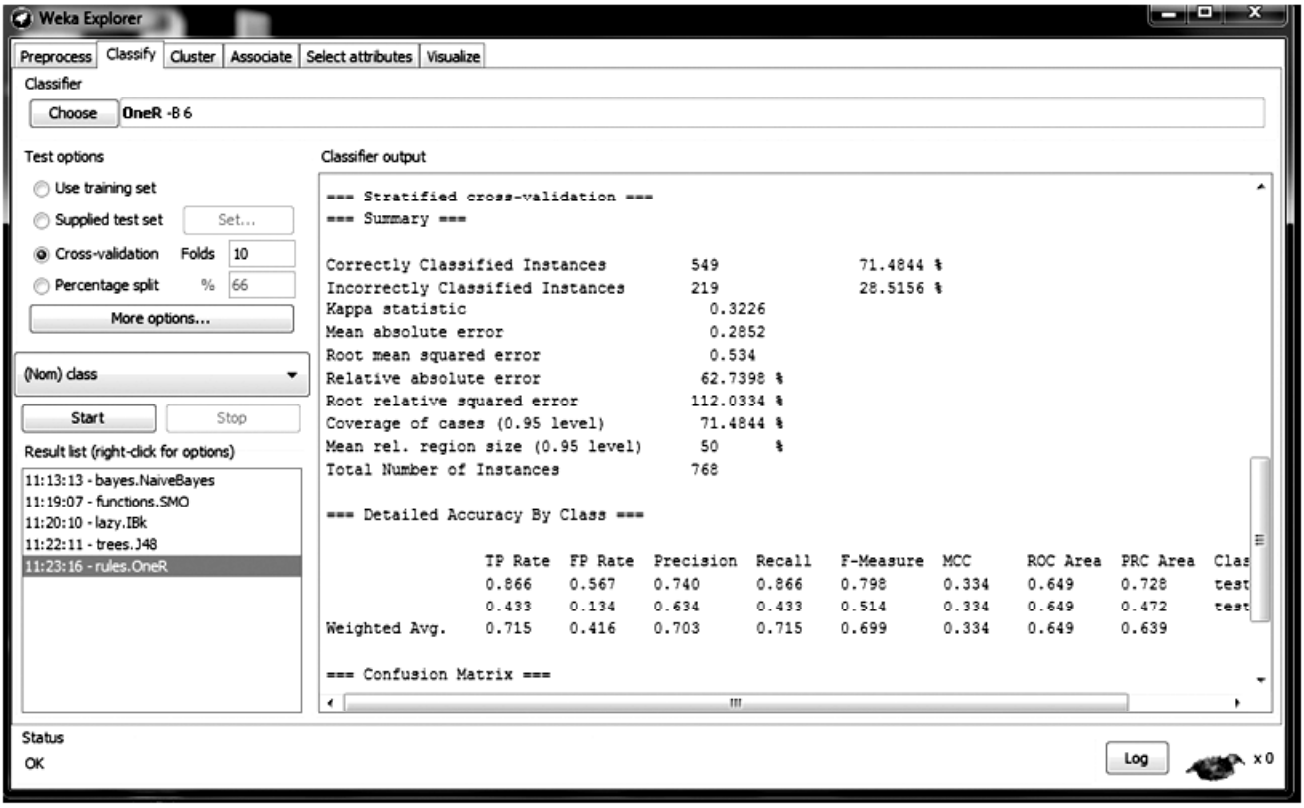

Figure 1: Snapshot of Classifiers in WEKA

percentage of all the classifiers across the dataset with original features while Table 5 provides the classification performance after a genetic algorithm based feature selection. Table 6 shows the classification performance of the classifiers when feature selection is performed using PSO based search. A Snapshot of Classifiers in WEKA is shown in Figure 1.

Table 4
Classification Accuracy in % with Original Features

Classifiers	*Classification Results with Original Features*			
	ROOT Mean Squared Error	*F-Measure*	*ROC Area*	*Accuracy*
NB	0.4168	0.760	0.819	76.30
SMO	0.476	0.763	0.720	77.34
IBK	0.5453	0.698	0.650	70.18
J48	0.4463	0.736	0.751	73.82
OneR	0.534	0.699	0.649	71.48

Table 5
Classification Accuracy in % after GA-based Features Selection

Classifiers	*Classification Results After GA –Based Feature Selection*			
	ROOT Mean Squared Error	*F-Measure*	*ROC Area*	*Accuracy*
NB	0.4008	0.769	0.829	77.474
SMO	0.4814	0.757	0.711	76.8229
IBK	0.5617	0.683	0.658	68.35
J48	0.4216	0.743	0.791	74.86
OneR	0.534	0.699	0.649	71.4844

