Expert Systems with Applications 38 (2011) 8144–8150

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Hybrid feature selection by combining filters and wrappers

Hui-Huang Hsu, Cheng-Wei Hsieh *, Ming-Da Lu

Department of Computer Science and Information Engineering, Tamkang University, 151 Ying-chuan Road, Tamsui, Taipei County 25137, Taiwan, ROC

ARTICLE INFO

Keywords:
Feature selection
Filters
Wrappers
Support vector machine
Disordered protein
Microarray

ABSTRACT

Feature selection aims at finding the most relevant features of a problem domain. It is very helpful in improving computational speed and prediction accuracy. However, identification of useful features from hundreds or even thousands of related features is a nontrivial task. In this paper, we introduce a hybrid feature selection method which combines two feature selection methods – the filters and the wrappers. Candidate features are first selected from the original feature set via computationally-efficient filters. The candidate feature set is further refined by more accurate wrappers. This hybrid mechanism takes advantage of both the filters and the wrappers. The mechanism is examined by two bioinformatics problems, namely, protein disordered region prediction and gene selection in microarray cancer data. Experimental results show that equal or better prediction accuracy can be achieved with a smaller feature set. These feature subsets can be obtained in a reasonable time period.

1. Introduction

The development of feature selection has two major directions. One is the filters (Liu et al., 2005) and the other is the wrappers (Kohavi & John, 1997). The filters work fast using a simple measurement, but its result is not always satisfactory. On the other hand, the wrappers guarantee good results through examining learning results, but it is very slow when applied to wide feature sets which contain hundreds or even thousands of features.

Through the filters are very efficient in selecting features, they are unstable when performing on wide feature sets. This research tries to incorporate the wrappers to deal with this problem. It is not a pure wrapper procedure, but rather a hybrid feature selection model which utilizes both filter and wrapper methods. In our method, two feature sets are first filtered out by *F*-score and information gain (Quinlan, 1979), respectively. The feature sets are then combined and further tuned by a wrapper procedure. We take advantages of both the filter and the wrapper. It is not as fast as a pure filter, but it can achieve a better result than a filter does. Most importantly, the computational time and complexity can be reduced in comparison to a pure wrapper. The hybrid mechanism is more feasible in real bioinformatics applications which usually involve a large amount of related features.

In the experiments, we applied the proposed hybrid feature selection mechanism to the problems of disordered protein prediction (Linding et al., 2003) and gene selection of microarray cancer data (Guyon, Weston, Barnhill, & Vapnik, 2002). The definition of disordered regions of a protein is the segments of a protein sequence that do not have a fixed conformation. According to the central dogma of structural biology, the function of a protein is determined by its tertiary structure. In the past, these disordered regions were thought to be useless or even harmful to structural stability. However, more and more recent studies showed that these regions have special functions, like signal controlling or regulation (Ishida & Kinoshita, 2007). To computationally predict this kind of regions requires a lot of information (features from protein primary structure). As for the gene selection of microarray cancer data, dealing with its thousands of features is essential. To identify the main disease genes from thousands of other regular genes in the microarray data, effective feature selection is always very helpful.

In the remainder of the paper, related work is first discussed in Section 2. The proposed hybrid feature selection mechanism is then delineated in Section 3. The learning model and datasets are introduced in Section 4. The experimental results are presented in Section 5. Finally, a brief conclusion is drawn in Section 6.

2. Related work

Feature selection methods have been applied to classification problems in order to select a reduced feature set that makes the classifier more accurate and faster. Some specific problems are always processed with a great number of features. For instance, microarrays, transaction logs, and web data are all very wide datasets with a huge amount of features. Here we first review papers about the filters and the wrappers.

Huang, Cai, and Xu (2006) used a filter approach for feature selection based on mutual information. In their point of view, there

* Corresponding author. Tel.: +886 920 782921.
E-mail address: 892190108@s92.tku.edu.tw (C.-W. Hsieh).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.12.156

are two types of input features perceived as being unnecessary. They are features completely irrelevant to the output classes and features redundant given other input features. By using the mutual information test on features vs. classes and features vs. features, feature selection can be done. This is from the concept of information theorem which analyzes the relationship between features and classes to remove the most related (redundant) features or the most irrelevant to the class. In their research, a greedy feature selection algorithm was proposed.

