IJCTA, 10(03), 2017, pp. 127-137

An Efficient Feature Selection Algorithm for Classifying Patterns on Medical Datasets

J. Jeyacelin* and P.O. Sinciya**

Abstract : Nowadays medical diagnosis got huge attraction in the field of information technology due to large cause of different diseases such as cancer, hepatitis and heart. We can reduce the death rate if we diagnose the disease at early stage. In some cases the manual diagnosis system fails due to inefficient expert and a lot of time taken to diagnose the disease. So in computer aided decision systems, different classification methods are used to help the physician to verify the disease of a patient. The medical dataset contains large number of irrelevant and redundant features. All these features are not necessary to identify whether the patient having cancer or not. Most of the existing feature selection algorithms are not given an optimal solution to the feature set. Thus this paper integrates feature ranking method to the optimized ant colony optimization for selecting relevant features and in each iteration the selected features are evaluated using support vector machine (SVM) classifier algorithm. The performance of this model is evaluated for six bench mark datasets from UCI repository in terms of accuracy, specificity and sensitivity. The classification results of our method out performs than other methods.

1. INTRODUCTION

In recent years, the computational intelligence in medical diagnosis is a new trend for huge medical data applications. Generally, this information is hidden in the large collection of raw data. So that, we are now searching in information, but famishing for knowledge [6]. As a result, data mining is used to extract knowledge from the series of data-mountains by means of data preprocessing. The need of best classifier systems in medical diagnosis is increasing day by day. However, in computer aided decision systems different artificial intelligence techniques for classification are adopted to help the physician in a great deal. The data taken for evaluation from patients and decisions of specialists are the leading factors in diagnosis. The intelligent classification systems can help to minimize possible errors due to inexperienced experts, and also provides more accurate and detailed output in shorter time.

In a pattern classification problem on medical datasets, the aim is to learn the predictive characteristics of decision surface that accurately maps an input feature space to an output space of predefined class labels [7]. A pattern, ordinarily, contains some features based on classifying a target or object. In medical field, a lot of researchers have tried to apply different mechanisms to improve the classification accuracy for the given data. Classification algorithms give better diagnosis accuracy if there is enough information to identify the disease of a patient. ie all the existing features are not necessary to identify the patient is sick or not. So the feature selection techniques plays a main role in classification. In the recent studies, error of global search algorithms such as simulated annealing, ant colony optimization, genetic algorithms, and particle swarm optimizations and also data mining techniques such as Bayesian networks, artificial neural network, fuzzy logic, and decision tree, probabilistic neural networks (PNN), Support Vector machine (SVM) were applied for classification of medical data and meaningful classification results are obtained.

* Professor, Dept of IT Noorul Islam University, Kumaracoil TamilNadu, India. *E-mail: jjeyacelin@gmail.com*

** Assistant Professor, Dept. of CSE Noorul Islam University, Kumaracoil TamilNadu, India. *E-mail: sinciyapo@gmail.com*

In case of data processing, feature selection (FS) is important in order to reduce the dimension of the dataset. It meaningfully reduces the unwanted information, that is, irrelevant, redundant, and noisy features, from the original data set and retains a subset of most relevant features. In practice, pattern classification methods are substantial in a wide range of applications, such as, financial engineering, medical diagnosis, and marketing.

For a given medical data classification, the problem of feature selection can be defined as follows:

Consider the input dataset N which contains n features, find subset of N, named as S contains s features, where $S \subset N$ and $s < n$. The main objective for selecting S features is to maximize the classification accuracy of different learning models. The selection of relevant features is greatly affected in the classification performance because the generalization performance of learning models is greatly dependent on the selected subset of features [8-11]. Furthermore, FS supports for visualizing and understanding the data, reducing storage space requirements, reducing training times and so on [12].

2. FEATURE SELECTION

Feature Selection is important for classifying patterns. In information Science, feature selection is applied to remove noise, includes irrelevant and redundant features. A number of feature selection algorithms are developed by researchers to avoid noise. Some of the commonly used feature selection algorithms are Stepwise selection, Branch and Bound, Principal Component Analysis, Genetic Algorithms. Sequential selection algorithm includes sequential forward selection (SFS) or sequential backward selection (SBS). It starts with either empty or full set of features and in each iteration it adds or remove a single feature until the termination criteria is met. Branch and bound algorithm constructs a tree and a depth first traversal is performed from left to right for checking the value at each level is less than the updated lower bound value. Then prune the corresponding node from the tree.

