OFFSS: Optimal Fuzzy-Valued Feature Subset Selection

E. C. C. Tsang, D. S. Yeung, and X. Z. Wang

***Abstract*—Feature subset selection is a well-known pattern recognition problem, which aims to reduce the number of features used in classification or recognition. This reduction is expected to improve the performance of classification algorithms in terms of speed, accuracy and simplicity. Most existing feature selection investigations focus on the case that the feature values are real or nominal, very little research is found to address the fuzzy-valued feature subset selection and its computational complexity. This paper focuses on a problem called optimal fuzzy-valued feature subset selection (OFFSS), in which the quality-measure of a subset of features is defined by both the overall overlapping degree between two classes of examples and the size of feature subset. The main contributions of this paper are that: 1) the concept of fuzzy extension matrix is introduced; 2) the computational complexity of OFFSS is proved to be NP-hard; 3) a simple but powerful heuristic algorithm for OFFSS is given; and 4) the feasibility and simplicity of the proposed algorithm are demonstrated by applications of OFFSS to fuzzy decision tree induction and by comparisons with three different feature selection techniques developed recently.**

***Index Terms*—Computational complexity, data mining, feature subset selection, fuzzy-valued feature, learning.**

I. Introduction

FEATURE subset selection is a well-known pattern recognition problem which is usually viewed as a data mining enhancement technique. This technique aims to reduce the number of features to be used, i.e., to reduce the entire feature space to a highly predictive subset of the space. This reduction may improve the performance of data mining algorithms to be used, in terms of speed, accuracy, and simplicity. In addition, because of this reduction, the identification of features which do not need to be stored, collected or bought, may bring financial savings [19].

The previous study on feature subset selection focused mainly on the statistical approaches such as the typical principle component analysis (PCA) method [18] and the linear discriminant analysis (LDA) method [9]. These methods attempt to reduce the dimensionality of input data by creating new features that are linear combinations of the original ones. The main drawback of these methods is that the new features (compared with the original ones) do not have true meaning. Moreover for PCA, simply scaling of the features can cause serious changes to the results.

Manuscript received August 22, 2001; revised October 15, 2001. This work was supported by the Hong Kong Polytechnic University under Research Project G-T209.

E. C. C. Tsang and D. S. Yeung are with the Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: csetsang@comp.polyu.edu.hk).

X. Z. Wang is with the Machine Learning Center, College of Mathematics and Computer Science, Hebei University, Baoding, Heibei, China.

Digital Object Identifier 10.1109/TFUZZ.2003.809895

An extensive amount of research has been conducted over the last two decades to obtain reliable approaches for feature selection. Blum [3] had given an excellent survey for selection of relevant features in machine learning. These approaches are different in the evaluation of feature subsets. A number of evaluation criteria such as gain-entropy [25], relevance [1], contingency table analysis [26] have been developed for feature values which can be real, symbolic, categorical or nominal.

Neuro-fuzzy approaches, e.g., [20], [21], [38], and [7] are usually based on an overall feature evaluation index (OFEI). These approaches view each class as a fuzzy subset, and according to the classification information entropy, define an overall feature evaluation index for a subset of features, and then use some searching technique to approximately find the solution. Neural network feature selector [29] and fuzzy feature selection [28] should be special cases of neuro-fuzzy technique. The main drawback of neural network feature selector is that the network usually suffers from the local minimum and slow convergence.

Appropriate features can be selected by genetic algorithms (GAs) [4], [27] where each feature subset (called a chromosome) is evaluated by a fitness function during an optimization cycle. In contrast to other feature selection techniques, GA can generate approximately a number of optimal feature subsets.

A different approach in feature selection is based on neural network output sensitivity, which uses a feature quality index (FQI) for each feature and sorts the features according to FQI values. The method to evaluate the value of FQI can be different. For example, Zurada in [42] and Engelbrecht in [8] used partial derivatives of the output with respect to the input to define the sensitivity measure and compute its value by Taylor approximate expansion. Yeung in [39] used the variance of the output error with respect to the input perturbation to define the sensitivity measure and Zeng in [41] used the expected value of output error with respect to the input change.

