

## COMBINING MULTIPLE FEATURE SELECTION METHODS

**Kwang Lee**

**Under the guidance of Professor Sung-Hyuk Cha  
CSIS, Pace University**

### **Abstract**

*This paper proposes a feature selection method that combines various feature selection techniques. Feature selection has been realized as one of the most important processes in various applications, especially pattern classification problems. When too many attributes are involved, training a machine to classify patterns into their respective classes is seemingly impossible. Hence, selecting good features is necessary. Albeit numerous methods to select features have been proposed, there exists no universal solution for this problem unless one searches all possible subsets of all attributes. Some techniques such as forward selection and backward elimination are feasible in terms of speed, but suffer from the effect of local optima problem. Exhaustive search technique guarantees to find the optimal subset, but it takes too long for users to wait for the output; its computational time complexity is exponential. Hence, we propose first to reduce the number of features to the minimum size, so that exhaustive search technique can handle in reasonable time, using forward selection and backward elimination techniques. In this way, the selected feature set is much better than those from forward selection and backward elimination and computed much faster than the exhaustive search technique. The proposed combined feature selection technique is tested on the off-line signature verification data set.*

### **1. Introduction**

The discovery of hidden knowledge and information management from raw data to good information in large databases has been increasingly challenged in data mining. As a result, the feature selections will take a major role in contributing toward the Knowledge Discovery in database processing. Feature selection is addressed in many fields, such as pattern recognition, statistics, information theory, psychology, and artificial intelligence. In this paper, we concentrate the feature selections in pattern classification.

There is a wealth of algorithms for good feature subset selection to reduce the dimensionality of the feature spaces. Albeit, the interest of feature selection has been around quiet some time, the noteworthy advance has progressed in the last three decades. In addition, continuous improvement has been reported, the efforts to decrease error rates and reject rates for the better performance have not stopped.

Feature Selection is a process to find an optimal subset of features from a large set of features, which includes noisy, irrelevant or redundant features, in order to maximize recognition and classification performances without hampering class distribution. Significant efforts have been made to develop different techniques to meet the requirement of increasing sizes and complexities of feature selection problems. Apparently, the ideal feature selection approach is the exhaustive search on the full set of features to find an optimal subset of features. Subsets of  $2^N$  in total, where  $N$  is the number of features of a data set, are listed and delivered to the prediction process. As a result, an optimal best subset will be selected. However, the exhaustive search is not practical and feasible for a large number of features. In general, the search approaches are classified three different categories: exhaustive, heuristic, and random [21]. The heuristic approach uses domain knowledge to prune the feature space to a manageable size [5] and the random approach sets a maximum number of iterations and stops searching at the iteration limit. Sequence forward selection and sequence backward elimination are considered as the heuristic approach.

The taxonomy of Feature Selection algorithms has been reported in many different ways. Liu and Dash divided into four steps as generation procedure, evaluation function, stopping criterion, and validation procedure. They classified 32 different feature selection methods [5]. Jain and Zongker constructed a hierarchical taxonomy in comparing fifteen search algorithms [22]. Rauber claimed a feature selection algorithm formulated by specifying three ingredients, which are independent from each other. They are a search strategy, a selection criterion, and stopping condition [16]. The search strategy decides the way the combinations of features are tested for a certain quality criterion. The selection criterion assesses the quality of a set of features and provides a ranking possibility for the selection process. The stopping condition is a predefined number of features to be needed for stopping the search process.

When too many attributes are involved, training a machine to classify patterns is seemingly impossible. Hence, selecting good features becomes a necessity. Albeit there have been numerous methods to select features, there exists no optimal solution for this problem unless one searches all possible subsets of all attributes. Some techniques such as forward selection and backward elimination are feasible in terms of speed, but suffer from the effect of local optima problem. On the other hand, exhaustive search technique guarantees to find the optimal subset, but it takes too long for users to wait for the output; its computational time complexity is exponential. To our best knowledge, there is no universal solution to keep abreast the optimal subset from the exhaustive with better runtime from heuristic method like forward selection and backward elimination. Hence we propose first to reduce the number of features to the minimum size, so that the exhaustive search technique can handle in reasonable time, using forward selection and backward elimination techniques. In this way, the selected feature set is much better than those from forward selection and backward elimination and computed much faster than the exhaustive search technique.