Unlike sequential search-based FS approaches, global search approaches (or, meta-heuristics) begins by considering full feature space instead of a partial feature space find an optimal set of features. These algorithms are mainly based on the mutual support of individual agents. Genetic Algorithm (GA) is an optimization technique searches the feature space using probabilistic measures. The accuracy of selected features are evaluated using fitness function. For finding the next generation subsets these features are combined with crossover and mutation operations. Nowadays the use of PSO based feature selection is increasing than other evolutionary techniques and it is used to improve performance of feature selection issues, together with their representation, initialization, search techniques and fitness functions. PSO performs well because of straight forward representation and GA performs well in terms of flexible representation. However either of these does not give better solution if the feature space contains thousand or ten thousand features. But ACO based graph approach is more flexible than others. ACO and PSO are the two population based search techniques and both uses filter or wrapper approach for selecting features. A large number of ACO based feature selection works based on either fuzzy or rough set approach. In some feature selection algorithm performs local search during FS by combining two operations, namely, deletion and addition pursue the least significant and most significant features. ACO is an optimized tool, in which a lot of researchers using this algorithm in different applications for selecting relevant features[13-15].

In our proposed approach a hybrid FS with classification algorithm has been proposed that integrates both filter and wrapper methods in a supportive manner. A filter approach involving mutual information computation is used here as a local search to rank features. A wrapper approach involving ACO is used here as global search to find a subset of salient features from the ranked features. For ranking the features Joint mutual Information maximization is used (JMIM).

A. Joint Mutual Information Maximization

JMI (Joint Mutual Information) is a filter feature selection approach which selects most relevant features by considering both relevancy and redundancy. It also considers the class labels when calculating MI (Mutual

Information). But it suffers the problem of overestimation of some features. To avoid this problem we added a maximum and minimum approach with joint mutual information. It selects the relevant features based on the following rule.

Rule 1: If a feature f_a is most relevant to class label C than feature f_b in the selected feature subset S when I ($f_{a\,,}$ S, C) > (f_b , S, C) *i.e*. The feature which is most relevant adds maximum information to that shared between S and C.

$$\text{JMIM} \;=\; \arg\max f_a \,€_{\text{F-S}} (\min_{fb} €_{\text{S}} (\text{I}(\, f_a , f_b, \text{C}))$$

B. Ant Colony Optimization

ACO [1, 2] is a heuristic swarm intelligence optimization algorithm, whose initial member, called Ant Agent, was proposed by Colorni, Dorigo and Maniezzo [4, 3] for finding food in a problem domain. In real world the ants are moving randomly to search for food. The interesting behavior of ant agent is the concurrent search over computational difficulties based on local problem data and the dynamic storage structure contains the details about previously obtained result. When some source of food is found, the ants puts some amount of pheromone whenever they travel as a form of indirect communication. If an isolated ant finds such a path they are likely to follow this path instead of moving at random. Again this ant will puts some amount of pheromone to reinforce pheromone trial of specific path. As the time passes, the pheromone substance starts to evaporates, thus reduces its energy level. If it takes more time to evaporate the pheromone trial means, there are more ants to follow the same path.

A colony of ants moving through different states of a problem gives partial solution to the problem domain. The ants move upon different trials by applying two local decision parameters namely, solution list and validity list. By moving each trial the ant agent constructs the local solution to the problem. During the construction phase, the ant agent analyses the solution to the problem and adjusts the trail value on different components used in its solution. This pheromone information will helps to search for more future ants to give global solution to the problem. An ACO algorithm includes two mechanisms: trail evaporation and daemon actions. Trail evaporation decreases all trail values to avoid indefinite accumulation of trails over certain component. Daemon actions can be used to implement centralized actions which cannot be performed by single ants, such as the invocation of a local optimization procedure, or the update of global information to be used to decide whether to bias the search process from a non-local perspective [5].