The mutual information-based feature selector [2] and [22] is universally accepted as a promising method to feature selection. This method considers mainly the dependence between features and selects some features with high independence. It can be briefly formulated as a FRn-k problem: Given an initial set F with n features and C set of all output classes, find the subset S in F with k features that minimize the entropy $H(C|S)$, i.e., maximize the mutual information $I(C|S)$.

All feature subset selection algorithms have two key components. One is the measure of the quality of a set of features. It concerns some measure of the predictive power of the features, as well as the size of the feature subset. The other is the search strategy to find the best feature subset as defined by the measure. It is worth noting that the enumerative search for all possible feature subsets is generally infeasible if the

considered database contains many records. Most researchers on feature subset selection try to show that their methods are computationally efficient in these two aspects. The means of research is usually restricted to experiments and comparisons. Obviously, they often suffer from the lack of theoretical analysis due to the fact that the study of an important theoretical issue, i.e., the computational complexity of optimal feature subset selection, is neglected. One may want to know, for example, whether or not there exists an exact feasible algorithm to find the best subset for fuzzy-valued features.

With the development of knowledge-based systems, the imprecise data such as "about 28," "young," "very big," "hot," and so on is considered in the learning phase of constructing expert systems. The imprecise feature-values in traditional data mining are usually regarded as either real numbers (the continuous case) or nominal symbols (the discrete case). There seems to be a gap between the two cases since real numbers have linear ordering and nominal symbols has no ordering at all. This gap may be filled by viewing the imprecise data (linguistic terms) as fuzzy sets. So far, very little work is found to address the selection of optimal fuzzy-valued feature subsets and its computational complexity. The only found references are [30], [32] where the focus is the fuzzy target (model) selection by using fuzzy clustering (fuzzy c-means) technique, rather than the fuzzy-valued feature selection.

This paper focuses on a problem of optimal fuzzy-valued feature subset selection (OFFSS). The measure of the quality of a set of features is defined by the overall overlapping degree between two classes of examples and the size of feature subset. The computational complexity of OFFSS is investigated by the introduction of fuzzy extension matrix. A heuristic search algorithm is proposed for the optimal feature subset selection. This algorithm finds a path in the extension matrix. Applications of OFFSS are discussed for fuzzy decision tree induction schema. The present paper has the following organization. Section II gives a formal definition of OFFSS, Section III investigates the computational complexity of OFFSS, Section IV proposes our heuristic algorithm for OFFSS, Section V studies the application of OFFSS to fuzzy decision tree induction and the comparison with three different feature selection techniques developed recently, and the last section offers conclusions of this paper.

II. Definition of OFFSS

Before giving a rigorous definition of OFFSS problem, we first review some notations and concepts used in this paper. Throughout this paper, for a given universe of discourse X, $F(X)$ denotes the set of all fuzzy subsets defined on X.

Definition 1: A mapping from $F(X)\times F(X)$ to [0, 1], SM, is called a similarity measure if SM satisfies that (1) $\mathrm{SM}(A,B) = \mathrm{SM}(B,A)$ for any $A, B \in F(X)$ and (2) $\mathrm{SM}(A,B) = 1$ whenever $A = B$.

The similarity measure between two fuzzy subsets can be defined by their membership functions. Discussions on similarity metrics can be found in many articles [43], [33], [24], [13], [35]. The following are two frequently used forms:

$$(1).\ \mathrm{SM}_1(A,B) = (1 + DM(A,B))^{-1}$$
$$(2).\ \mathrm{SM}_2(A,B) = \vee_{i=1}^{n} (A(x_i) \wedge B(x_i))$$

in which $\vee$ and $\wedge$ denote max and min, respectively, and $DM(A,B)$ denotes the distance measure of A and B and is defined as

$$DM(A,B) = \sqrt[r]{\sum_{i=1}^{n} |A(x_i) - B(x_i)|^r} \qquad (r \geq 1).$$

It is clear that the aforementioned distance equation is Euclidean metric when $r = 2$. Our study on OFFSS is based on a similarity measure between two fuzzy sets. It is worth noting that there exist many forms of similarity measure between two fuzzy sets. We cannot guarantee that our selected two equations have the best performance for the investigated feature selection problem. However, some experiments have shown that our proposed method is not much sensitive to the choice of similarity measure.