Another filter work was done by Deisy, Subbulakshmi, Baskar, and Ramaraj (2007). They used the analysis of symmetrical uncertainty with information gain. By calculating the difference between the entropy of the whole class and the features, features with less information can easily be identified. In addition, some other feature selection methods are based on the features' discrimination ability. For example, Chen and Lin (2003) used *F*-score to perform feature selection. In their work, the support vector machine (SVM) was used as the feature set performance measurement. The *F*-score analyzes the decimation ability of each feature. Owing to the SVM also tries to find a separation hyper-plane to divide different classes' data apart, the *F*-score may helpful for SVM to remove some features of low decimation ability.

Backstrom and Caruana (2006) presented an internal wrapper feature selection method for cascade correlation. The internal wrapper feature selection method selects features while hidden units are being added to the growing cascade correlation network architecture. Liu, Yin, Gao, and Tan (2008) developed a wrapper-based optimized SVM model for demand forecasting. At first, wrappers based on the genetic algorithm are employed to analyze the sales data of a product. Then the selection result is applied to build a SVM regression model.

Next, two bioinformatics problems were tested in this research. One of the major inventions in biology is the microarray technique. Disease classification based on gene expression data from microarray is very important. This topic is also related to feature selection. Classification without feature selection would certainly affect both the processing time and the classification accuracy. When the microarray cancer data classification is performed, genes related to this particular cancer can also be identified through feature selection. Each gene on the microarray chip is considered as a unique feature.

For the gene selection problem, the goal is to select a few important genes from thousands of genes. Thus, feature selection would be an essential step. Vapnik, Guyon, Weston, and Barnhill (2002) applied the SVM to investigate the gene selection problem, and it was found that 16–64 genes are able to get the best accuracy in acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) cancer classification problems. Cho and Ryu (2002) compared seven feature selection methods in AML and ALL datasets. They selected 30 genes from 7129 genes, and the best accuracy was 94.1%. Zhang, Lee, and Wang (2003) investigated a microarray expression dataset without feature selection. They listed nine advantages and limitations of the SVM on this problem. Fujibuchi and Kato (2007) discussed three classifiers and six kernels in AML and ALL problems. Their method can achieve 97.8% accuracy with a complete feature set. Cho and Won (2007) used another classifier to predict the same problem, and they found that the same feature numbers (around 25–30, as the paper they proposed earlier (Cho & Ryu, 2002)) can also achieve the best accuracy of 97.1%.

Another bioinformatics problem studied in this research is the protein disordered region prediction. The first research on disordered proteins prediction was done by Williams (1978). They noticed the abnormally low charge/hydrophobic ratio for the two disordered proteins, and used this special property for prediction. Uversky, Gillespie, and Fink (2000) did the same analysis but on a much larger set of proteins in 2000, and produced a list of disordered propensity for each amino acid. Peng, Radovojac, Vucetic, Dunker, and Obradovic (2006) developed the VSL2 disordered region predictor which used an output smoothing procedure for its prediction result. The smoothing algorithm is based on calculating the average of raw predictions for neighboring residues within an output window of a size of 61 to remove obvious misclassifications of discontinuous results.

The above-mentioned studies on both problems used some filters and/or wrappers for feature selection. For microarray expression data classification, several approaches were done with different filter models. However, the filters could not guarantee the best result and it only utilized the information of each feature. On the other hand, the wrappers pursue higher prediction accuracy through a machine learning model. However, wrappers cannot be tried in microarray cancer data classification, because the computational time and complexity would be unacceptable. Also for the disordered region prediction problem, lots of features need to be considered. Hence, feature selection of disordered protein data is also important for reducing its number of features. Feature selection not only can point out critical features, but also can decrease the noisy (unrelated) features from the original feature set.

In this research, a hybrid feature selection mechanism was used to solve the two problems. The mechanism takes advantage of both the efficiency of filters and the accuracy of wrappers.

3. A hybrid feature selection mechanism

3.1. Filters vs. wrappers

From the viewpoint of the information theorem, the information of a set of features could be calculated by various statistical measures, and that is the core of the filter type of feature selection methods. Because of the fast calculation, filters are often applied to feature selection in high-dimensional data.