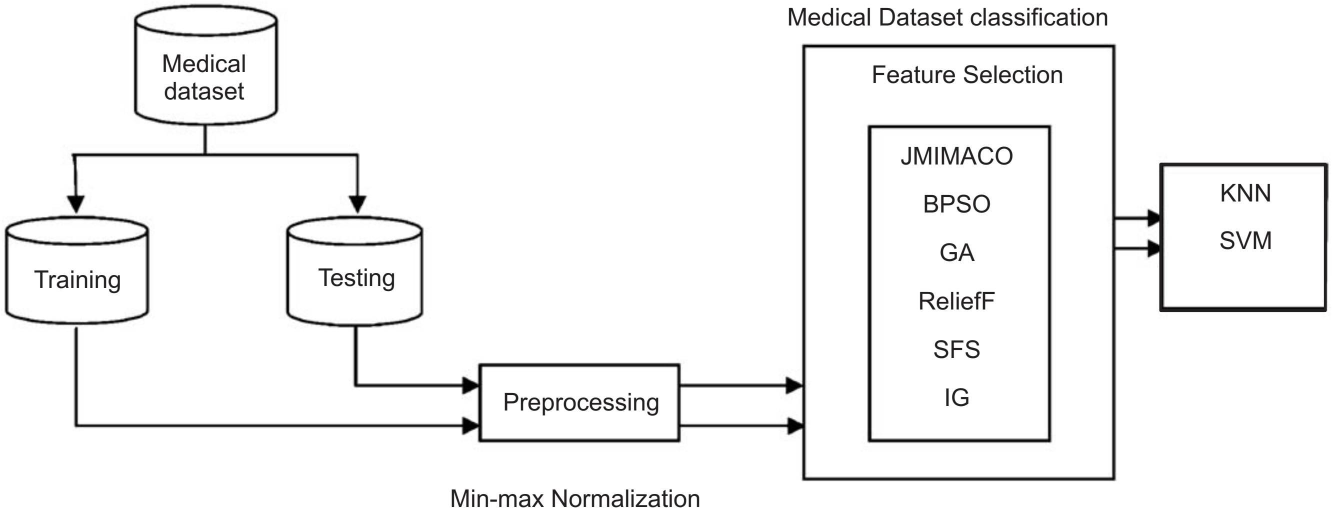

Figure 1: Overview of Proposed approach

3. PROPOSED FEATURE SELECTION FOR MEDICAL DATASETS

To select a set of optimal subset of features for classifying medical datasets an Ant Colony Optimization (ACO) based feature selection mechanism is proposed. In this approach, a group of ants were selected which moves around different trials for finding local optimal subset of features by depositing pheromone. Each ant has a memory list which contains solution list and validity list. The solution list stores the features in the pheromone deposition and the energy value of the ant agent. Initially validity list contains all the features in the dataset and solution list is empty.

Each ant agent have either true movement or false movement to deposit pheromone denoted by true position and false position. Initially the ant moves in a true direction and it takes the values between 0 and n, where n is the features selected from validity list. For negative direction it takes the values 0 for the initial trial. The value in the true position gives the number of features selected from the validity list and false position gives the number of features selected from the solution list.

For initial configuration, we assumes that the ant moves only in true direction, there is no false movement. So the initial value of false position is zero and the number of pheromone trials are set to the number of features in the dataset. Depending on the number selected in true position, the ant agent selects n number of features from the validity list that satisfies maximum JMIM (Joint mutual information maximization. This evaluation method rank the features in the validity based on highest value of JMIM and the highest ranked n features are added to pheromone deposition and are removed from validity list. Then the energy level of ant agent is calculated by finding the classification accuracy of C4.5 decision tree algorithm using K-fold cross validation [6], where k=10 evaluation method. Now, the pheromone starts to evaporate. The ant agent move to the next trial and the number of trials are decremented by 1. To update the pheromone deposition value, the ant agent selects two random numbers in true and false position. For true position, the ant agent selects n number of features from the validity list that satisfies maximum JMIM (Joint mutual information maximization and are added to pheromone deposition. For false position the ant selects a set of p number of features from solution list that satisfies maximum JMIM(Joint mutual information maximization and the features not yet selected are also added to pheromone deposition.. If the energy value of current trial is less than previous value, then it is updated with previous trial value. Again the pheromone starts to evaporate. This process is repeated until the number of trials becomes zero. The features in the solution that has the maximum energy value is considered as the local best (lbest) set of features. This process is repeated for k number of ant agents. The best set of features in lbest is taken as the global best (gbest) set of features.

A. Algorithm for JMMIACO

```
JMMIACO( Number of attributes n, Dataset D)
    Input : Set of features $F_j\{1,2,3\ldots..n\}$
    Output : Selected set of features with gbest
    Initialization :          Initialize count  =  1
                      Set number of ant agents  =  $k$
                                     For count  =  1 to count > $k$
    Initialize                               $i$  =  1;
    Set the number of trials ($t$).
    If ($n$ < 20)
    Then                                     $t$  =  20
    Else                                     $t$  =  50
    Set T and F as the true and false position, where T and F are the number of true and false positions selected by the ant agent.
    If the ant agent in the first trial,
                                         (T,F)  =  (rmd(1, $n$),0)
    Select T features from the storage list that maximizes Joint Mutual Information (JMMI) and add to pheromone trial.
    Select F features from the storage list that maximizes Joint Mutual Information (JMMI) and add to pheromone trial.
    Remove T features from the storage list.
        Energy value $E_i$ (Ant agent) = 10 fold  cross validation ( SVM)
        Solution list(Ant agent) = ($E_i$ (Ant agent, features in pheromone $trial_i$)
    Pheromone Updation // The pheromone trial starts to evaporate when the ant agent moves to the next trial.
        For $i$ = $i + 1$ to $i >= t$
            $n$ = length ( features in storage list)
            $p$ = length (features in solution list)
            (T, F) = (rmd (0, $n$), rmd(0, $p$))
    Select T features from the storage list that maximizes Joint Mutual Information (JIM) and add to pheromone trial.
    Select F features from the storage list that maximizes Joint Mutual Information (JIM) and add to pheromone trial.
    The features not yet selected from solution list are added to pheromone trial.
            Energy value $E_i$(Ant agent) = 10 fold  cross validation ( SVM(features in pheromone trial))
            If($E_i$(Ant agent) < ($E_{i-1}$(Ant agent) in the solution list) then
            Solution list(Ant agent) = ($E_{i-1}$(Ant agent, features in pheromone $trial_i$)
    Delete T features from storage list.
            lbest = features in solution list having maximum $E_{i.}$
    end
    return solutionlist(Ant agent);
            gbest = max(lbest)
    end JMIACOC4.5;
```

4. EVALUATION METHODOLOY

In this study, the performance of proposed method is evaluated on two real world classification problem on medical datasets from UCI Machine Learning Repository [20]. These datasets includes Wisconsin Breast Cancer and Pima Indians Diabetes. The description of dataset is included in table1. Table1 summarizes the number of instances, number of features and number of classes, Missing values, Origin of dataset and Data Source.