Now, let us consider a group of examples (objects, instances, cases) and a feature space $FS = \{F_1, F_2, \ldots, F_m\}$. Each F_i ($1 \leq i \leq m$), called a fuzzy-valued feature or a fuzzy-valued attribute, is supposed to take value in $F(X_i)$ (X_i is a universe of discourse). Each example e is characterized by the m features, that is, $e = (v_1, v_2, \ldots, v_m)$ in which $v_i = e(F_i)$ is the value of example e with respect to F_i ($i = 1, 2, \ldots, m$). For any feature, the similarity measure between two feature-values is written as SM. This group of examples is supposed to be classified into two classes, P and N, called positive class and negative class, respectively.

Definition 2: Let $e = (v_1, v_2, \ldots, v_m)$ be an example and $S = \{F_{i_1}, F_{i_2}, \ldots, F_{i_n}\}$ a given feature subset ($S \subset FS$, $n \leq m$). The notation $e|S$ is used to denote $(v_{i_1}, v_{i_2}, \ldots, v_{i_n})$.

Definition 3: Let S be a given feature subset ($S \subset FS$), p_e a positive example ($p_e \in P$), and n_e a negative example ($n_e \in N$). The similarity degree between p_e and n_e on S is defined as

$$\mathrm{SM}(p_e|S, n_e|S) = \wedge_{F_i \in S} \mathrm{SM}\,(p_e(F_i), n_e(F_i))$$

in which the notation $\wedge$ denotes Min. Particularly, $\mathrm{SM}(p_e, n_e) = \mathrm{SM}(p_e|FS, n_e|FS)$.

Definition 4: For a given feature subset S ($S \subset FS$), the overlapping degree of the positive class P and the negative class N is defined as

$$OV(P,N|S) = \vee_{p_e \in P} \vee_{n_e \in N} \mathrm{SM}\,(p_e|S, n_e|S)$$

in which the notation $\vee$ denotes Max.

To make Definition 4 clear, we restrict ourselves to crisp case. For two given examples a and b, a given feature subset S, and a given similarity measure SM, one can consider that the degree of similarity between $a|S$ and $b|S$ is equal to 1 if and only if $a|S = b|S$ and is equal to 0 if and only if $a|S \neq b|S$. This consequent implies that $OV(P,N|S) = 0$ if and only if $(P \cap N)|S = \phi$ and $OV(P,N|S) > 0$ if and only if $(P \cap N)|S \neq \phi$ where ϕ denotes empty set and $OV(P,N|S)$ is the overlapping degree given in Definition 4. That is, Definition 4 shows whether the intersection of two sets is empty for crisp case. Therefore, when fuzzy case is considered, Definition 4 can naturally be regarded as the maximal degree of overlapping (intersection) of two fuzzy sets.

From Definition 3, one can see that the similarity degree $\mathrm{SM}(p_e|S, n_e|S)$ will become small as the cardinality of feature subset S increases. Hence, the overlapping degree $OV(P, N|S)$ will also decrease as the cardinality of S increases. For an appropriate threshold T ($T \geq OV(P, N|FS)$ in which FS is the entire feature set), there always exists at least one feature subset S with properties: 1) $S \subset FS$; 2) $OV(P, N|FS) \leq T$; and 3) the cardinality of S attains minimum. This is the concept of OFFSS which is formulated in the following Definition 5.

Definition 5 (OFFSS): Let P denote a given class of positive examples, N a class of negative examples, FS the entire set of fuzzy-valued features, and T a given threshold. The problem of OFFSS is to seek a feature subset S^* ($S^* \subset FS$) such that

$$|S^*| = \mathrm{Min}_{S \subset FS} \{|S| : OV(P, N|S) \leq T\}$$

where $|\bullet|$ denotes the cardinality of a crisp set.

We use the following simple example to illustrate the above notations and definitions.

Example 1: Consider a set of examples shown in Table I. This set of examples is classified into $P = \{1, 2\}$ and $N = \{3, 4, 5, 6\}$ and is described by four fuzzy-valued features. The entire feature set is $FS = \{A, B, C, D\}$. Each feature takes value from three fuzzy linguistic terms (fuzzy sets), Small, Mid, and Big, of which the membership functions are shown in Fig. 1. We would like to find the optimal feature subsets for threshold $T = 0.25$.