As we can see in Fig. 1, the filters have three main stages: feature set generation, measurement, and tested by a learning algorithm. In the feature set generation stage, a feature subset is generated. Next, the measurement step is performed, which measures the information of the current feature set. While the result does not match the stop criterion, the above steps will be performed repeatedly. In this step, the stop criterion could be a threshold of the measurement results. When the result has not reached the threshold, a new feature set would be generated and the measurement would be performed again. Hence, the final feature set would contain the most informative features. Finally, the testing step is proceeded by a learning algorithm, like SVMs or neural networks (NN). The result includes the testing result of the selected features.

Fig. 2 presents the working procedure of wrappers. It is the same as that of the filters except that the measurement stage is

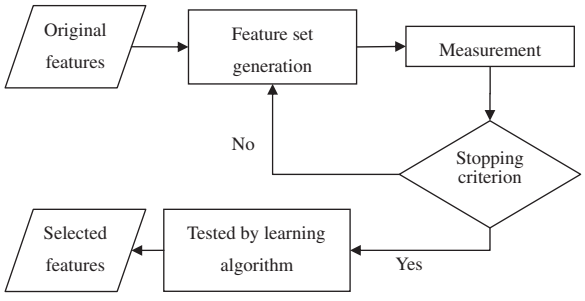

Fig. 1. The filters.

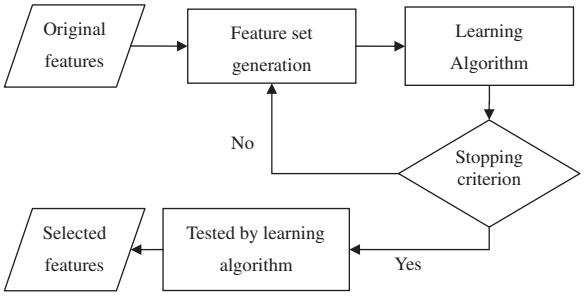

Fig. 2. The wrappers.

replaced by a learning algorithm. And this is the main reason that the wrappers always perform slowly. On the other hand, owing to the learning algorithm, the wrappers could achieve better feature selection results in most cases. For the stopping criterion, when the result starts to get worse or the number of features reaches a predefined threshold, the procedure stops.

Table 1 lists the pros and cons of the filters and the wrappers. The filters process quickly, but their results are not always acceptable. The wrappers have high classification accuracy, but process slowly. In addition, the filters calculate the information from features; therefore, its feature selection results will depend on the measured information of the features. The wrappers use the learning algorithm for the judgment; hence, their classification result is biased by the learning algorithm.

In this study, we propose a new feature selection mechanism which utilizes the advantages of both filters and wrappers. By combining the filters and the wrappers, we not only can improve the classification accuracy of pure filters, but also decrease the processing time of pure wrappers.

3.2. Hybrid feature selection

Fig. 3 shows the hybrid feature selection procedure. Two filter models were chosen as the preliminary screening to remove the most redundant or irrelevant features. *F*-score and information gain are the core of preliminary screening. These two resulted feature sets are combined together as the preprocessed feature set for fine tuning. This step is called the combination model. Finally, the wrapper model is applied to improve the classification accuracy, and this is the fine-tuning step. The following subsections describe these three critical steps in detail.

3.2.1. Preliminary screening

In the first step, we chose *F*-score and information gain to remove redundant and irrelevant features. *F*-score is a novel filter model which calculates the discriminative ability of each feature. That is to say, features with higher *F*-score have better separation ability in classification problems. *F*-score is defined in the following equation.

$$F(i) \equiv \frac{(\bar{x}_i^{(+)} - \bar{x}_i)^2 + (\bar{x}_i^{(-)} - \bar{x}_i)^2}{\frac{1}{n_+ - 1}\sum_{k=1}^{n_+}(x_{k,i}^{(+)} - \bar{x}_i^{(+)})^2 + \frac{1}{n_1 - 1}\sum_{k=1}^{n_-}(x_{k,i}^{(-)} - \bar{x}_i^{(-)})^2} \quad (1)$$

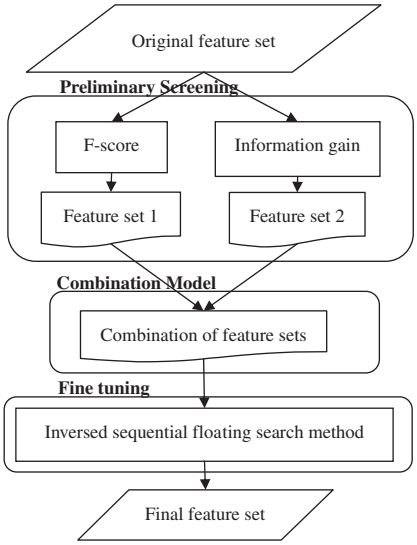

Fig. 3. Hybrid feature selection procedure.

where $\bar{x}_i^{(+)}, \bar{x}_i^{(-)}$ and $\bar{x}_i$ are the averages of the ith feature of the positive, negative and whole datasets; n_+ and n_- are the number of positive and negative instances, respectively; and $x_{k,i}^{(+)}$ and $x_{k,i}^{(-)}$ are the ith feature of the kth positive instance and the ith feature of the kth negative instance.