Wisconsin Breast Cancer. Dr. William H. Wolberg (1989–1991) collected this dataset from the University of Wisconsin-Madison Hospitals. It contains 699 instances and nine attributes: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. Each instance is used to check whether the patient have benign or malignant growths. In this dataset, total 241 instances are malignant and 458 instances are benign.

Hepatitis. This dataset is obtained from the Carnegie-Mellon University. It contains 155 instances and 19 features (age, sex, steroid, antivirals, fatigue, malaise, anorexia, big liver, liver film, palpable spleen, spiders, ascites, varices, bilirubin, alk phosphate, SGOT, albumin, protime, and histology). This dataset is used to check whether patient is in live or die state.

Table 1
Descrpition of Dataset

Dataset	Number of instances	Number of features	Number of classes	Missing values	Data Origin
Wisconsin Breast Cancer	699	9	2	No	UCI
Hepatitis	155	19	2	Yes	UCI

A. Confusion Matrix

The confusion matrix is commonly named as contingency table. For our experimentation, we consider two class medical datasets namely Breast cancer datasets and Hepatitis. The correctly classified instances are determined by the values in the confusion matrix. High diagonal values in the confusion matrix results with good classification accuracy whereas low values results in poor classification performance.

Table 2
Confusion Matrix with True Positives and True Negatives

Classification Results	*Predicted Class*		
		Class1	*Class2*
Actual Class	Class1	TP	FN
	Class2	FP	TN

B. Evaluation Metrics

For classifying an unknown dataset, depending on the class predicted by the classifier and the true class of the disease dataset, there are four probable cases of results can be detected for the prediction as follows:

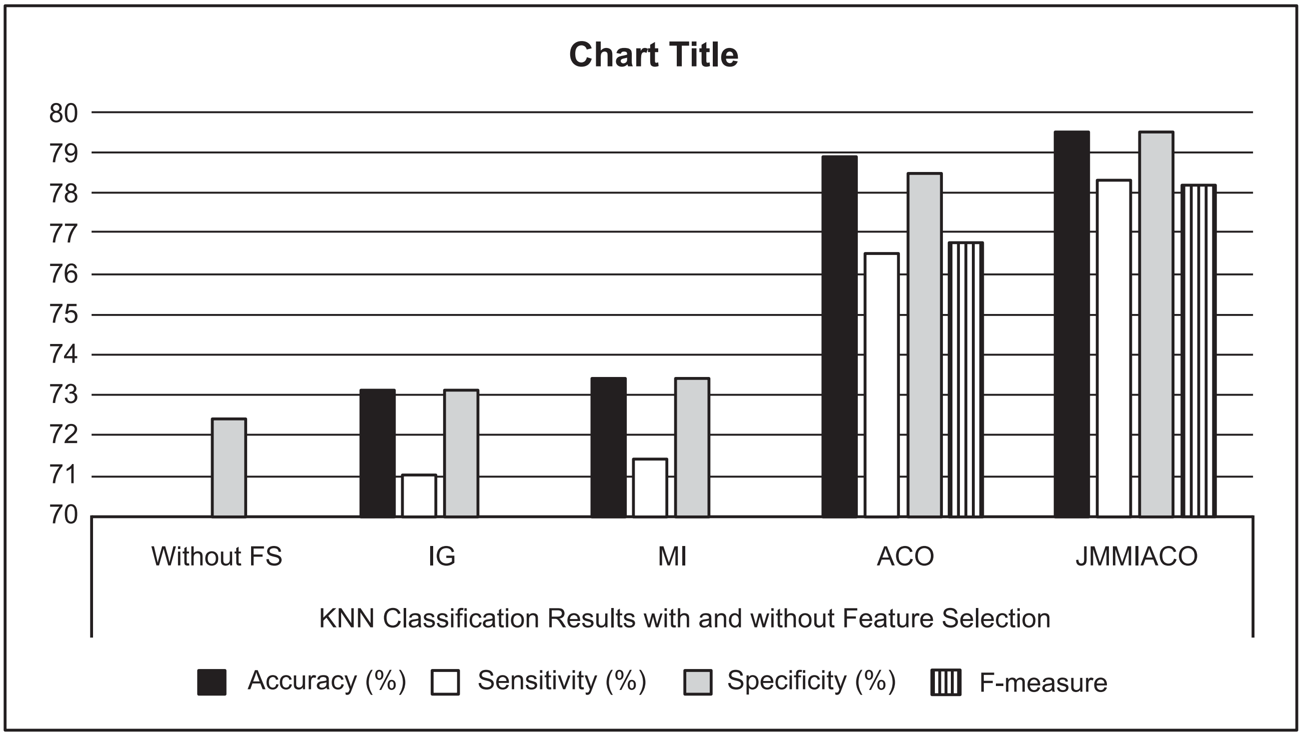

Figure 2: Performance results for breast cancer datasets using KNN

1. **True positive (TP) :** The amount of data items predicted as positive classes belonging to the true class.
2. **False positive (FP) :** The amount of data items predicted as positive classes belonging to the false class.
3. **True negative (TN) :** The amount of data items predicted as negative classes belonging to the false class.
4. **False negative (FN) :** the amount of data items predicted as negative classes belonging to the true class.

These relationships are shortened in the confusion matrix of fig 2.

C. Evaluation Measures

The performance of the proposed ACOC4.5 is evaluated by using six benchmark datasets from UCI and KEEL repositories. Our proposed method is evaluated by considering medical datasets from UCI and KEEL in terms of Sensitivity, Specificity, Accuracy and Error rate.

Let *a*, *b*, *c*, and *d* denotes the total amount of true negatives, false positives, false negatives, and true positives, respectively.

1. Sensitivity (true positive fraction) is the probability that a diagnostic test is positive, given that the person has the disease:

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

2. Specificity (true negative fraction) is the probability that a diagnostic test is negative, given that the person does not have the disease

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

3. Accuracy is the probability that a diagnostic test is correctly performed :

$$\text{Accuracy} = \frac{(\text{TP + TN})}{(\text{TP + TN + FP} + \text{FN})}$$

where TP (true positives) is correctly classified positive cases.

TN (true negative) is correctly classified negative cases.

FP(false positives) is incorrectly classified negative cases.

FN (false negative) is incorrectly classified positive cases.

4. **F-Measure(F) :** It is the average value of sensitivity and specificity.

F = 2. sensitivity*specificity / Sensitivity + specificity

5. EXPERIMENTAL RESULTS

We have evaluated our proposed feature selection algorithm with filter methods and wrapper methods in terms of both performance and training time. Two medical datasets from UCI repository is taken for evaluation and two traditional classification algorithms are used for evaluating classification performance for the selected set of features. The data set for classification is divided in to training and testing dataset. For classification a model is created using training dataset and the test data is used to evaluate performance. Here we explore a 10-fold cross validation for evaluating classification accuracy. For that dataset is divided into 10 folds. From that nine sets of data are used for training and one sets of data are used for testing.

Table 3
Classification results with Breast Cancer dataset

Evaluation metrics	*KNN classification results with and without Feature selection*				
	Without FS	*IG*	*MI*	*ACO*	*JMMIACO*
Accuracy (%)	72.37	73.07	73.42	78.9	79.5
Sensitivity (%)	69.9	71.0	71.4	76.5	78.3
Specificity (%)	72.4	73.1	73.4	78.5	79.5
F-measure	69.7	68.3	69.1	76.8	78.2
Selected features	All	1, 2, 3, 5, 6, 7, 8	2, 3, 6, 7, 8	2, 3, 8	2, 3, 7

Table 4
Classification results with Breast Cancer dataset using SVM

Evaluation metrics	*SVM classification results with and without Feature selection*				
	Without FS	*IG*	*MI*	*ACO*	*JMMIACO*
Accuracy (%)	72.57	73.07	75.46	82.3	82.15
Sensitivity (%)	70.83	71.0	76.4	86.51	87.3
Specificity (%)	71.34	73.1	74.42	81.7	83.5
F-measure	72.7	68.3	76.1	83.6	81.2
Selected features	All	1, 2, 3, 5, 6, 7, 8	2, 3, 6, 7, 8	2, 3, 8, 7	2, 3, 6, 7

