Classifying Different Feature Selection Algorithms Based on the Search Strategies

Mohammad-Reza Feizi-Derakhshi, and Manizheh Ghaemi

***Abstract*—** Data mining is an inevitable step in knowledge discovery and it helps discovering hidden and useful patterns among data. These days, the number of stored attributes for each entity in databases is rapidly growing. But not only all of these attributes (features) are not useful for data mining, but also some irrelevant attributes make the result of data mining more complex and less understandable. As a result, in databases with so many features to handle, more valuable and relevant features are selected and redundant and irrelevant features are ignored by feature selection algorithms. Feature selection reduces the dimension of databases and makes the results more useful.

Feature subset selection algorithms can be divided into two categories: filter approach and the wrapper approach. From the other point of view, we can categorize feature selection algorithms-regardless of their correspondence with filter or wrapper approach in 4 groups: complete search, heuristic search, meta-heuristic methods and methods that use artificial neural networks. The aim of this article is to classify the recent presented methods into the mentioned groups and we will review some of them. Also, we have attempted to compare the groups with each other.

***Keywords*—** Data mining, feature selection, dimension reduction

I. Introduction

DATA mining is an inevitable step in knowledge discovery and the knowledge obtained as the result of data mining is used in many trends; like business and medical use. Recently, there has been an increase in the number of collected and stored features in databases but most of the times, some of the features are irrelevant or redundant. These features not only have no use in the process of knowledge discovery, but also they increase the complexity and incomprehensibility of the results. So, dimension reduction with the help of feature selection (FS) is a useful step before data mining.

Feature selection is concerned in databases with many features; because in such databases when there are n features, time complexity to evaluate all the subsets of features is exponential ($O(2^n)$), which is practically impossible. The basis of database dimension reduction is to keep useful features for latter learning tasks alongside the ignoring of the most irrelevant and less important ones [1]. In fact feature selection techniques helps to ignore irrelevant features and as a result, learning process can be done more efficiently. It is also proved that feature selection increases the classification accuracy [1] and produces more intelligible results. Also, it is reported that methods like feature selection and more specifically Feature Weighting (FW) methods can improve the classification accuracy of machine learning algorithms like KNN classifier [12].

FS can be considered as a special case of feature weighting when the weights are limited to just '0' and '1' [1]. These weights illustrate the importance of the features for classification. In this case, a weight '0' illustrates the omission of the corresponding feature while '1' as the weight, guarantees the feature's admission. Tahir and et al. [11] proposed an algorithm which is the combination of feature selection and feature weighting using Tabu search method. They tested their method on some data sets and reported promising results in comparison with other methods.

There are many criterions for evaluating the selected feature subset, but classification accuracy (CA) on new instances is the most common performance evaluation criterion. In fact, we will reduce the dimension of a database if after dimension reduction, classification accuracy increases or at least it remains the same.

II. Literature Review

There has been much effort for solving the feature selection problem up to now and many researchers have attempted to speed up the process of selecting informative features in databases.

Filters are the earliest methods in FS literature based on the machine learning algorithms [1]. All the filters make use of heuristic techniques based on the General Characteristics of data such as information gain and distance, instead of learning algorithms. Fig. 1 illustrates the overall flowchart of the filters. As it is obvious from the Fig. 1, machine learning algorithms are not involved in the filtering of irrelevant features.

Another approach in feature selection is wrapper methods. Wrappers, in contrast to filters, use learning algorithms to investigate the worthy of features [1]. The principal idea behind this approach is that, the induction algorithm that eventually will use the selected features, can predict the accuracy of the selected features better than any other

Mohammad-Reza Feizi-Derakhshi is with Computer Engineering Department, University of Tabriz, Tabriz, Iran (corresponding author's phone: +98-9143141876; e-mail: mfeizi@tabrizu.ac.ir).

Manizheh Ghaemi is with Computer Science Department, University of Tabriz, Tabriz, Iran (e-mail: m_ghaemi89@ms.tabrizu.ac.ir).