Pattern Classification problem is to classify an unknown input to its respective class. To do so requires training a machine using the large instances of known classes. Fig 1.shows

examples of classes, instances, and features. There are two classes called P and N. Each class has nine instances  $P_1, \dots, P_9, N_1, \dots, N_9$ . Each instance is represented by eight attributes or often called features, and features are denoted as  $f_1, \dots, f_8$ . We shall use these notations throughout the rest of this paper.

Class P		Class N	
	$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8$		$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8$
P1	1, 2, 6, 7, 2, 5, 2, 5	N1	4, 5, 7, 7, 1, 5, 2, 7
P2	7, 4, 4, 9, 1, 4, 1, 6	N2	5, 6, 6, 8, 1, 4, 2, 5
P3	5, 3, 2, 7, 3, 2, 1, 6	N3	5, 7, 3, 8, 1, 4, 9, 5
P4	3, 3, 3, 8, 1, 2, 1, 5	N4	6, 6, 6, 8, 2, 4, 9, 5
P5	3, 5, 4, 9, 2, 5, 2, 5	N5	6, 4, 8, 8, 2, 5, 3, 6
P6	3, 4, 2, 9, 3, 2, 2, 5	N6	6, 7, 2, 7, 2, 5, 2, 5
P7	5, 3, 3, 7, 2, 5, 2, 6	N7	7, 2, 6, 7, 1, 5, 2, 4
P8	4, 1, 3, 8, 1, 3, 4, 6	N8	7, 5, 4, 6, 1, 5, 3, 5
P9	5, 2, 1, 7, 3, 4, 4, 6	N9	8, 6, 1, 7, 2, 7, 9, 5

**Fig 1.** Classes, Instances, and Features

There are numerous advantages of selecting smaller number of good features, i.e., features having the more discriminatory power over the irrelevant features. First, the performance improves. Second, computing time is reduced. Typical pattern recognition applications consist of feature extraction and classification stages. A faster pattern recognition system is built when only necessary features are extracted and the classifier takes smaller number of inputs. Finally, the reduced number of features helps understand patterns better than the larger number of features.

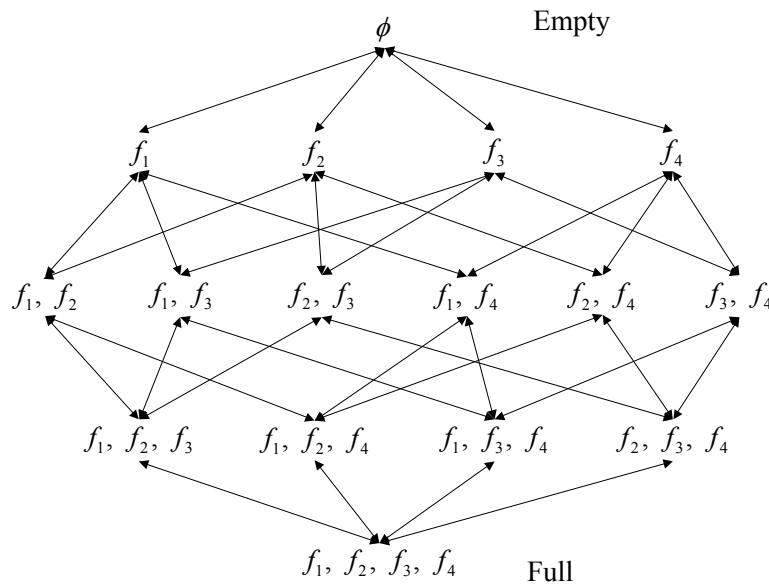
The rest of the paper is organized as follows. Chapter 2 discusses the necessity of feature selection. Chapter 3 shows several feature selection techniques and then presents a proposed method to compensate each technique's disadvantages by combining existing techniques. Chapter 4 demonstrates the effect of the proposed method using the off-line signature verification data set. Finally, Chapter 5 concludes this paper.