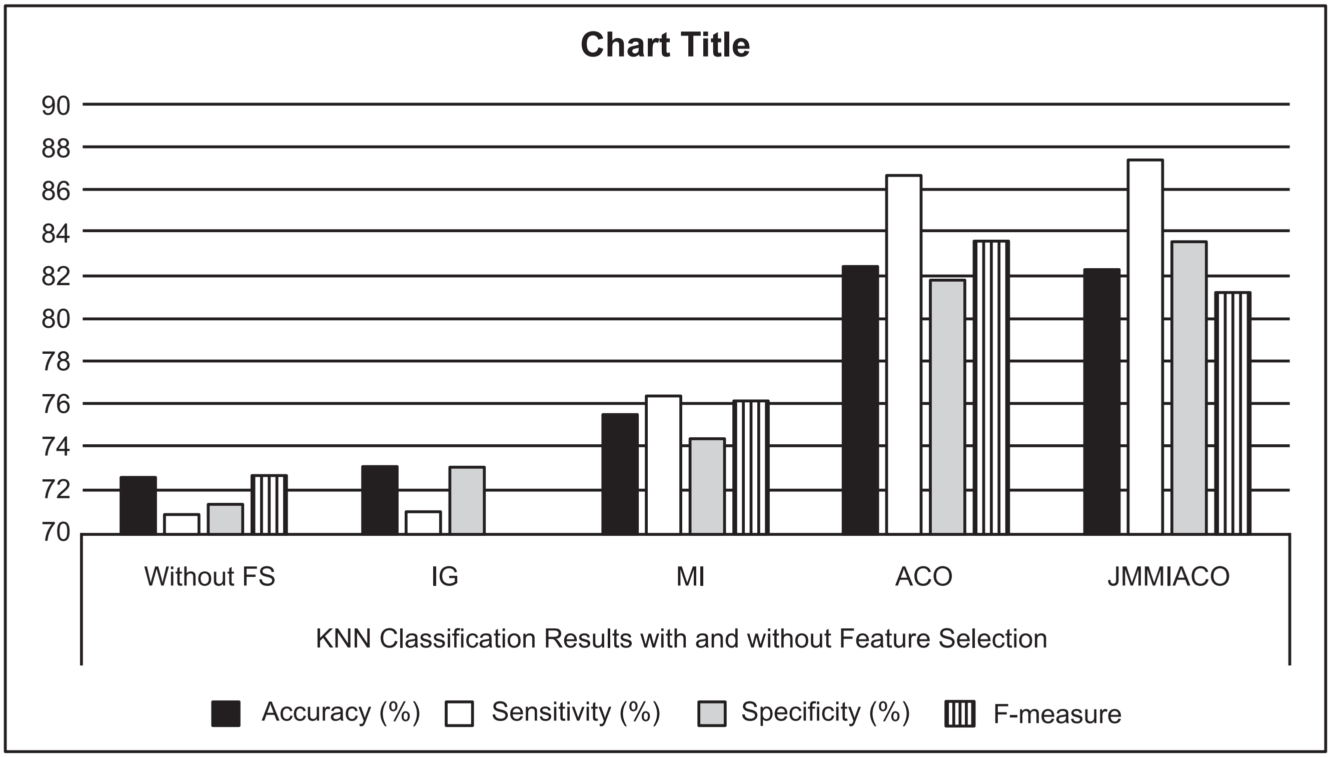

Figure 3: Performance results for breast cancer datasets using SVM

Table 5
KNN Classification results with Hepatitis dataset

Evaluation metrics	*KNN classification results with and without Feature selection*				
	Without FS	*IG*	*MI*	*ACO*	*JMMIACO*
Accuracy (%)	70	71.1	78	82.3	86.04
Sensitivity (%)	71	78	79	82.1	85.91
Specificity (%)	69	79	78	83.7	86.04
F-measure	71	78	82.3	81.2	91.27
Selected features	ALL	1, 2, 6, 7, 8	1, 2, 6, 8	1, 2, 6, 8	2, 6, 8