According to Definition 5, we can use the following algorithm to find the optimal feature subsets.

```
Step 1. Determine the similarity measure.
Step 2. For each pair of different lin-
guistic terms, evaluate the similarity.
Step 3. For each feature subset S, use
Definition 3 to evaluate OV(P,N|S).
Step 4. Determine feature subsets, of
which the value OV(P,N|S) is less than or
equal to T.
Step 5. From the feature subsets obtained
in Step 4, select the ones with minimum
cardinality.
```

Now, we illustrate the algorithm by Example 1. Step 1: Define a similarity measure as $\mathrm{SM}(A, B) = \mathrm{Max}_{x \in X} (\min(A(x), B(x)))^2$. Step 2: According to the defined similarity measure, evaluate similarities $\mathrm{SM}(\mathrm{Small}, \mathrm{Small}) = \mathrm{SM}(\mathrm{Mid}, \mathrm{Mid}) = \mathrm{SM}(\mathrm{Big}, \mathrm{Big}) = 1$, $\mathrm{SM}(\mathrm{Small}, \mathrm{Mid}) = \mathrm{SM}(\mathrm{Mid}, \mathrm{Big}) = 0.25$, and $\mathrm{SM}(\mathrm{Small}, \mathrm{Big}) = 0$. Step 3: Following Step 2, evaluate the value of overlapping degree $OV(P, N|S)$ for subset S. For instance

$$\begin{aligned} OV(P, N|\{A, B\}) =& (0 \wedge 1) \vee (1 \wedge 0.25) \vee (1 \wedge 0.25) \\ & \vee (0 \wedge 0.25) \vee (0 \wedge 0.25) \\ & \vee (1 \wedge 0) \vee (1 \wedge 0) \vee (0 \wedge 1) \\ =& 0.25. \end{aligned}$$

The results are listed in Table II. Step 4: Determine feature subsets with $OV(P, N|S) \leq 0.25$. Table II shows 7 feature subsets $\{A, B\}, \{B, C\}, \{A, B, C\}, \ldots, \{A, B, C, D\}$ satisfying $OV(P, N|S) \leq 0.25$. Step 5: The subsets with minimum cardinality are $\{A, B\}$ and $\{B, C\}$. Therefore, the output optimal feature subsets are $\{A, B\}$ and $\{B, C\}$.

TABLE I
GROUP OF EXAMPLES WITH FOUR FUZZY-VALUED FEATURES

Case.	Feature A	Feature B	Feature C	Feature D	Class
1	Small	Mid	Mid	Small	Positive
2	Small	Big	Big	Mid	Positive
3	Big	Mid	Big	Small	Negative
4	Small	Small	Big	Big	Negative
5	Small	Small	Mid	Small	Negative
6	Big	Big	Small	Mid	Negative

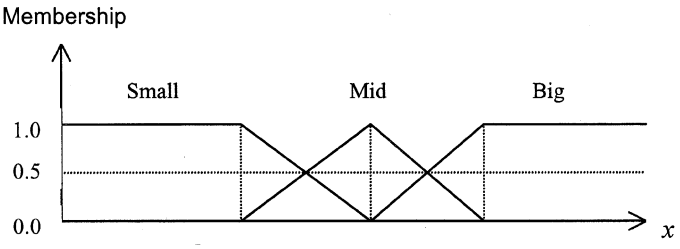

Fig. 1. Three membership functions.

It is worth noting that the previous enumeration algorithm for finding optimal feature subsets is not practical due to its exponential complexity. Developing heuristic algorithms are necessary.

Now, let us give a geometrical explanation of the OFFSS problem (Definition 5). It can be obtained by considering examples E1 and E2, shown in the following table:

No.	X	y	Class
E1.	Small	Mid	Positive
E2.	Big	Mid	Negative

with two features x and y.

Intuitively or by the Definition 5, the feature x can be regarded as the best (optimal feature subset). Fig. 2 gives us a very clear geometrical explanation for the OFFSS.