From Eq. (1), the larger $F(i)$ (F-score) is, the stronger discriminative ability the feature has. This attribute of *F*-score is very suited for the SVM, because the SVM also tries to find an optimal hyper-plane to separate two classes. However, the *F*-score can only examine the discriminative ability of each individual feature. It cannot identify the discriminative ability of multiple features. Hence, features with low scores will be disregarded, even if they are complementary to the top features and might be very useful.

Therefore, we also used the information gain (IG). IG is another filter kind of feature selection. It chooses those candidate features with more information.

$$\text{Entropy}(N) = \sum_{i=1}^{k} P_i \log_k\left(\frac{1}{P_i}\right) = -\sum_{i=1}^{k} P_i \log_k P_i \quad (2)$$

$$\text{Entropy}(D_j) = \sum_{i=1}^{|D_j|} \frac{D_{ji}}{N} \times \text{Entropy}(D_{ji}) \quad (3)$$

$$\text{IG}(D)_j = \text{Entropy}(N) - \text{Entropy}(D) \quad (4)$$

IG concerns how much information each feature can provide. Eqs. (2)–(4) are the steps for calculating IG. In Eq. (2), P_i is the probability of class i, which appears in all N points of data, and this equation calculates the information of all classes. In Eq. (3), D_{ji} means that the jth feature contains i kinds of different values. Eq. (4) calculates IG of the jth feature by finding the difference of Eqs. (2) and (3).

Table 2 shows the strength of *F*-score and information gain. In our research, we want to take advantage on both of them. Therefore, we used both *F*-score and IG in the preliminary screening step of our hybrid feature selection.

Table 1
The comparison of the filters and the wrappers.

Items	The filters	The wrappers
Processing speed	Fast	Slow
Classification accuracy	Depends	High
Depend on learning methods	No	Yes

Table 2
Strength of *F*-score and information gain.

Measure	Strength
F-score	Discrimination ability
Information gain	Information amount

3.2.2. Combination

When the preliminary screening procedure is completed, two feature subsets are selected by *F*-score and IG, respectively. These features are considered as the most class-related features from all features. Putting all of above features together as the final feature set may not be a wise decision. Not only the training or testing procedure of the learning model would take a lot of time, but also the classification accuracy might not be good. The key here is to effectively combine the two feature subsets. To reduce redundant tests in the fine-tuning step, we divide the union of *F*-score and IG's feature sets into two parts: intersection (AND) and exclusive-OR (XOR). Fig. 4 illustrates these two parts.

Feature sets 1 and 2 are selected by *F*-score and IG, respectively. The intersection part of feature sets 1 and 2 is recommended by both *F*-score and IG and the features might be conserved in the final feature set. As for the exclusive-OR part of feature sets 1 and 2, some of the features might be valuable and should be included. Thus, a fine-tuning step was designed to further test these selected features in both interaction and exclusive-OR parts. The wrapper procedure with a machine learning algorithm would further examine the features starting from the intersection part to the exclusive-OR part. The worst fine-tuning result, in respect to feature reduction, is the union of feature sets 1 and 2 (the maximum number of features from the preliminary screening procedure).

3.2.3. Fine tuning

In the previous preliminary screening and combination steps, most redundant and irrelevant features are removed and useful features are kept for the next fine tuning stage. In this stage, we try to take advantage of the wrapper kind of feature selection, and that is to use a searching algorithm and a machine learning model to select a feature set that can result in higher classification accuracy.

The wrappers are not suitable for wide feature set with thousands of features. Owing to the previous feature reduction procedure, the wrappers can be applied now with less computational effort. The sequential floating search method (SFSM) (Pudil, Novovicova, & Kittler, 1994) is modified to fit the fine-tuning procedure, which can avoid the nesting effect caused by using only sequential forward or backward search. The working flow of the SFSM is reversed with the sequential backward search (SBS) performed before the sequential forward search (SFS). The SFSM usually starts from an empty feature set, but our mechanism starts from the intersection part which already includes a set of important features.