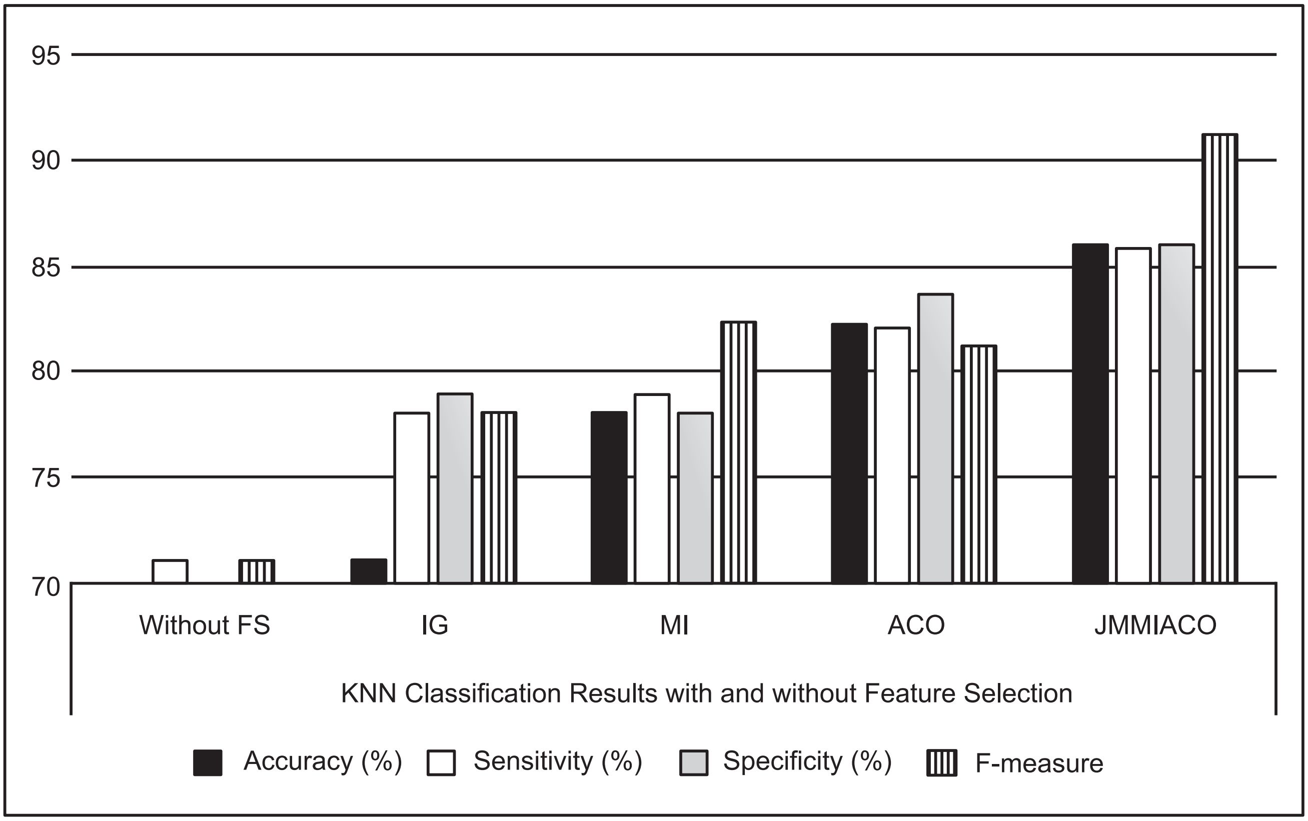

Figure 4: Performance results for Hepatitis datasets using KNN

Table 6
SVM Classification results with Hepatitis dataset

Evaluation metrics	*SVM classification results with and without Feature selection*				
	Without FS	*IG*	*MI*	*ACO*	*JMMIACO*
Accuracy (%)	76	78.1	81.3	84.3	88.04
Sensitivity (%)	76.3	78.7	82.4	82.7	89.91
Specificity (%)	73.5	79.9	86.5	85.9	91.04
F-measure	74.1	78.4	85.9	87.2	92.47
Selected features	ALL	1, 2, 5, 6, 7, 8	1, 2, 6,	1, 2, 8	2, 6, 8

