# A New Approach for Handling Null Values in Web Log Using KNN and Tabu Search KNN

Yogendra Kumar Jain[1] and Vivek Suryawanshi[2]

[1]Head of computer Science and Engineering department, SATI, Vidisha, (M.P.) India
ykjain_p@yaho.co.in
[2]Research Scholar M.Tech CSE department, SATI, Vidisha, (M.P.) India
vivek.suryawanshi@rediffmail.com

## *Abstract*

*When the data mining procedures deals with the extraction of interesting knowledge from web logs is known as Web usage mining. The result of any mining is successful, only if the dataset under consideration is well preprocessed. One of the important preprocessing steps is handling of null/missing values. Handlings of null values have been a great bit of test for researcher. Various methods are available for estimation of null value such as k-means clustering algorithm, MARE algorithm and fuzzy logic approach. Although all these process are not always efficient.*

*We propose an efficient approach for handling null values in web log. We are using a hybrid tabu search – k nearest neighbor classifier with multiple distance function. Tabu search – KNN classifier perform feature selection of K-NN rules. We are handling null values efficiently by using different distance function. It is called Ensemble of function. It gives different set of feature vector. Feature selection is useful for improving the classification accuracy of NN rule. We are using different distance metric with different set of feature, so it reduces the possibility that some error will common. Therefore, proposed method is better for handling null values.*

*The proposed method is using hybrid classifier with different distance metrics and different feature vector. It is evaluated using our MANIT database. Results have indicated that a significant increase in the performance when compared with simple K-NN classifier.*

## *Keywords*

*Web Log, Null Value, KNN, Tabu Search*

## 1. Introduction

Web mining has recently become a major field for research and applications development. It is a technique of finding or extracting useful knowledge from web logs or user history. Web log is a text file that is created automatically by server that records each page requests. In Web usage analysis, data preprocessing includes tasks of repairing erroneous data and treating missing values [1]. The pre-processing of web usage data, which is mainly web logs, is usually complex and time consuming. It can take up to 80% of the time spend analyzing the data.

Missing values attribution is an actual yet challenging issue confronted in machine learning and data mining [2]. Missing values may generate bias and affect the quality of the supervised learning process or the performance of classification algorithms [2]. However, most learning algorithms are not well adapted to some application domains due to the difficulty with missing

values e.g. Web applications. The existing algorithms are designed with the assumption that there are no missing values in datasets. Hence, there is necessity of a reliable method for dealing with those missing values. Generally, dealing with missing values means to find an approach that can fill them and maintain (or approximate as closely as possible) the original distribution of the data.

In this paper, we are using a hybrid tabu search k nearest neighbor classifier with different distance function. The K-nearest neighbors of an unknown sample are selected from the training set in order to predict the class label as the most frequent one occurring in the K-neighbors [3]. Feature selection technique is useful for improving the classification accuracy of the K-NN rule [4].

The term feature selection refers to algorithms that select the best subset of the input feature set. The choice of an algorithm for selecting the features from an initial set depends on N (number of feature) [5]. The feature selection problem is of small scale, medium scale, or large scale if n belongs to range (0-19), (20- 49), or (50- ∞), respectively [5]. Modern iterative heuristics such as Tabu Search and genetic algorithms have been found effective in tackling this category of problems which have an exponential and noisy search space with numerous local optima [5].

In this work, the Tabu Search performs the feature selection in combination with different distance function as an objective function. This objective function is used to evaluate the classification performance of each subset of the features selected by the TS. Feature selection vector in TS is represented by a 0/1bit string where 0 indicates the feature is not included in the solution while 1 indicates the feature is included [5]. The remainder of this paper is organized as follows: Section 2 is about related work, Section 3 describes proposed hybrid approach using multiple distance function. Section 4 is about experimental result and analysis. Section 5 gives the conclusion.

## 2. Related Work

Zhang et al. [2] proposed a method of imputation. It is a popular strategy. In comparison to other methods, it uses as more information as possible from the observed data to predict missing values. Chen and Chen [6] presented a method for estimation of null values, where a fuzzy similarity matrix is used to represent fuzzy relations. The method is used to deal with one missing value in an attribute.

Chen and Huang [7] constructed a genetic algorithm to impute in relational database systems. The machine learning methods also include auto associative neural network, decision tree imputation, and so on.

Kuncheva and Jain [8] have attempted to combine feature subset variance with different types of classifiers and then optimize the resulted population with genetic algorithm. They encapsulate all the features of an ensemble in only one chromosome. It is stated that this increases the search space and thus the time of optimization, besides limiting the ability to extend algorithm to include other diversification techniques.