In Fig. 5, the SBS starts with the whole feature set and removes one feature at one time, and then a learning model would be applied to test its result. It will be performed repeatedly until the stopping criterion is reached. The procedure stops when the test result starts to get worse or the number of features reaches a predefined threshold. Fig. 6 is the SFS which starts from an empty set and adds one feature at a time. For the stopping criterion, when the test result starts to get worse, the procedure stops.

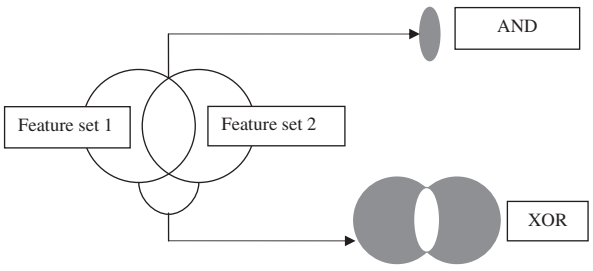

Fig. 4. Intersection and exclusive-OR sets.

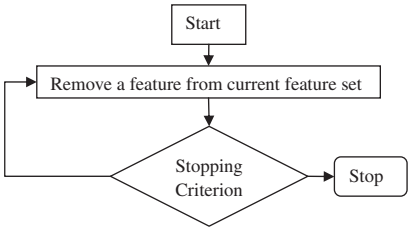

Fig. 5. Sequential backward searching (SBS).

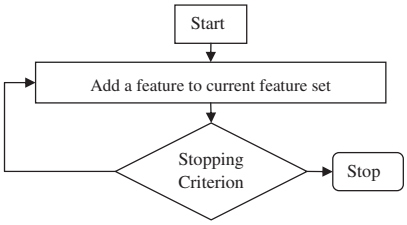

Fig. 6. Sequential forward searching (SFS).

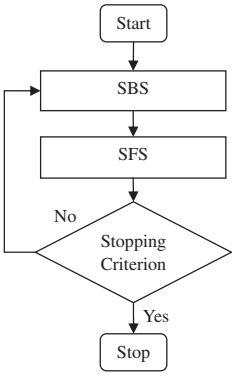

Fig. 7. Inverse sequential floating search method (iSFSM).

The original concept of the SFSM is to use the SFS first, which add and test features one by one. When the learning model's classification accuracy is decreased, the SFS stops and the SBS is then executed to reduce the number of features. These steps will perform repeatedly until the stopping criterion is reached. In Fig. 7, the inverse SFSM (iSFSM) is performed as the following steps: (1) The SBS is executed on the intersection part of the combination model to remove features in this intersection part. In this part, features would be removed if they are not helpful for the learning result. The SBS removes features until the test accuracy is not increased anymore. (2) The SFS searches for features in the exclusive-OR part of *F*-score and IG. Features in this part will be added and tested by the SFS procedure. It will perform repeatedly until the test accuracy drops. (3) The stopping criterion is checked. If the test accuracy has stopped increasing, the iSFSM is completed; otherwise, it performs the SBS and the SFS again.

This procedure limits the search region from the whole feature set to the union set of the preliminary screening results. Also, it starts the sequential floating search from the intersection part, not an empty set or the entire feature set. These two modifications can greatly reduce the processing time comparing to the standard wrapper procedure.

4. Learning model and datasets

4.1. Learning model

In the fine tuning procedure, the wrappers work with a machine learning model. Different kinds of learning models can be applied to wrappers. However, different kinds of learning machines have different discrimination abilities. The SVM keeps training until the separation of classes reaches the maximal margin. Therefore we chose the SVM as the core of fine tuning. As for the kernel function, the most-used RBF kernel was selected.

The SVM has two main advantages. One is that the SVM can project the original data to a higher dimensional space through a nonlinear kernel function. An optimal hyperplane can then be determined to separate the data into two classes. The margin between these two classes is the maximal margin. The other is that the SVM is based on the SV (support vector) learning. That means the SVM selects data points near the optimal hyperplane as its SVs, and uses these SVs for further classification. Fig. 8 shows the maximal margin between two classes, which are separated by the hyperplane, in the SVM model. The two dotted lines are the boundaries. And the nodes which are located near these two lines are the support vectors.