Table 6
Classification Accuracy in % after PSO-based Features Selection

Classifiers	*Classification Results After PSO –Based Feature Selection*			
	ROOT Mean Squared Error	*F-Measure*	*ROC Area*	*Accuracy*
NB	0.4008	0.829	0.823	78.58
SMO	0.4814	0.711	0.691	76.29
IBK	0.5617	0.658	0.644	69.594
J48	0.4216	0.791	0.777	74.98
OneR	0.534	0.649	0.639	72.84

Given the dataset, it is evident from the Figure 2 that the performance of the classifiers on feature reduction (or selection) is not uniform. Performance of SMO and k-Nearest Neighbour decreases with feature reduction. Similarly, performance of OneR ,Naive Bayes and Decision Tree increases with feature reduction .Data reduction and, in particular, feature reduction are important steps in the knowledge discovery in database (KDD) process. In general, it improves the predictive capability of the classifiers. We tested the effect of feature reduction on a Diabetes Dataset from UCI repository. The results from preliminary computer simulation are mixed and indicate that data feature reduction is not very beneficial in this case. More experiments with different datasets and different feature-scenarios are needed for validation of the results.

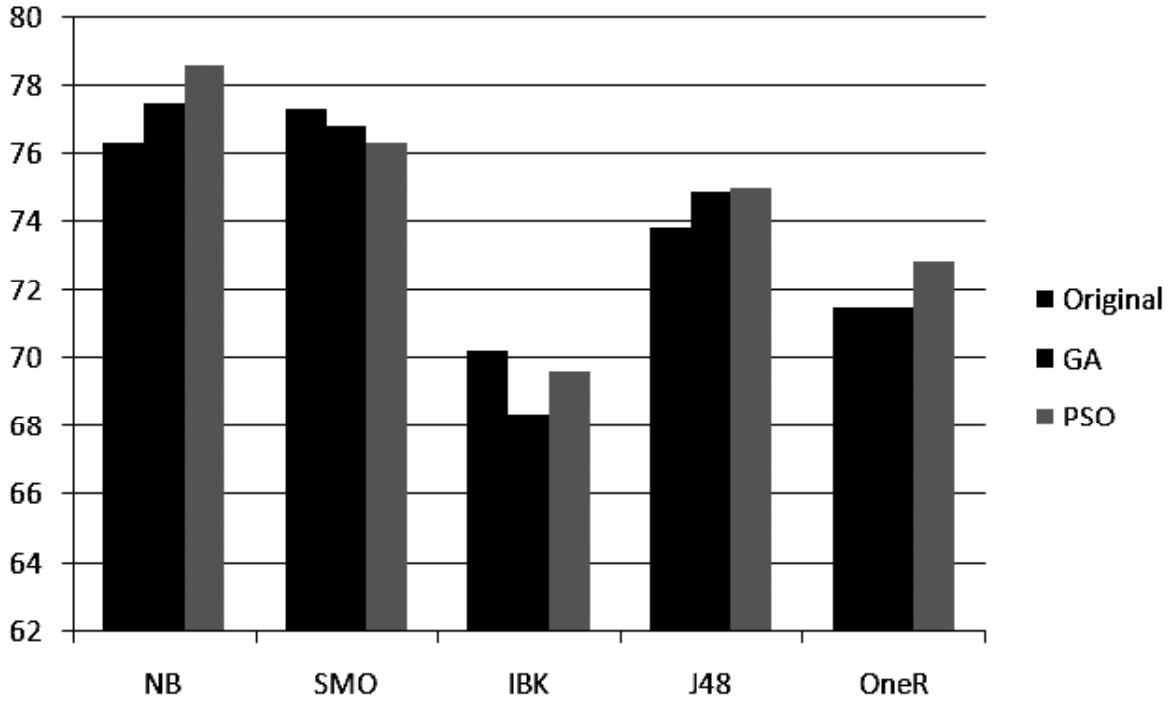

Figure 2: Classifiers Performance

We conducted an experiment to find the impact of feature selection on the predictive performance of different classifiers. We select five popular classifiers considering their qualitative performance for the experiment. We also choose 2 feature selection algorithms and classification accuracy for a given Diabetes dataset available at UCI machine learning repository. It is observed in the tabulated data that the performance of all the classifiers is not linear across the datasets on the feature selection. The classifiers perform at least same or better when PSO based feature selection is performed in comparison to GA based feature selection. This is depicted in Figure4.2.