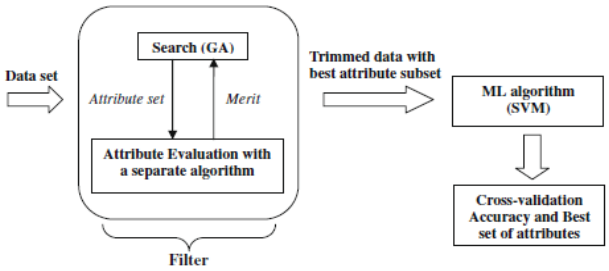

Fig. 1 flowchart of filters [6]

methods. Generally, wrappers produce better results than filters [1]; because they consider the relationship between the learning algorithm and the training data. From the other side, wrappers are slower than filters; because for every selected feature subset, the learning algorithm must be repeatedly executed. The flowchart for wrapper method is shown as Fig. 2.

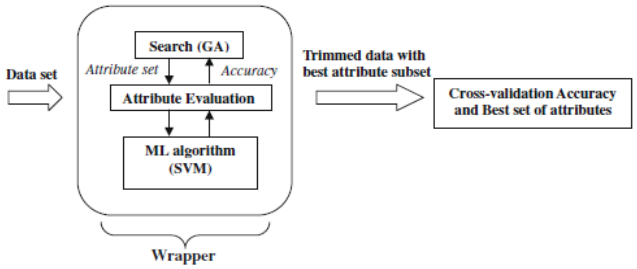

Fig. 2 Flowchart of wrappers [6]

Feature selection algorithms, regardless of what category they correspond (filters or wrappers), can be classified into 4 groups: complete search, heuristic search, meta-heuristic methods and methods that use artificial neural networks. In this section we will briefly review some of the earlier methods for solving the feature selection problem. At first, we will briefly discuss the complete search methods; then we will have little review of heuristic and meta-heuristic techniques. Finally we will end up this section with a brief review of the use of artificial neural networks in feature selection.

A. *Complete search methods*

Almuallim and Dieterich presented the FOCUS method [1]; which was firstly designed for the binary spaces. FOCUS completely searches the search space up to reaching to the smallest set of features that divides the training data into pure classes. Liu and Setiono presented LVF, which is like FOCUS but it can handle noisy domain [1].

Complete search methods traverse the solution space completely up to reaching to the optimum feature subset. With n features to consider, it is obvious that there are 2^n different combinations of features. So, evaluating all of the combinations for big ns is impossible. As a result, complete search methods are seldom used because of their time and storage complexity in databases with many features (more than 50 features) [11].

B. *Heuristic search methods*

Greedy hill climbing algorithm, branch and bound method, beam search and best first algorithm are the heuristic methods of feature selection problem. Greedy hill climbing algorithm considers all local changes in order to select the relevant features [1]. In this algorithm adding a feature to the selected features and deleting one of them can be considered as local changes. SFS (Sequential Forward Selection) and SBS (Sequential Backward Selection) are two kinds of hill climbing. While SFS starts with empty set of selected features and each step of the algorithm adds one of the informative features to the set, SBS starts will the full set of features and in each step, one of the redundant or irrelevant features is omitted. Another method is bi-directional search; which considers both adding and deleting the features simultaneously. Both SFS and SBS algorithms have the "nesting effect" problem, which means that while a change is considered positive, there is no chance of re-evaluating that feature. SFFS (Sequential Forward Floating Selection) and SBFS (Sequential Forward Floating Selection) are two methods that were presented to overcome the "nesting effect" of SFS and SBS algorithms [1].

Best first search is another method based on artificial intelligence methods, which allows backtracking in the search space [1]. This algorithm, like greedy hill climbing algorithm, makes use of local changes in the search space. But in contrast to it when the path for reaching the optimum solution is not hopeful, it is possible to backtrack the search space. Fig. 3 shows the overall flowchart of this algorithm.

Cadenas and et al. [13] proposed a method which is the combination of filter and wrapper approach. They used sequential search procedure in order to improve the classification accuracy and reported promising results in both low quality data and crisp data.

Heuristic algorithms perform better than complete search methods, but recently meta-heuristic algorithms like Genetic Algorithm (GA), Particle Swarm Intelligence Optimization (PSO) and Ant Colony Optimization (ACO) show more desirable results while comparing time complexities.