4.2. Disordered protein datasets

We used the protein disordered and ordered data to test our hybrid feature selection mechanism. In general databases, there are no sufficient disordered protein data in a single database. Thus, we used the disordered protein dataset from DisProt database (Vucetic et al., 2005), which collected more than 500 disordered proteins, and randomly selected a dataset from Protein Data Bank (PDB) database (Berman et al., 2000) as ordered examples. Taking datasets from both databases can avoid the training problem with an unbalanced dataset. Finally, 119 protein sequences were collected and there were totally 21,676 residues.

We used three types of protein features, including (1) position-specific scoring matrix (PSSM) (Altschul, Gish, Miller, Myers, & Lipman, 1990) scoring matrix value, (2) statistical analysis within the current sliding window of the protein sequence, and (3) side chain properties of each amino acid in the protein sequence. The PSSM scoring matrix can be calculated by PSI-BLAST (Altschul et al., 1990) tool automatically, a sequence with a length of N is mapped to a matrix of $N \times 20$, which contains the conserved information of the current protein's family. Therefore, the result of PSSM for each position has 20 features which are the scores of 20 amino acids. Next, we calculated the frequency of each amino acid within a predefined window. This information gives the distribution of amino acids within a position's neighborhood. Therefore the number of statistical features is 20 corresponding to 20 amino acids. Finally, side chain properties of each amino acid were also considered. We collected eight side chain properties. They are aliphatic, tiny, small, aromatic, hydropathy index (Kyte-Doolittle), polar, charged, and hydrophobic. Hence, there are totally 440 protein features with a window size of 15. They are PSSM (20 values $\times$ 15 window size = 300), statistical values (20), and side chain properties (8 $\times$ 15 window size = 120).

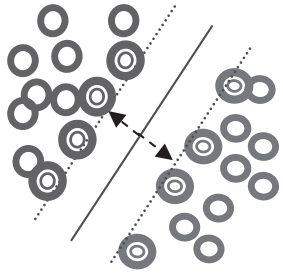

Fig. 8. The SVM could find out the maximum margin and use the SVs to predict the prediction targets. SVs are located on the two boundaries. (Doubled circles are SVs).

4.3. Microarray cancer datasets

In this research, we also tried the microarray cancer data classification problem. We used the AML and ALL (leukemia) dataset and the Lung cancer dataset. These datasets were downloaded from the Kent Ridge Bio-medical Data Set Repository which stores both experimental values and the gene names.

In total, there are 72 samples in the AML and ALL dataset, each with 7,129 features (genes). Forty-seven of them are ALL data, and 25 are AML data. In the Lung Cancer dataset, there are 181 samples; each with 12,533 features (genes). Thirty-one of them are of MPM, and the other 150 samples are of ADCA.

5. Experimental results

5.1. Disordered protein prediction

In the preliminary screening procedure, *F*-score and information gain (IG) are used to filter the features. The results are listed in Table 3. In this procedure, the threshold setting is resolved with a greedy process. We can observe that the accuracies for the two reduced feature sets (82.12% and 82.87%) maintained at the same level as when the original feature set was used (82.75%). Moreover, the number of features was reduced from 440 to 288 and 320, respectively. With removal of the features, the core of wrappers for fine tuning can perform much faster. Next, the feature sets generated in preliminary screening were combined. Table 4 shows the numbers of features with different combinations.

Table 3
Preliminary screening on disordered protein data.

Method	Threshold	Removed features	Final features	Accuracy (5-fold cross validation) (%)
None	–	–	440	82.75
F-score	0.0001	152	288	82.12
IG	0.01	120	320	82.87

Table 4
Combinations of disordered data after screening.

Relationship	Number of features	Accuracy (5-fold cross validation) (%)
Total feature set	440	82.75
F-score $\cup$ IG	355	82.71
F-score $\cap$ IG	253	81.96
F-score XOR IG	102	–

Table 5
Preliminary screening on microarray cancer data.