III. COMPUTATIONAL COMPLEXITY OF OFFSS

In this section, we investigate the computational complexity of OFFSS problem using the concept of extension matrix. The extension matrix, which plays an important role in studying the theory of learning from crisp examples [11], is initially introduced for crisp case in [10] and is extended to fuzzy case in this paper by using similarity measure.

A. Extension Matrix of Fuzzy Case

We continue to use the notations introduced in the previous section. That is, P denotes the positive class, N denotes the negative class, and FS denotes the entire feature space. Each example takes value in the form of m-dimensional vector in which components are fuzzy sets, and SM denotes a given similarity measure between fuzzy sets.

Definition 6: Let T be a given threshold and FS be the entire set of features, $S \subset FS$, $e^+ = (v_1^+, v_2^+, \ldots, v_m^+) \in P$, and

TABLE II
FEATURE SUBSETS AND DEGREES OF INTERSECTIONHOOD ABOUT TABLE I

S	OV(P,N | S)	S	OV(P,N | S)
ϕ	-----	{B, C}	0.25
{A}	1.00	{B, D}	1.00
{B}	1.00	{C, D}	1.00
{C}	1.00	{A, B, C}	0.25
{D}	1.00	{A, B, D}	0.25
{A, B}	0.25	{A, C, D}	0.25
{A, C}	1.00	{B, C, D}	0.25
{A, D}	1.00	FS	0.25

$e^- = (v_1^-, v_2^-, \ldots, v_m^-) \in N$. e^+ and e^- are said to be T consistent with respect to S if $\mathrm{SM}(e^+|S, e^-|S) \le T$. e^+ and N are said to be T consistent with respect to S if e^+ and e^- are T consistent with respect to S for arbitrary $e^- \in N$. P and N are said to be T consistent with respect to S if e^+ and e^- are T consistent with respect to S for arbitrary $e^+ \in P$ and arbitrary $e^- \in N$.

From Definitions 5 and 6, one can easily see that P and N are T consistency with respect to the optimal fuzzy-valued feature subset S^*.

Definition 7: The extension matrix of a positive example $e^+ = (v_1^+, v_2^+, \ldots, v_m^+)$ with respect to a negative example $e^- = (v_1^-, v_2^-, \ldots, v_m^-)$ is defined as $\mathrm{EM}(e^+, e^-) = (r_1, r_2, \ldots, r_m)$ where $r_j = \mathrm{SM}(v_j^+, v_j^-)$ for $j = 1, 2, \ldots, m$. If $r_j \le T$ $(1 \le j \le m)$, then r_j is called an under_T element of extension matrix. (In the crisp case [10], the under_T element is called nondead element).

Definition 8: Let $P = \{e_1^+, e_2^+, \ldots, e_K^+\}$ and $N = \{e_1^-, e_2^-, \ldots, e_L^-\}$. The extension matrix of e_j^+ $(1 \le j \le K)$ with respect to N is defined as $\mathrm{EM}(e_j^+, N)$ and the extension matrix of P with respect to N is defined as $\mathrm{EM}(P, N)$, where

$$\mathrm{EM}(e_j^+, N) = \begin{bmatrix} \mathrm{EM}(e_j^+, e_1^-) \\ \vdots \\ \mathrm{EM}(e_j^+, e_L^-) \end{bmatrix}_{L \times m}$$

$$\mathrm{EM}(P, N) = \begin{bmatrix} \mathrm{EM}(e_1^+, N) \\ \vdots \\ \mathrm{EM}(e_K^+, N) \end{bmatrix}_{KL \times m}.$$

Example 2: Let us continue to discuss the six examples given in Example 1 where $K = 2$, $L = 4$, and $m = 4$. The extension matrix of the first positive example with respect to N, $\mathrm{EM}(e_1^+, N)$, is shown in Fig. 3; and the extension matrix of P with respect to N, $\mathrm{EM}(P, N)$, is shown in Fig. 4.

Definition 9: Let $e = (v_1, v_2, \ldots, v_m)$ be an example denoting a row of the extension matrix and $S = \{F_{i_1}, F_{i_2}, \ldots, F_{i_n}\}$ be a given feature subset. The term "S place of e" is used to denote $\{i_1, i_2, \ldots, i_n\}$.