5. CONCLUSION AND FUTURE WORK.

Feature selection is a dynamic field and an important activity in data mining techniques. This paper attempts to survey this fast developing field, show some effective applications, and point out interesting trends and challenges. We conducted an experiment to compare five most commonly used classification models to classify the data taken from UCI machine learning repository. Popular feature selection techniques were used to select features and develop feature-subset scenarios based on which classifiers predictive performance was recorded quantitatively. The experimental data showed mixed result and indicate that feature reduction is not very beneficial in this case. While some classifier display improved performance, others give poor results. The general idea that the predictive performance of a classifier is better with feature selection could not be re-established nor refuted.

We analysed the performance of most popular classifiers based on feature selection. The experimental data show mixed results. It could not refute the general idea that classifiers show improved performance with feature selection nor re-established it. The experiment considered a single dataset. We propose to extend our work by considering multiple datasets drawn from different domains, so that the results will be sound enough for generalization.

REFERENCES

[1] Han J. and Kamber M., Data mining concept and techniques**,** Morgan Kaufmann Publishers, London, 2001.

[2] Klosgen W and Zytkow J M (eds.), Handbook of data mining and knowledge discovery, OUP, Oxford, 2002.

[3] Liu, H. Feature Extraction, Construction and Selection: A Data Mining Perspective, ISBN0-7923-8196-3, Kluwer Academic Publishers, 1998.

[4] UCI Machine Learning Repository, Available at http://archive.ics.uci.edu/ml/machine-learning-databases/statlog/german.

[5] Kantardzic M., Data Mining: Concepts, Models, Methods, and Algorithms, Wiley, 2003.

[6] Berry, M. and Linoff, G., Data mining techniques, Wiley Publishing, Inc, 2004.

[7] Lui, H. Li, J. and Wong, L., A Comparative Study on Feature Selection and Classification Methods Using Gene Expression Profiles and Proteomic Patterns. Genome Informatics Vol. 13 p51-60, 2002.

[8] Caruana, R. and Mizil, A. N, Data Mining in Metric Space: An Empirical Analysis of Supervised Learning Performance Criteria Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 69-78, 2004.

[9] Caruana, R. and Mizil, A. N., An empirical comparison of supervised learning algorithms. Proceedings of the 23rd international conference on machine learning ICML, 2006.

[10] Juan Zhang, Xuegang Hu, Yuhong Zhang, and Pei-Pei Li. An efficient ensemble method for classifying skewed data streams. In De-Shuang Huang, Yong Gan, Prashan Premaratne, and Kyungsook Han, editors, ICIC (3), volume 6840 of Lecture Notes in Computer Science, pages 144-151. Springer, 2011.

[11] Witten I and Frank E. Data Mining: Practical Machine Learning Tools and Techniques with Java, Morgan Kauffman Publishers, California, USA, 1999.

[12] J. Han, M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, 2001.

[13] Domingos P and Pazzani M. "Beyond Independence: Conditions for the Optimality of the Simple Bayesian Classifier", in Proceedings of the 13th Conference on Machine Learning, Bari, Italy, pp105-112, 1996.

[14] Elkan C. Magical Thinking in Data Mining: Lessons From CoIL Challenge 2000, Department of Computer Science and Engineering, University of California, San Diego, USA, 2001.

[15] Elkan C. Naive Bayesian Learning, Technical Report CS97-557, Department of Computer Science and Engineering, University of California, San Diego, USA, 1997.

[16] Witten I and Frank E. Data Mining: Practical Machine Learning Tools and Techniques with Java, Morgan Kauffman Publishers, California, USA, 1999.

[17] Berry M and Linoff G. Mastering Data Mining: The Art and Science of Customer Relationship Management, John Wiley and Sons, New York, USA, 2000.

[18] Bigus J. Data Mining with Neural Networks, McGraw Hill, New York, USA, 1996.

[19] Moraglio, A., Di Chio, C., and Poli, R. (2007), Geometric Particle Swarm Optimization, EuroGP 2007, LNCS 445, pp. 125-135.

[20] WekaPSO (2015), PSOSearch: An implementation of the Particle Swarm Optimization (PSO) algorithm to explore the space of attributes, available at: http://weka.sourceforge.net/packageMetaData/PSOSearch/, accessed on December 15, 2015.

[21] Subhendu Kumar Pani and Satya Ranjan Biswal and Santosh Kumar Swain, A Data Mining Approach to Identify Key Factors for Systematic Reuse. The IUP Journal of Information Technology, Vol. VIII, No. 2, June 2012, pp. 24-34. Available at SSRN: http://ssrn.com/abstract=2169262

[22] Subhendu Kumar Pani and Amit Kumar and Maya Nayak, || Performance Analysis of Data Classification Using Feature Selection || (October 24, 2013). The IUP Journal of Information Technology, Vol. IX, No. 2, June 2013, pp. 36-50.