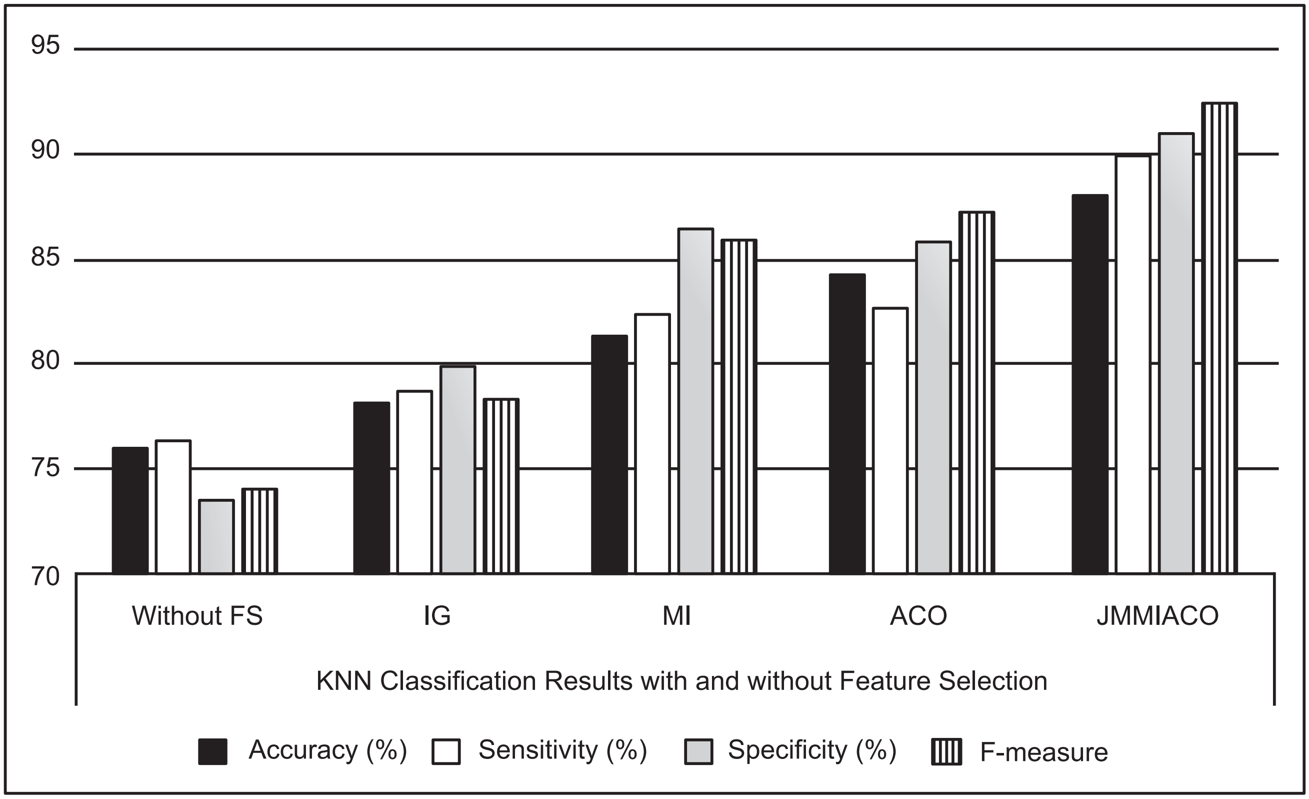

Figure 5: Performance results for Hepatitis datasets using KNN

6. CONCLUSION

In this paper presents a hybrid feature selection methods based on filter and wrapper approaches: :Joint Mutual Information Maximisation (JMIM) and Ant colony optimization(ACO). This method overcome the problem of selecting irrelevant and redundant features in some situations because of using joint mutual information maximization and the performance of the classification problem is improved because of ACO. This hybrid approach avoids both problems in filter and wrapper approaches. The proposed algorithm selects the features in an efficient way and gives better classification accuracy in terms of sensitivity, specificity and f-measure. We have used two intelligent classification algorithms for evaluating the classification performance on the selected features of medical datasets namely an instance based KNN and SVM. In our experimental results shows that SVM gives a better classification results in terms of accuracy, sensitivity, specificity and f-measure. But in terms of training time KNN gives better results. For training the datasets KNN takes very less time as compared to SVM because KNN follows instance based strategy for classifying the datasets.

7. REFERENCES

1. M. Dorigo, Ant colony optimization web page, http://iridia.ulb.ac.be/mdorigo/ACO
2. M. Dorigo, G. Di Caro, L.M. Gambardella (1999) Ant Algorithms for Discrete Optimization. Artificial Life, 5(2):137-172
3. M. Dorigo, V. Maniezzo, A. Colorni (1991) the ant system: an autocatalytic optimizing process, Technical Report TR91-016, Politecnico di Milano.
4. A. Colorni, M. Dorigo, V. Maniezzo (1991) Distributed optimization by ant colonies, In Proceedings of ECAL'91European Conference on Artificial Life, Elsevier Publishing, Amsterdam, The Netherlands, pp 134-142.
5. M. Dorigo, T. Stützle (2002) the ant colony optimization metaheuristic: Algorithms, applications and advances. In F. Glover, G. Kochenberger (eds) Handbook of Metaheuristics, Kluwer Academic Publishers, Norwell, MA, pp. 251-285.

6. A. L. Blum and P. Langley, “Selection of relevant features and examples in machine learning,” Artificial Intelligence, vol.97, no.1-2, pp.245–271, 1997.

7. S. Das, “Filters, wrappers and a boosting-based hybrid for feature selection,” in Proceedings of the 18th International Conference on Machine Learning (ICML ’01) ,pp.74–81,Morgan Kaufmann Publishers, San Francisco, Calif, USA, 2001

8. Saeys, Y.,Inza,I., & Larranaga,P.(2007). A review of feature selection techniques in bioinformatics. *Bioinformatics, 23, 2507*–2517.

9. Brown, G., Pocock, A., Zhao, M., & Lujan, M.(2012). Conditional likelihood maximisation: aunifying framework for information theoretic feature selection. *Journal of Machine Learning Research*, *13*, 27–66.

10. Cheng, H., Qin, Z., Feng, C.,Wang,Y., & Li, F.(2011). Conditional mutual information-based feature selection analysing for synergy and redundancy. *Electronics and Telecommunications Research Institute, 33*, 210–218.

11. Karegowda, A.G., Jayaram, M. A., & Manjunath, A.S. (2010 Feature subset selection problem using wrapper approach in supervised learning. *International Journal of Computer Applications,* 1,13–17.

12. Guyon, I., Gunn, S., Nikravesh, M., & Zadeh, L.A. (2006). *Feature extraction foundations and applications*. NewYork/Berlin, Heidelberg: Springer Studies in fuzziness and soft computing.

13. P. Jafari and F. Azuaje, “An assessment of recently published gene expression data analyses: reporting experimental design and statistical factors,” BMC Medical Informatics and Decision Making ,vol.6,no.1, article 27, 2006.

14. M. A. Hall, “Correlation-based feature selection for machine learning,” Tech. Rep., 1998.

15. P. Yang, B. B. Zhou, Z. Zhang, and A.Y. Zomaya,“Amulti-filter enhanced genetic ensemble system for gene selection and sample classification of microarray data,” BMC Bioinformatics ,vol.11, supplement 1, article S5, 2010