1. Begin with the OPEN list containing the start state, the CLOSED list empty, and BEST←start state.
2. Let s = arg max e(x) (get the state from OPEN with the highest evaluation).
3. Remove s from OPEN and add to CLOSED.
4. If e(s) ≥ e(BEST), then BEST ← s.
5. For each child t of s that is not in the OPEN or CLOSED list, evaluate and add to OPEN.
6. If BEST changed in the last set of expansions, goto 2.
7. Return BEST.

Fig. 3 Pseudo-code of best first algorithm [1]

C. *Meta-heuristic search algorithms*

Due to the random nature of meta-heuristic search methods, the application of genetic algorithms and ant colony optimization in feature selection domain has been studied. Researchers have reported Promising results of applying meta-

heuristic methods. In the following we will discuss some of them.

Hamdani et al. have proposed a new algorithm based on genetic algorithms with bi-coded chromosome representation and new evaluation function [4]. They used a Hierarchical algorithm with homogeneous and heterogeneous population in order to minimize the computational cost and also speed up the convergence speed. In their method, heterogeneous GA performs a global search among the solutions with different sizes and then a number of best solutions are sent to homogeneous GAs in order to locally optimize the solutions. Due to the parallel nature of their proposed method, they reported a good performance in comparison with heuristic algorithms and simple GA.

Zhu et al. proposed a new algorithm WFSSA (Wrapper–Filter feature Selection Algorithm), which is a combination of genetic algorithm and local search method [5]. It is based on both filter and wrapper approach. In this algorithm first, the GA population is generated randomly so that an individual represents a set of selected features. Then, local search is applied to all of the individuals of the population. They applied local search operators on the solutions of genetic algorithm in order to improve the classification accuracy and speed up the searching process. In fact, local search operators improve the GA population by adding the informative features and deleting the redundant features. This process is repeated until the stopping conditions are satisfied. They used the binary representation for chromosomes where each chromosome has the length equal to the number of all features. This way a '1' in a gene represents the selection of the corresponding feature and '0' shows a feature's omission from the selected subset. Fig. 4 shows a chromosome's representation in this algorithm. The experiments showed a good performance of WFSSA in comparison with simple GA and also ReliefF algorithm.

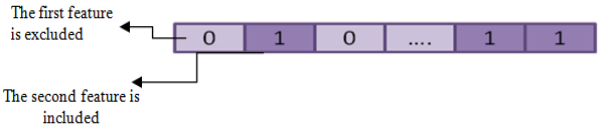

Fig. 4 Representation of a chromosome [5]

Tan et al. combined GA and SVM (Support Vector Machine) based on wrapper approach [6]. In their proposed algorithm, GA searches for the best feature subset by applying the evolutionary operators and then SVM classifies the chromosomes. Binary representation of the chromosomes is used and single point crossover and flipping bit mutation are GA operators in this algorithm. Classification accuracy of SVM is used as the fitness value of the chromosomes. They evaluated their method on different UCI repository and they reported that combining these two algorithms produces much better results of GA or SVM alone.

Gheyas et al. proposed a hybrid algorithm which is named SAGA [7]. They combined SA (Simulated Annealing) and GA to use the advantage of both of them. GA helps to escape from local optimum of SA with the crossover operator. Greedy algorithms are used to evaluate the local changes. SAGA has the three main stages:

- SA performs a local search. While the temperature is high, SA accepts every new solution. When the temperature becomes near to zero, just improvements are accepted. SA takes the 50% of overall time of the proposed algorithm.
- Next, GA with 100 individuals is executed. These 100 individuals are selected among the best answers of SA. This stages use the 30% of the time.
- SAGA in this stage uses the greedy search to form the final solution. The local search is performed on the k best solutions of both SA and GA.

Generalized regression neural network (GRNN) learning algorithm is employed as the fitness function, but each feature candidate subset is normalized before evaluation. They compared their method with ACO, GA, PSO, SA, SBS, SFBS, SFFS and SFS algorithms and they reported better results than all of them.

Nemati et al. proposed a new hybrid algorithm of GA and ACO in order to use the advantages of both algorithms [8]. In this algorithm ACO performs a local search, while GA is used to perform a global search. The main stage in ACO is to model the problem as the finding a minimum cost path problem in a graph. Nodes of this graph represent the features and the edges determine the next selected feature. Searching for the optimum feature subset can be considered as a kind of ant traversal through the graph where a minimum number of nodes are visited. Fig. 5 shows an example of graph representation for feature selection problem.