## **2. Necessity of Feature Selection**

The necessity of feature selection comes from the famous paradox in pattern classification [2] [4]. The paradox is as follows; when three features are considered, the accuracy of classification should be better than considering only two features. Beyond a certain point, however, the inclusion of additional features leads to lower rather than higher accuracy. Thus, when a very large set of features is present, it is difficult to design a classifier because the accuracy can be poor and instable. In other words, the presence of huge numbers of features often creates many disturbances for performance of inducing

algorithms. In particular, the mix with irrelevant or redundant or noisy features will degrade even more, the classifier's performance accuracy.

The motivation of Feature Selection is illustrated as follows. Suppose one achieves 80% accuracy of a classifier when two features,  $f_1$  and  $f_2$  are used. When another feature, called,  $f_3$ , is introduced, the performance increases to 85%. Now, suppose that when another additional feature,  $f_4$ , is used, the performance be reduced to 84% rather than increasing. This means that there is a subset of  $\{f_1, f_2, f_3, f_4\}$  whose discriminatory power is greater than the entire set. Suppose that the performance of a subset,  $\{f_2, f_4\}$ , is 90%. The goal of the feature selection problem is to find this subset. This search space is illustrated in Fig 2.



**Fig 2.** Search space of Feature Selection

In general, feature selection strategy is to select the best subset of size  $k$  from the given set of  $N$  features. The best subset means a subset that optimizes function  $F(.)$  over all possible subsets of  $n$  features. After features have been collected, we do not know the degree of feature quality, whether they are relevant or noisy or redundant features. Relevant features are mostly unknown in databases and we do not understand the domain or its problems, so we have to discover through experiments.

As we present an illustration above, the rational of monotonicity, which claims adding features cannot decrease the classification accuracy, is solely a theoretical concept. The demand for a large number of instances for an application grows exponentially with the dimensionality of the feature spaces. This is referred to as the curse of dimensionality, which, coined by Bellman [2], forces restrictions on the number of features.

Furthermore, adding more features increases the amount of information available but it causes an adverse impact that degrades the performance of the classifier. This effect is

referred to as a peaking phenomenon. In the result, dimensionality reduction of feature spaces is the foremost important task that is the role of Feature Selection. The dimensionality reduction can be achieved whether Feature Extraction [1][8] or Feature Selection; however, we just focus on Feature Selection.

The number of features at the disposal of the designer of a classification system is usually a very high number. The number can easily become of the order of a few dozen or even hundreds [19]. This is the necessity to reduce the number of features to a sufficient minimum to accomplish the dimensionality reduction in order to overcome the peaking phenomenon and at the same time to retain as much as their class discriminatory information. It is the reason for the necessity of Feature Selection existence on the whole.

### **3. Combining Algorithm**

In the nature of function, Feature Selection algorithms have two components: an evaluation function and a search function. The evaluation part examines the candidate feature sets and selects one that maximizes the evaluation function and the search function locates the subset. In the course of evaluation function, it measures the discriminating power of a feature set.

#### **3.1 Filter and Wrapper**

In general, feature subset selection methods, which were derived from the evaluation function, have been classified in two broad categories as filter and wrapper methods [10]. This classification is based on inclusion or exclusion of algorithms while Feature Selection process. The filter method filters features and passes to an induction algorithm and the wrapper method embraces feature selection around the induction algorithm process.

#### **3.2 Search strategy**

To find the best subset of  $N$  features, a naïve algorithm is to evaluate the entire subsets. As this is often impractical, we consider three heuristic search strategies: sequential backward, sequential forward, and floating search.

##### **Algorithm 1. Sequential Backward**

- Begin with the full set of selected features
- Select the best one for deletion.
- Do until  $d$  features remain

##### **Algorithm 2. Sequential Forward**

- Begin with the best selected feature
- Next, select the best one to add
- Do until  $d$  features selected