Dataset	Method	Threshold	Final features	Accuracy (5-fold cross validation) (%)
AML and ALL	–	–	7129	68.06
	F-score	50	873	98.61
	IG	0.64	1510	98.61
Lung cancer	–	–	12,533	86.74
	F-score	100	996	99.45
	IG	0.455	1571	99.45

There are 253 features in the intersection part (*F*-score ∩ IG) which would not be removed in the fine tuning procedure. The procedure would determine which features of the 102 features in the exclusive-OR part (*F*-score XOR IG) should be included. Here, we also list the accuracies of (*F*-score ∪ IG) and (*F*-score ∩ IG). They are very close to the accuracy produced by the original feature set. After taking the iSFSM, totally 97 features were added to the starting feature set of 253 features, which resulted in a set of 350 features and a prediction accuracy of 82.72%. Most of the features from preliminary screening (*F*-score ∪ IG) were kept. This means that the original feature set of 440 features are already very compact for this problem and most features that are not so helpful can be removed solely by the preliminary screening procedure. However, the fine-tuning procedure still plays an important role in ensuring that the filters in the preliminary screening procedure have done a good job. Furthermore, although the prediction accuracy remains about the same, the number of features is reduced by 20.5% (440 → 350). This certainly will accelerate the subsequent process for this domain problem.

5.2. Gene selection for microarray cancer data

In the first step, we again tested two kinds of filters, *F*-score and information gain (IG). The results are listed in Table 5.

In Table 5, the threshold setting is resolved from a greedy process. Originally, the classification accuracy of AML and ALL and Lung Cancer datasets were 68.06% and 86.74%, respectively. After the preliminary screening procedure on *F*-score and IG, the AML and ALL feature set was reduced from 7129 to 873 and 1510, and the prediction accuracy was improved to 98.61%. As for the Lung Cancer dataset, *F*-score and IG reduced the features from 12,533 to 996 and 1571, and also the accuracy was raised to 99.45%.

Table 6
Combinations of microarray cancer data after screening.

Dataset	Relationship	Number of features
AML and ALL	Total feature set	7129
	F-score ∩ IG	276
	F-score XOR IG	1831
Lung cancer	Total feature set	12,533
	F-score ∩ IG	326
	F-score XOR IG	1915

Table 7
Prediction accuracy after fine tuning.

Dataset	Relationship	Number of features	Accuracy (5-fold cross validation) (%)
AML and ALL	*F*-score ∩ IG	276	98.61
	Best feature set	70	98.61
Lung cancer	*F*-score ∩ IG	326	99.45
	Best feature set	70	100

Table 8
The comparison with other methods (AML and ALL).

Methods	Accuracy (%)	# of features
Fujibuchi and Kato (2007)	97.8	170
Cho and Ryu (2002)	94.1	30
Cho and Won (2007)	97.1	50
Proposed method	98.6	70

Table 6 shows the combinations of features with the preliminary *F*-score and IG filtering. From Table 6, the AML and ALL dataset retained totally 2107 (276 + 1831) features for further examination by iSFSM. For the Lung cancer dataset, 2241 (326 + 1915) features were left. Next, we list the results after fine tuning in Table 7. The number of features reduced to 70 for both AML and ALL and Lung Cancer datasets and the classification accuracy was further improved to 98.61% and 100%, respectively. For both cancer datasets, 70 genes (features) were picked as most related to the particular disease. This information is very useful in medicine.

Finally, Table 8 compares our proposed method with other existing feature selection methods on the AML and ALL dataset. The result shows that our method resulted in a better result in classification accuracy. It is quite successful in this example.

5.3. Discussion

From the above results, for the disordered region prediction problem, we can see that the feature set of disordered region is reduced from 440 to 350, while the prediction accuracy is maintained. As for the microarray cancer data classification problem, our model greatly decreases the number of features from thousands to 70 and the accuracies are improved to nearly 100%. The natures of the two bioinformatics problems are quite different. The features of the disordered protein dataset were carefully chosen and there is little room to further remove "unnecessary" features (though our method still reduced the number of features by 90). On the other hand, the genes on a microarray chip are designed for general purpose. So for a particular disease, most genes (features) can be disregarded by the hybrid mechanism. This shows that besides taking advantages of both filters and wrappers, the mechanism can also serve for various kinds of datasets in feature selection.

6. Conclusion

A hybrid feature selection mechanism was proposed and tested in this paper. The idea is to utilize the efficiency of filters and the accuracy of wrappers. A three-step procedure including preliminary screening, combination, and fine tuning, was designed. Preliminary screening and combination can quickly remove most irrelevant features. Fine tuning then further examines the combined feature set. The hybrid mechanism was applied to two bioinformatics problems: disordered protein prediction and microarray cancer data classification. The results show that the mechanism is useful for these two types of feature sets. We believe that it is also applicable to other feature selection problems.