Liu et al. [10] have utilized feature subset selection to create a primary set of KNN classifiers, and then have used MOGA to optimize this initial population. Furthermore, it exploit genetic algorithm to optimize the weights of features used in an AdaBoost of KNN classifiers, combining structure manipulation with hypothesis space traversal.

Langdon and Buxton [11] reached an optimal way to combine different classifiers (i.e. set of accessible hypothesis) using genetic programming with classifiers as functions and their thresholds as the terminals of genotypes.

Opitz [12] investigates ensemble feature selection by generating an initial set of Neural Networks with different input feature subsets (i.e. a method in the second category), each of which are encoded in a representative chromosome, and then evolves the population with genetic algorithm operators (i.e. a method in the third category).

Tahir and Ahmed [3] proposed a novel hybrid approach for simultaneous feature selection and feature weighting of K-NN rule based on Tabu Search (TS) heuristic. The proposed TS heuristic in combination with K-NN classifier is compared with several classifiers on various available data sets. The results have indicated a significant improvement in the performance in classification accuracy. The proposed TS heuristic is also compared with various feature selection algorithms. Experiments performed revealed that the proposed hybrid TS heuristic is superior to both simple TS and sequential search algorithms.

But, these methods are not a completely satisfactory way to handle missing value problems and all of these are pre-replacing methods [13]. First, these methods only are designed to deal with the discrete values and the continuous ones are discredited before imputing the missing value, which may lose the true characteristic during the converting process from the continuous value to discredited one. Secondly, these methods usually studied the problem of missing covariates (conditional attributes) [13].

The experimental results have shown that the Tabu Search not only has a high possibility to obtain the optimal or near-optimal solution for handling null values, but also requires less computational effort than the other suboptimal and genetic algorithm based methods.

## 3. Proposed Work

In this paper, a Hybrid Tabu Search/K-NN algorithm is proposed to perform feature selection with the objective of improving the classification accuracy. This approach uses feature vector on the encoding solution of Tabu Search. The feature vector consists of real values, while feature binary vector consisting of either 0 or 1. K-NN classifier is used to evaluate each value evolved by TS. In addition to feature and binary vectors, the value of K used in K-NN classifier is also stored in the encoding solution of TS. Neighbors are calculated using a squared Euclidean distance defined as:

$$D(x, y) = \sum_{i=1}^{m} (x_i - y_i)^2$$

Where x and y are two input vectors and m is the number of features.

In the proposed approach, the feature binary vector value can be 0 for some features. Thus, the feature space is expanded in the dimensions associated with highly fruitful features. This allows the K-NN classifier to distinguish null values more accurately. The classification accuracy obtained from TS/K-NN classifier for null value handling can be increases by using different distance function.

Fig. 1 shows the training phase of proposed hybrid model of TS/K-NN classifier. Major steps of proposed method are given below. Flow diagram is shown in figure 1.

**Step 1: Original Feature Vector (V)**

A feature vector is an n-dimensional vector of numerical features that represent some object. Many algorithms in machine learning require a numerical representation of objects, since such representations facilitate processing and statistical analysis.Features typically correspond to other words that co-occur with the characterized word in the same context. It is then assumed that different words that occur within similar contexts are semantically similar. Once feature vectors have been constructed, the similarity between two words is defined by some vector similarity metric.

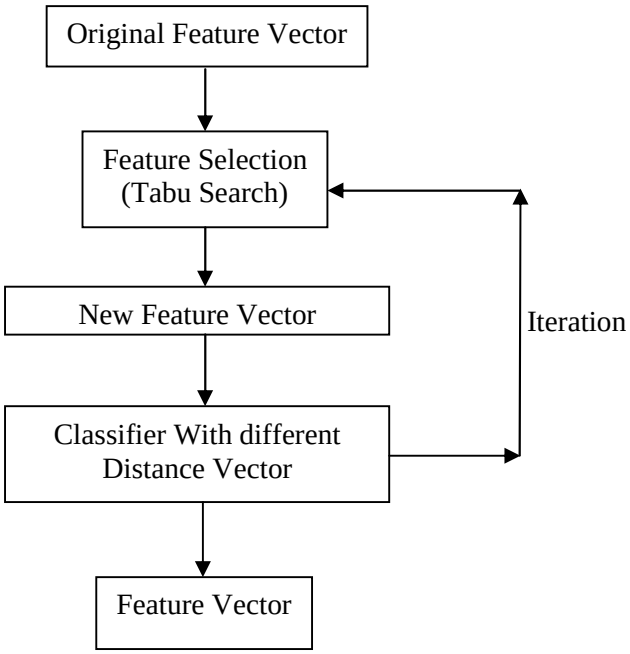


Figure 1: Training phase of proposed hybrid TS/K-NN classifier with different distance function

**Step 2: Feature Selection**

The term feature selection refers to the selection of the best subset of the input feature set. These methods used in the design of pattern classifiers have three goals:

1. To reduce the cost of extracting the features.
2. To improve the classification accuracy.
3. To improve the reliability of the estimation of the performance.