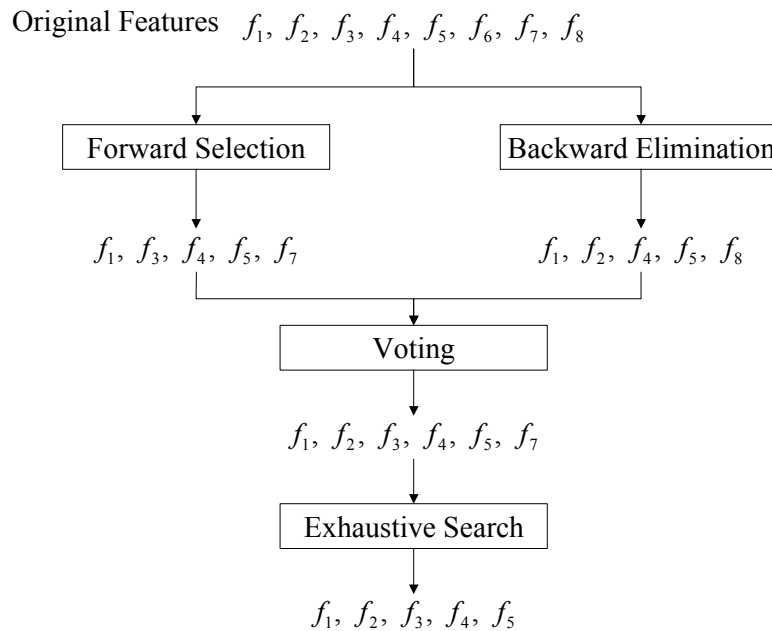
##### **Algorithm 3. Floating Search**

Forward:  
 Select the best one from  $y_{m-k}$  to add to  $x_k$  and check if we can exchange it for another one in  $x_k$  to increase  $c(x_k)$ .  
 If Yes, then go to Backward  
 If No, then add new one to  $x_k$ , like as  $x_{k+1}$ , in

Increment k and repeat  
 Backward:  
 Select the best one from  $x_k$  to eliminate and check if it yields a better  $x_{k-1}$   
 If Yes, then delet one form  $x_k$ , yielding  $x_{k-1}$ , decrement k, and repeat  
 If No, then go to Forward.

There have been many different angles of approaches to select an optimal subset without sacrificing to perform a given task. In this paper, we utilize the algorithm 1 and 2 for a combined feature selection method.

### 3.3 Combined Feature Selection Procedure



**Fig 3.** Combined Feature Selection procedure

Fig. 3 illustrates the proposed combined Feature Selection procedure. The first step is that Forward Selection and Backward Elimination methods apply to the original feature set and produces two minimum number of intermediate feature subsets. The second step consolidates two subsets into a pool and selects the same number of features with the number of each subset based on majority votes. In case of same vote counts rewarded for two features, the one who produces the better performance rate of pattern classification in combining with the rest of features will be selected. In the third step, Exhaustive method will use the voted features for the selection of the final feature set.

## **4. Experiment**

The newly proposed combined feature selection technique is tested on a user verification system employing off-line signatures. The user verification problem can be formalized as a pattern classification problem that categorizes the input pattern into one of two classes: valid user or invalid user.

This problem plays an important role in systems that require high-confidence security. Off-line signature can be utilized to verify the authenticity of the user. Signature verification involves the comparison of a test signature with one or several reference signatures that have been collected when the user enrolls in the system (see the extensive survey [13][14][15][18]).

We approach the automatic signature verification problem as a two-class classification problem, so called dichotomy: valid user or invalid user [3]. Features are extracted from the signatures and distances between the features of the input and reference signatures are computed. As a result, since these features and distances are numeric, we have multivariate numeric patterns.

Each of the collected signature samples is scanned and digitized. After scanning and digitization, feature extraction techniques available in image processing and optical character recognition areas [18] are used to obtain characteristic features of handwriting. They are coarse and fine mesh density, gradient direction, micro-structural features, two directional projection features, bounding box, centroid, aspect ratio, vertical correlation, vertical, horizontal and center symmetries, ascenders, descenders, word length, gaps between words, etc. [18].

Using these features, the time elapsed for each feature selection techniques will be reported and the advantage of the proposed combined method will be reported in [11].

## **5. Conclusion**

In this paper, we combined various feature selection techniques to design a better feature selector. Forward selection and backward elimination techniques were first used to select candidate features. By voting, a good subset of all features was generated where the size of the subset is much smaller than the original full set but larger than the desirable subset size. We reduced the number of features to the minimum size, so that the exhaustive search technique can handle it in a reasonable time.