Acknowledgement

This research was partially supported by the research grant NSC# 96-2221-E-032-051-MY2.

References

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *Journal of Molecular Biology, 215*(3), 403–410.

Backstrom, L., & Caruana, R. (2006). C2FS: An algorithm for feature selection in cascade neural networks, IEEE International Joint Conference on Neural Networks. Canada: Vancouver, BC, pp. 4748–4753.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., et al. (2000). The Protein Data Bank. *Nucleic Acids Resource, 28*, 235–242.

Chen, C., & Lin, J. (2003). Libsvm: A library for support vector machines. Available from: <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

Cho, S., & Ryu, J. (2002). Classifying gene expression data of cancer using classifier ensemble with mutually exclusive features. *Proceedings of the IEEE, 90*(11), 1744–1753.

Cho, S., & Won, H. (2007). Cancer classification using ensemble of neural networks with multiple significant gene subsets. *Applied Intelligence, 26*(3), 243–250.

Deisy, C., Subbulakshmi, B., Baskar, S., & Ramaraj, N. (2007). Efficient dimensionality reduction approaches for feature selection, International Conference on Computational Intelligence and Multimedia Applications. India: Sivakasi (pp. 121–127).

Fujibuchi, W., & Kato, T. (2007). Classification of heterogeneous microarray data by maximum entropy kernel. *BMC Bioinformatics, 8*, 267–277.

Guyon, I., Weston, J., Barnhill, S., & Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning, 46*(1–3), 389–422.

Huang, J., Cai, Y., & Xu, X. (2006). A filter approach to feature selection based on mutual information. Proceedings of the Fifth IEEE International Conference on Cognitive Informatics. Beijing: China (pp. 84–89).

Ishida, T., & Kinoshita, K. (2007). PrDOS: Prediction of disordered protein regions from amino acid sequence. *Nucleic Acids Research, 35*, 460–464.

Kohavi, R., & John, G. (1997). Wrappers for feature subset selection. *Artificial Intelligence, 97*, 273–324.

Linding, R., Jensen, L. J., Diella, F., Bork, P., Gibson, T. J., & Russell, R. B. (2003). Protein disorder prediction: Implications for structural proteomics. *Structure, 11*(11), 1453–1459.

Liu, H., Dougherty, E. R., Dy, J. G., Torkkola, K., Tuv, E., Peng, H., et al. (2005). Evolving feature selection. *Intelligent Systems IEEE, 20*(6), 64–76.

Liu, Yue, Yin, Yafeng, Gao, Junjun, & Tan, Chongli (2008). Wrapper feature selection optimized SVM model for demand forecasting. The International Conference on Young Computer Scientists. China: Hunan (pp. 953–958).

Peng, K., Radovojac, P., Vucetic, S., Dunker, A. K., & Obradovic, Z. (2006). Length-dependent prediction of protein intrinsic disorder. *Bioinformatics, 7*, 208–225.

Pudil, P., Novovicova, J., & Kittler, J. (1994). Floating search methods in feature selection. *Pattern Recognition Letters, 15*(11), 1119–1125.

Quinlan, J. R. (1979). *Expert systems in the microelectronic age*. Edinburgh University Press, Scotland: Edinburgh (pp. 168–201).

Uversky, V. N., Gillespie, J. R., & Fink, A. L. (2000). Why are "natively unfolded" proteins unstructured under physiologic conditions? *Proteins, 41*, 415–427.

Vapnik, V., Guyon, I., Weston, J., & Barnhill, S. (2002). Gene selection for cancer classification using support vector machines. *Machine Learning, 46*(1–3), 389–422.

Vucetic, S., Obradovic, Z., Vacic, V., Radivojac, P., Peng, K., Iakoucheva, L. M., et al. (2005). DisProt: A database of protein disorder. *Bioinformatics, 21*(1), 137–140.

Williams, R. J. (1978). The conformational mobility of proteins and its functional significance. *Biochemical Society Transactions, 6*, 1123–1126.

Zhang, J., Lee, R., & Wang, Y. J. (2003). Support vector machine classifications for microarray expression dataset. IEEE International Conference on Computational Intelligence and Multimedia Applications. Xi'an, China (pp. 67–71).