Since a reduced feature set requires less training samples in the training process of a pattern classifier Feature selection produces savings in the measuring features (since some of the features are discarded) and the selected features retain their original physical interpretation. A number of algorithms have been proposed for feature selection to obtain near-optimal solution [14]. The choice of an algorithm for selecting the features from an initial set depends on *n*. In this paper we used tabu search for this purpose.

**Step 3: New feature vector ($V_{new}$)**

Some of the features may not be interested or not qualify threshold value. In this paper feature selection is performed using tabu search which gives most fruitful direction. It helps to find most rich and related data classes. It can improve overall performance by decreasing search area. In this step, we get new feature vector ($V_{new}$).

**Step 4: Classification with multiple distance function**

In this section, we will discuss about the purpose multiple distance function significance in null value evaluation. There is n number of solutions are to find feature vectors for various distance measure in which errors are not correlated [1]. Since errors are not correlated so it can achieve significant performance improvement for null value handling.

**Step 5: Output Feature Vector**

This is the final feature vector which is used for training purpose shown in figure 2. The feedback from the K-NN classifier allows the Tabu Search to iteratively search for a feature vector that improves the classification accuracy. In the testing phase, only K-NN classifier is used as shown in Fig. 2.

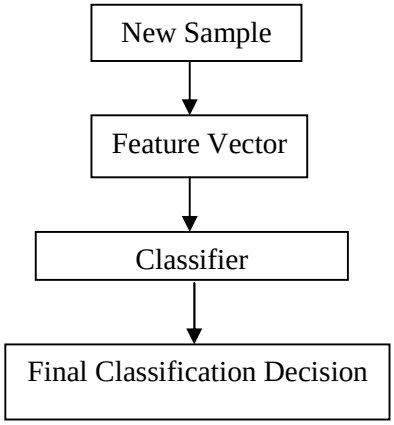


Figure 2: Testing Phase

## 3.1 Multiple Distance Function

In this section, we will discuss about the purpose of multiple distance function significance in null value evaluation. Figure 3 clearly depicted that the M (best neighbors) are selected from N neighbors during each iteration. There is n number of solutions are available to find out the feature vectors, for various distance measure, in which errors are not correlated [5]. Since errors are not correlated, so it can achieve significant performance improvement for null value handling. By using n distance functions, n feature vectors are obtained using TS in the training phase. The use of different distance functions, each with a potentially different set of features give errors of the individual classifiers is not correlated.

The following three distance metrics are used for NN classifier.

Squared Euclidean Distance:

$$E = \sum_{i=1}^{m} (x_i - y_i)^2$$

Manhattan Distance:

$$M = \sum_{i=1}^{m} (x_i - y_i)$$

Canberra Distance:

$$M = \sum_{i=1}^{m} (x_i - y_i) / (x_i + y)$$

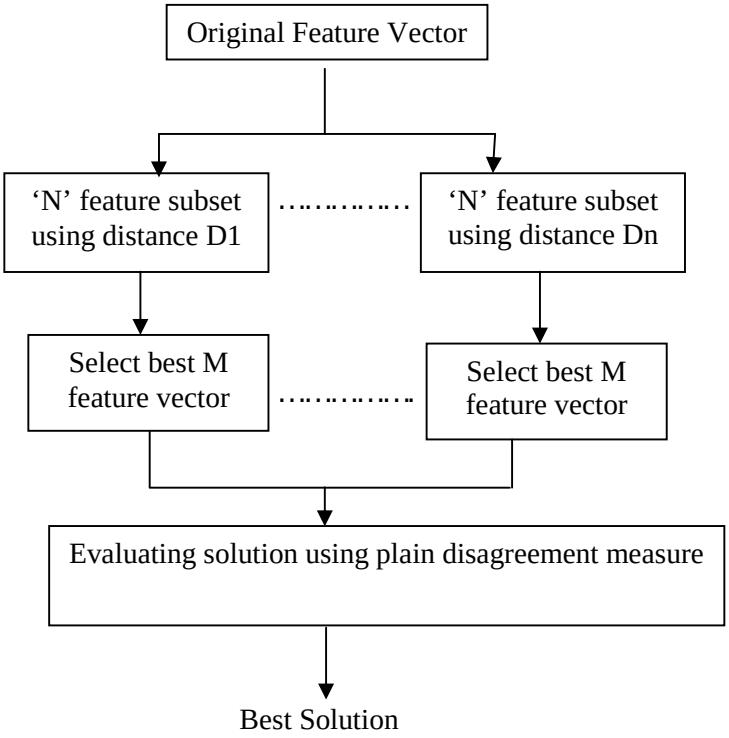


Figure 3: Working of multiple distance function

The simplest and most common form of diversity measure is the plain disagreement (which is used in our work). For two classifiers *i* and *j*, the plain disagreement is a number between 0 and 1 and equal to the proportion of the instances on which the classifiers make different predictions [15]. Other pair-wise diversity criterion is the fail/non-fail disagreement measure. This measure shows that the percentage of test instances for which the classifiers make different predictions, but for which one of them is correct [15]. Therefore, this criterion has to do with a subset of what the plain disagreement measures and is usually less than it [15].

## 3.2 Feature Selection using Tabu Search

Tabu Search (TS) was introduced by Fred Glover [16] as a general iterative meta heuristic for solving combinatorial optimization problems. Tabu Search is conceptually simple and elegant. It is a form of local neighborhood search.

| | |
|---|---|
| $\Omega$ | : set of feasible solution |
| $s$ | : Current solution |
| $s^*$ | : Best solution |
| Cost | : Objective function |
| N(S) | : Neighborhood of solution S |
| $V^*$ | : Sample of Neighborhood solution |
| T | : Tabu list |
| AL | : Aspiration level |

**Begin**

1. Start with an initial feasible solution $s \in \Omega$ .
2. Initialize Tabu list and Aspiration level
3. Fixed number of iteration Do
4. Generate Neighborhood of solution $V^* \subset N(S)$
5. Find best $s^* \in V^*$
6. If moves $s$ to $s^*$ is not in T then
7. Accept move and update best solution.
8. Update Tabu list Aspiration level
9. Increment iteration level
10. Else
11. if Cost ( $s^*$ ) < AL Then
12. Accept move and update best solution
13. Update Tabu list Aspiration level
14. Increment iteration level
15. End If
16. End If
17. End For

**End**

Figure 4: Algorithmic description of Tabu Search (TS).

**Objective Function:** Simple Voting Scheme is used in each instance of *n* classifiers. The objective function is the number of instances incorrectly classified using Simple Voting Scheme. The objective is:

$$Cost = \sum_{i=1}^{s} C_i$$

## 4. Experiment Results and Analysis

To evaluate the effectiveness of our method, extensive experiments are carried out. Comparisons with K-NN methods are also performed in this section. The tabu list size and Number of Neighborhood Solutions are determined using equation: T = N = ceil (√F), where T is the Tabu List Size, N is the number of neighborhood solutions and F is the number of features.

In K-Nearest Neighbor (K-NN) classifier, the K nearest neighbor of an unknown sample in the training set is computed in order to predict the class label as the most frequent one occurring in the K-neighbors [17].

To demonstrate the procedure, we take considered the live web log records of Maulana Azad National Institute of Technology (MANIT), Bhopal, India and perform the experiment using the proposed approach. First we parse these log records by the tool WebLogExpert [18]. The Figure 5 shows that daily search phrases for the Maulana Azad National Institute of Technology Bhopal, server. In Figure 6, we shows that some possible errors on the web log, these possible errors may create a null entry in the web log. For measuring performance, accuracy of the proposed method, we perform our operation on the 500 records of the MANIT, Bhopal, India log records. To support our methodology, we designed and implemented in MATLAB 7.8. In this experiment, we handled the null values efficiently in data preprocessing steps.

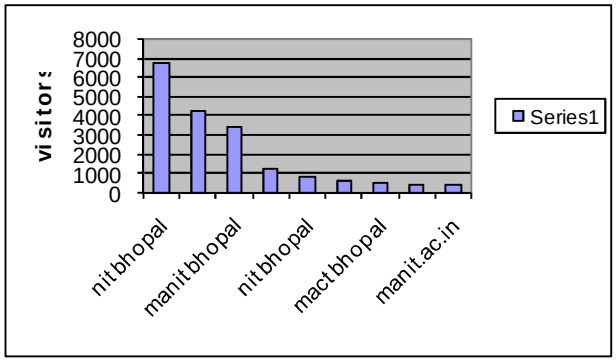


Figure 5: Top Search Phrases in our Log Records.