Theorem 1: Let T be a given threshold and $S = \{F_{i_1}, F_{i_2}, \ldots, F_{i_n}\}$ be a feature subset. P and N are T consistent with respect to S if and only if there exists at least one under_T element in the S place of each row of extension matrix of P with respect to N.

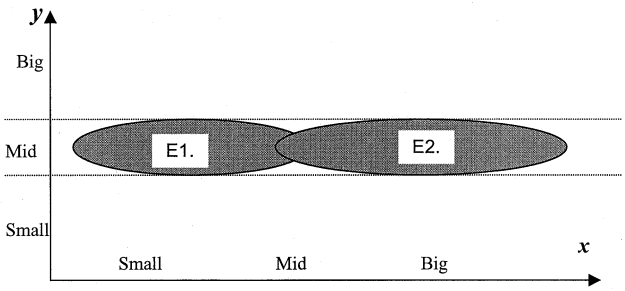

Fig. 2. Geometrical explanation of OFFSS (feature subset $\{x\}$ is the best).

$$\begin{bmatrix} 0.00 & 1.00 & 0.25 & 1.00 \\ 1.00 & 0.25 & 0.25 & 0.00 \\ 1.00 & 0.25 & 1.00 & 1.00 \\ 0.00 & 0.25 & 0.25 & 0.25 \end{bmatrix}$$

Fig. 3. Extension $\mathrm{EM}(e_1^+, N)$ about Table I.

$$\begin{bmatrix} 0.00 & 1.00 & 0.25 & 1.00 \\ 1.00 & 0.25 & 0.25 & 0.00 \\ 1.00 & 0.25 & 1.00 & 1.00 \\ 0.00 & 0.25 & 0.25 & 0.25 \\ 0.00 & 0.25 & 1.00 & 0.25 \\ 1.00 & 0.00 & 1.00 & 0.25 \\ 1.00 & 0.00 & 0.25 & 0.25 \\ 0.00 & 1.00 & 0.00 & 1.00 \end{bmatrix}_{8 \times 4}$$

Fig. 4. Extension matrix $\mathrm{EM}(P, N)$ about Table I.

Proof: Let $P = \{e_1^+, e_2^+, \ldots, e_K^+\}$, $N = \{e_1^-, e_2^-, \ldots, e_L^-\}$, $e_i^+ = (v_{i1}^+, v_{i2}^+, \ldots, v_{im}^+)$ $(1 \le i \le K)$ and $e_j^- = (v_{j1}^-, v_{j2}^-, \ldots, v_{jm}^-)$ $(1 \le j \le L)$. If there exists at least one under_T element in the S place of each row of extension matrix of P with respect to N, then for each i($1 \le i \le K$) and each j $(1 \le j \le L)$ there exists at least one integer k ($k \in \{i_1, i_2, \ldots, i_n\}$) such that $\mathrm{SM}(v_{ik}^+, v_{jk}^-) \le T$. From Definition 3, one can obtain that

$$\begin{aligned} \mathrm{SM}(e_i^+|S, e_j^-|S) &= \wedge_{F \in S} \mathrm{SM}\left(e_i^+(F), e_j^-(F)\right) \\ &= \wedge_{t=1}^{n} \mathrm{SM}\left(v_{ik_t}^+, v_{jk_t}^-\right) \le T \end{aligned} \quad (1)$$

which results in the T consistency of P and N with respect to S. Conversely, if P and N are T consistent with respect to S, then (1) is valid for all i and all j $(1 \le i \le K, 1 \le j \le L)$. This implies that there exists at least one integer k ($k \in \{i_1, i_2, \ldots, i_n\}$) such that $\mathrm{SM}(v_{ik}^+, v_{jk}^-) \le T$ for all i and j $(1 \le i \le K, 1 \le j \le L)$. Therefore, according to Definition 7, there exists at least one under_T element in the S place of each row of the extension matrix of P with respect to N. This completes the proof.

B. Path of Extension Matrix and Feature Subset

Definition 10: A path of an extension matrix refers to a connection of its under_T elements which are obtained by selecting one under_T element from each row of the extension matrix.

Example 3: Consider the extension matrix $\text{EM}(e_1^+, N)$ shown in Fig. 3. If we take the threshold $T = 0.25$, then there exist several paths in this extension matrix. One of them can be $r_{11} \to r_{22} \to r_{32} \to r_{41}$ which is indicated by arrows in Fig. 3.