Forward selection and backward elimination techniques are feasible in terms of speed, but suffer from the effect of the local optima problem. Exhaustive search technique guarantees to find the optimal subset, but it takes too long for users to wait for the output; its computational time complexity is exponential. Hence, we proposed a method that compensates each technique's disadvantages by combining them; the selected feature set

is much better than those from forward selection and backward elimination and computed much faster than the exhaustive search technique.

The proposed technique is successfully applied to an off-line signature verification dataset. When the artificial neural network is used with the reduced number of features, we achieved as a good performance as when all features are used. This reduces great amount of time to extract unnecessary features as well as the training and classifying time.

## **Reference**

- [1] Steve De Backer, Antoine Naud and Paul Scheunders. Non-linear Dimensionality Reduction Techniques for Unsupervised Feature Extraction, *Pattern Recognition Letters*, vol. 19, pp711-720, 1998.
- [2] Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1957.
- [3] Sung-Hyuk Cha, and Sargur N. Srihari, Writer Identification: Statistical Analysis and Dichotomizer, In *Proceedings of SPR & SSPR, LNCS-Advanced in Pattern Recognition*, vol 1876 p123-132, 2000
- [4] Edgar Chavez, Gonzalo Navarro, Ricardo Beeza-Yates and JoseL. Marroquin. *Searching in metric spaces*. Technical Report TR/DCC-99-3, DCC. University of Chile, June 1999.
- [5] Manoranjan Dash and Huan Liu. Feature Selection for Classification. *Intelligent Data analysis*, Vol. 1, no. 3,1997.
- [6] Byron Dom, Jacob Sheinvald and Wayne Niblack, Feature Selection with Stochastic Complexity, *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 241-248, Rosemont, IL., 1989.
- [7] Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*, John Wiley & Sons, Second Edition, 2001.
- [8] Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press London, 1990.
- [9] Anil Jain, Robert Duin and Jianchang Mao. Statistical Pattern Recognition: A Review”, *IEEE Transactions on Pattern Analysis and Machine Intelgence*, Vol 22, No1, January 2000.
- [10] George John, Ron Kohavi and Karl Pfleger. Irrelevant Features and the subset Selection problem. *Proceedings of the Eleventh International Conference on Machine*

Learning. pp. 121-129, New Brunswick, Morgan Kaufmann, 1994.

[11] Kwang Lee. Procedural Feature Selection in Pattern Classification, CSIS Pace university, DPS-dissertation, 2002.

[12] Tom M. Mitchell. *Machine Learning*, McGraw-Hill, 1997.

[13] Réjean Plamondon, Special issue on automatic signature verification, *Pattern Recognition*, Vol. 8 No. 3 June 1994

[14] Réjean Plamondon and Guy Lorette, Automatic Signature Verification and Writer Identification – The State of the Art, Vol. 22 No. 2 pp. 107-131, 1989

[15] Réjean Plamondon, and Sargur N. Srihari, On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.12 No.8 pp. 63-84, 2000

[16] Thomas Walter Rauber. *Pattern Recognition. Journey of Actualization in Computer Science*. xvii Congress of the Brazilian Computer Science Society, Brasilia, Brazil, 1997.

[17] Sarunas Raudys and V. Pikelis. *On Dimensionality, sample size, classification error, and Complexity of Classification Algorithms in Pattern Recognition*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 2, no. 3, pp. 243-252, 1980.

[18] Charles C. Tappert, Ching Y. Suen, and Toru Wakahara, *The State of the Art in On-Line Handwriting Recognition*, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 12 No. 8 pp. 787-808, 1990.

[19] Sergios Theodoridis and Konstantinos Koutroumbas. *Feature Selection*. Pattern Recognition, Academic Press. 1998.

[20] N. Wyse, Richard Dubes and Anil Jain. A critical evaluation of intrinsic dimensionality algorithms. In Gelsema, E. and Kanal, L., editors, *Pattern Recognition in Practice*. pp 415-425. Morgan Kaufmann Publishers, Inc., 1980.

[21] Jihoon Yang and Vasant Honavar. Feature Subset Selection using a Genetic Algorithm, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, 1997.

[22] Douglas Zongker and Anil Jain. Algorithms for Feature Selection: An Evaluation. *Proceedings of the 1996 International Conference on Pattern Recognition*. 1996.