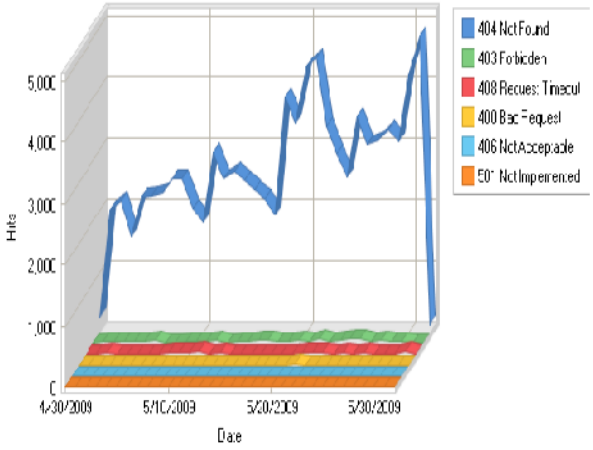


Figure 6: Daily Error Types in our Log Records

Cross-validation (*K*-fold), sometimes called rotation estimation is used in this work for analysis. One round of cross-validation involves partitioning a sample of data into complementary subsets, performing the analysis on one subset (called the training set), and validating the analysis on the other subset (called the validation set or testing set). To reduce variability, multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds.

In *K*-fold cross-validation, the original sample is randomly partitioned into *K* sub samples. Out of the *K* sub samples, a single sub sample is retained as the validation data for testing the model, and the remaining $K - 1$ sub samples are used as training data. The cross-validation process is then repeated *K* times (the *folds*), with each of the *K* sub samples used exactly once as the validation data. The *K* results from the folds then can be averaged (or otherwise combined) to produce a single estimation. The advantage of this method over repeated random sub- sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. 10-fold cross-validation is commonly used [19].

In this work, we investigate the performance aspect of hybrid tabu search-K-NN for handling null values. We present a comparative performance evaluation model of proposed hybrid approach versus simple K-NN. A quantitative measurement are performed to compare these two method using mean absolute error, root mean error, relative absolute error, root relative square error and average accuracy as performance evaluating parameters. The experimental results are shown in table1.1. This is clearly shown in result that proposed hybrid approach using multiple distance function is more suitable for null value handling. Major performance gain can be found in accuracy rate which is 74.4 % compare than 67.6 % in simple KNN. There is improvement of 4.5% in accuracy rate.

Table 1 Performance comparison

| **Parameters** | **K-NN** | **Proposed Tabu-KNN** |
|---|---|---|
| Mean Absolute Error | 8.4 | 8.0 |
| Root Mean Error | 2.89 | 2.82 |
| Relative Absolute Error | 8.4 | 8.0 |
| Root Relative Square Error | 2.9 | 2.8 |
| Average Accuracy | 67.66 | 74.4 |

The experimental results are shown in Figure 7 and figure (8).

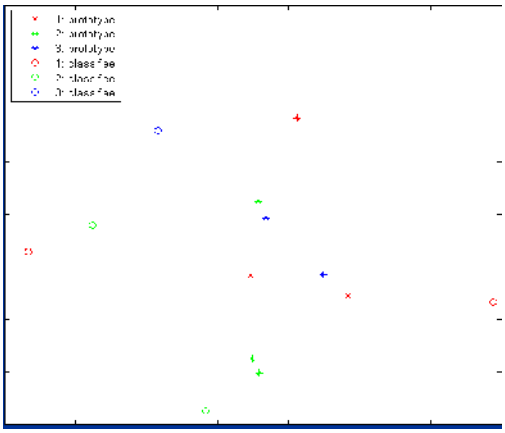

Figure 7: Nearest neighbor measurement for simple KNN using 0.2 threshold values

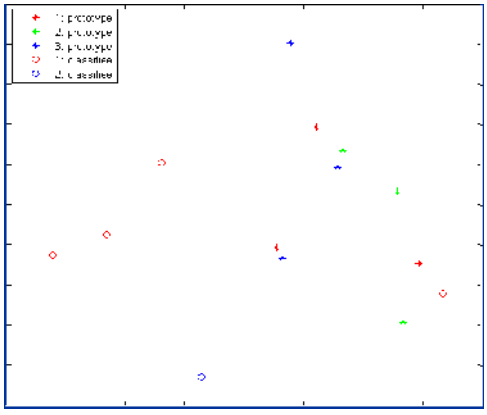

Figure 8: Nearest neighbor measurement for proposed classifier using 0.2 threshold values.