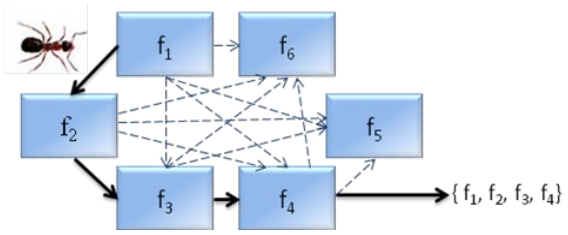

Fig. 5 problem representation of ACO for FS [8]

The ant is currently in node f1 and has to choose the next feature. Upon arrival at f4, the current subset {f1, f2, f3, f4} is determined to satisfy the traversal-stopping criterion. In this algorithm GA and ACO are combined in order to complement each other. Both GA and ACO start searching for the solution in parallel and then the results of them are gathered to be evaluated and then the best solution is selected. If the algorithm finds the near-optimum solution or if it has ran for the specified iterations, the algorithm stops executing and the best solution can be selected. In this algorithm, the final solution can be the result of executing either GA or ACO.

D. ANN aided methods

In this section, we will review some of methods that combine meta-heuristic algorithms with artificial neural

network (ANN). In these algorithms, the use of ANN as a learning algorithm is evaluated.

Classification is the process of classifying any given input feature vector into pre-defined set of classes. The choice of features used for classification has an impact on the accuracy of the classification function, the time required for classification, training data set requirements, and implementation costs associated with the classification.

Karthik et al. combined ACO and ANN for solving the feature selection problem [9]. In this algorithm, ACO searches for near-optimum solution and ANN is used as a classifying function. First, a set of ants is initialized based on the number of features. Each initialized ant will select a subset of n features from the original set of N features. Once all ants have completed constructing their subsets, a global updating rule is applied to the solution set which produces the least classification error. Similarly, a local pheromone updating rule is applied to the rest of the ants.

ANNs are used to evaluate the goodness of the subsets developed by ants as solution in each iteration. The networks are trained using Levenberg–Marquard's back propagation algorithm and the number of incorrect classification is considered as the selected subset error. In this hybrid algorithm, ANN acts like a guide for ACO.

ElAlami proposed an algorithm based on GA, which optimizes the output nodes of ANN [2]. In this method ANN is used to give a weight to each of the features. This algorithm starts with the learning on input features. Learning phase continues up to reaching to an acceptable level of error rate. In the network, each input unit corresponds typically to a single feature and each output unit corresponds to a class value. After training the ANN, the weights between input-hidden and hidden-output layers are extracted. Therefore, each output node of ANN can be represented as a general function of input features and extracted weights. Sigmoid function is used as the activation function in hidden and output nodes.

For each output node, GA is used to find the best values for input features and also it is used to maximize the output function. At last, the obtained features represent the relevant features for each class value. Because the output function is not linear, so the use of GA is suggested. Each iteration in this algorithm produces one of the output nodes. In fact, GA searches for the feature subset that maximizes the output function. Generally these features are the ones with the maximum weights in ANN layers.

Monirul Kabir et al. proposed a new hybrid algorithm that combines GA with local search methods (HGAFS1) [10]. The important aspect of this method is selecting the feature subset with a limited size. This method is a wrapper based method that uses both GA and ANN. Search process based on correlation acts as the local search and GA performs the global search. ANN is used as the evaluation function. The proposed algorithm in their article uses GA as a global search method and it also restricts the number of '1' in each chromosome to find the optimum feature subset with less features.

[1] Hybrid Genetic Algorithm Feature Selection

III. Comparison

Although complete search methods for feature selection problem can find the optimum feature subset, but they are not applicable; because their time complexity grows exponentially when the number of features to consider increases. So, complete search methods are seldom used in databases with many features. Heuristic methods restrict the search space by some methods like branch and bound algorithm and they can reduce the time complexity of complete methods. But, meta-heuristic methods, because of their random nature, can perform faster than heuristic and complete search methods. These methods like GA and ACO don't guarantee the optimality of the solution but they are acceptable due to their time complexity. ANNs can aid the feature selection problem as they can learn and improve the performance.