## 5. Conclusion

A new technique is proposed in this paper to improve the performance of K-nearest neighbor (KNN) classifier. The proposed method combines multiple NN classifiers, where each classifier uses a different distance function and potentially a different set of features (feature vector). These feature vectors are determined independently for each distance metric using Tabu Search (TS). To increase the diversity, simple voting scheme is introduced in the cost function of TS. The proposed classifier is evaluated using with our MANIT database. We have taken various parameters to evaluate the performance of proposed method. The results have shown that a significant increase in the performance is achieved, when compared with existing simple K-NN method. Major performance gain can be found in accuracy rate which is 74.4 % as compared with 67.6 % in simple KNN. Therefore, there is improvement in accuracy with the rate of 4.5%.

# References


[1] Cunningham, P., Carney, J., "Diversity versus quality in classification ensembles based on feature selection", Proc. Of 11th European Conf. on Machine Learning, Barcelona, Spain, LNCS, vol. 1810, Springer, pp. 109–116, 2000.

[2] Zhang, S.C., et al. "Missing is useful: Missing values in cost-sensitive decision trees", IEEE Transactions on Knowledge and Data Engineering, Vol. 17(12), pp. 1689-1693, 2005.

[3] Muhammad Atif Tahir, and Ahmed Bouridane, "Simultaneous feature selection and feature weighting using Hybrid Tabu Search/K-nearest neighbor classifier", Science direct: Pattern Recognition Letters, Vol. 28, pp. 438–446, 2007.

[4] Raymer, M. L. et al., "Dimensionality Reduction using Genetic Algorithms", IEEE Transaction on Evolution Computing, Vol. 4 (2), pp. 164–171, 2000.

[5] Muhammad Atif Tahir, and James Smith, "Improving Nearest Neighbor Classifier using Tabu Search and Ensemble Distance Metrics", Proceedings of the Sixth International Conference on Data Mining (ICDM'06), pp. 1086-1090, 2006.

[6] Chen, S.M., and Chen, H.H., "Estimating null values in the distributed relational databases environments", Cybernetics and Systems: An International Journal, Vol.31: 851-871, 2000.

[7] Chen, S.M., and Huang, C.M., "Generating weighted fuzzy rules from relational database systems for estimating null values using genetic algorithms", IEEE Transactions on Fuzzy Systems, Vol.11, pp. 495-506, 2003.

[8] L. I. Kuncheva and L. C. Jain, "Designing Classifier Fusion Systems by Genetic Algorithms", IEEE Trans. Evolutionary Computation, pp. 327~336, 2000.

[9] Han, J., and Kamber, M., "Data Mining: Concepts and Techniques", MorganKaufmann Publishers, 2006, 2nd edition.

[10] K. Liu, B. Li, J. Zhang, J. Du , "Ensemble component selection for improving ICA based micro-array data prediction models", Pattern Recognition, vol. 42, pp. 1274 – 1283, 2009.

[11] W. B. Langdon, B. F. Buxton, "Genetic Programming for Combining Classifiers", Proceedings of GECCO'2001, San Francisco, pp. 66-73, 2001.

[12] D. Opitz, "Feature selection for ensembles", Proc. of 16th National Conf. on Artificial Intelligence, AAAI Press, pp. 379–384, 1999.

[13] GJ, McLachlan; K.A. Do, C. Ambroise (2004), "Analyzing microarray gene expression data", Wiley.

14] Shichao Zhang1 and Jilian Zhang, "Missing Value Imputation Based on Data Clustering", PAKDD, LNAI Vol. 4426, pp. 1080–1087, 2007.

[15] Zhang, H., Sun, G., "Feature selection using Tabu Search method", Pattern Recog. Vol. 35, pp.701–711, 2002.

[16] Glover, F. and M. Laguna, "Tabu Search", Kluwer, Norwell, MA, 1997.

[17] A. Tsymbal, M. Pechenizkiy, P. Cunningham, "Diversity in search strategies for ensemble feature selection", Information Fusion, Vol. 6, pp. 83–98, 2005.

[18] Duda, R.O., Hart, P.E., Stork, D.G., "Pattern Classification", Willey-Interscience, 2001.

[19] WebLogExpert, http://www.weblogexpert.com.