IV. Conclusion

Because the feature selection problem is known to be NP-Hard, so many different heuristic and meta-heuristic algorithms have been proposed by many researchers. Although complete search methods can find the optimum solution, but they need exponential time to find the answer. As a result, recently the use of meta-heuristic algorithms has gained much interest in feature selection trend. Meta-heuristic algorithms like GA attempt to find near-optimum solutions based on evolutionary approach. Of course they don't guarantee the optimality of the answer, but their time complexity is acceptable.

There are many criterions for evaluating the selected feature subset, but classification accuracy on new instances is the most common performance evaluation criterion that is used. In this article we have attempted to classify different feature selection algorithms into four groups: complete search, heuristic search, meta-heuristic methods and methods that use artificial neural network.

References

[1] Mark A. Hall, "Correlation-based Feature Selection for Machine Learning," PHD thesis, Hamilton, NewZealand, 1999.

[2] ElAlami, M.E. "A filter model for feature subset selection based on genetic algorithm," Elsevier, Knowledge-Based Systems journal, pp. 356-362, 2009.

[3] Yonghong Peng, Zhiqing Wu, Jianmin Jiang, "A novel feature selection approach for biomedical data classification," Elsevier, Journal of Biomedical Informatics, PP. 15-23, 2010.

[4] Tarek M. Hamdani, Jin-Myung Won, Adel M. Alimi, Fakhri Karray, "Hierarchical genetic algorithm with new evaluation function and bi-coded representation for the selection of features considering their confidence rate", Elsevier, Applied Soft Computing, vol. 11, PP. 2501-2509, 2011.

[5] Zexuan Zhu, Yew-Soon Ong, and Manoranjan Dash, "Wrapper–filter feature selection algorithm using a memetic framework," IEEE transactions on systems, man and cybernetics, vol. 37, No. 1 , p.p 70-76, february 2007.

[6] K.C. Tan, E.J. Teoh, Q. Yu, K.C. Goh, "A hybrid evolutionary algorithm for attribute selection in data mining," Expert Systems with Applications, Elsevier, pp. 8616-8630, 2009.

[7] Iffat A.Gheyas, LeslieS.Smith, "Feature subset selection in large dimensionality domains," Elsevier, Pattern Recognition, vol. 43, PP. 5-13, 2010.

[8] Shahla Nemati, Mohammad Ehsan Basiri, Nasser Ghasem-Aghaee, Mehdi Hosseinzadeh Aghdam, "A novel ACO–GA hybrid algorithm for

feature selection in protein function prediction," Elsevier, Expert Systems with Applications, vol 36, pp.12086-12094, 2009. http://dx.doi.org/10.1016/j.eswa.2009.04.023

[9] Rahul Karthik Sivagaminathan, Sreeram Ramakrishnan, "A hybrid approach for feature subset selection using neural networks and ant colony optimization," Elsevier, Expert Systems with Applications, vol 33, pp. 49-60, 2007. http://dx.doi.org/10.1016/j.eswa.2006.04.010

[10] Md. Monirul Kabir, Md. Shahjahan, Kazuyuki Murase, "A new local search based hybrid genetic algorithm for feature selection," Elsevier, Neurocomputing, vol 74, pp. 2914-2928, 2011. http://dx.doi.org/10.1016/j.neucom.2011.03.034

[11] Muhammad Atif Tahir, Ahmed Bouridane, Fatih Kurugollu, "Simultaneous feature selection and feature weighting using Hybrid Tabu Search/K-nearest neighbor classifier", Pattern Recognition Letters, vol. 28, pp 438–446, 2007. http://dx.doi.org/10.1016/j.patrec.2006.08.016

[12] Isaac Triguero, Joaqun Derrac, Salvador Garcia, Francisco Herrera, "Integrating a differential evolution feature weighting scheme into prototype generation". Elsevier, Neurocomputing, vol. 97, pp. 332-343, 2012. http://dx.doi.org/10.1016/j.neucom.2012.06.009

[13] Cadenas Jose M., M. Carmen Carrido, Raquel Martinez, "Feature subset selection Filter-wrapper based on low quality data", Elsevier, Expert Systems with Applications, vol. 40, pp. 6241-6252, 2013. http://dx.doi.org/10.1016/j.eswa.2013.05.051