The following theorem gives us the relation between an optimal feature subset and a path of extension matrix.

Theorem 2: Let T be a given threshold and $\text{EM}(P, N)$ be the extension matrix of P with respect to N. Then, looking for an optimal feature subset is equivalent to searching for a path in the extension matrix $\text{EM}(P, N)$ which involves the minimum number of columns.

Proof: Let S be a feature subset ($S \subset FS$). Then, according to Definition 5 and Definition 6, one knows that S is an optimal feature subset if and only if 1) P and N are T consistent with respect to S and 2) the cardinality of S reaches a minimum.

1) By Theorem 1, P and N are T consistent with respect to S if and only if there exists at least one under_T element in the S place of each row of extension matrix of P with respect to N. Therefore, the given feature subset S can correspond to a path of extension matrix $\text{EM}(P, N)$. This path can be obtained by selecting one 1-element from the S place of each row of the extension matrix
2) Each column, which is involved in the process of selecting under_T elements, corresponds to a feature. Hence, the number of involved columns is the number of considered features. Furthermore, the minimum cardinality of S is equivalent to the minimum number of involved columns. This completes the proof.

According to Theorem 2, the OFFSS problem can be transformed into a search in $\text{EM}(P, N)$ for a path which involves the least columns. The heuristic search algorithm established in Section IV is based on this transformation. In the following, we prove that the search for a path is NP-hard.

C. OFFSS Problem is NP-Hard

The following theorem gives the computational complexity of selecting an optimal fuzzy-valued feature subset.

Theorem 3: The OFFSS problem described in Definition 5 is NP-hard.

Proof: Noting that "If problem (A) is NP-hard and problem (A) can be reduced into problem (B) within polynomial time, then Problem (B) is also NP-hard," we complete the proof by constructing a transformation which can reduce a known NP-hard problem into the OFFSS problem within polynomial time. By Theorem 2, the OFFSS problem is equivalent to the problem of searching for a path with the least columns, so we only need to reduce a known NP-hard problem into the problem of searching for a path with the least columns. The problem of optimal set cover described below is a known NP-hard problem [17].

Problem of Optimal Set Cover: Let R be a finite set, $U = \{S_1, S_2, \ldots, S_p\}$ be a group of subsets of R. We say U is a cover of R if $\cup_{i=1}^{p} S_i \supset R$. We say U^* is an optimal cover of R if U^* is a cover of R and $|U^*| \leq |U|$ for any arbitrary R's cover U where $|\bullet|$ denotes the cardinality of a set.

Without loss of generality, we explicitly give the process of constructing the transformation (from the problem of optimal set cover to the problem of searching for a path with the least columns in extension matrix) via examples [10] and [6].

Consider a universe of discourse $R = \{1, 2, 3, 4, 5, 6, 7\}$ and a group of R's subsets $S_1 = \{1, 4, 5, 7\}$, $S_2 = \{3, 4, 5\}$, $S_3 = \{2, 7\}$, $S_4 = \{1, 2, 6\}$, $S_5 = \{1, 3, 7\}$, and $S_6 = \{3, 5, 6\}$. It is clear this group of subsets constitutes a cover of R. By arranging these six subsets, Table III can be formed. From Table III, one can find that $\{S_1, S_2, S_4\}$ constitutes an optimal cover of R. Now we replace the six subsets in Table III with their characteristic sets, e.g., replace S_1 with $1/1 + 0/2 + 0/3 + 1/4 + 1/5 + 0/6 + 1/7$. The result of replacement is shown in Table IV. Consequently, searching for an optimal set cover in Table III is equivalent to searching for a group of characteristic sets with the minimum cardinality in Table IV such that there is at least one under_T element in each row of Table IV restricted in these characteristic sets. For example, $\{F_1, F_2, F_4\}$ is such a group of characteristic sets. Therefore, we have given the validity of the conclusion that searching for an optimal set cover in Table III is equivalent to searching for a path involving the least number of columns in Table IV.

The remaining is to show that Table IV can be regarded as an extension matrix of P with respect to N. We regard the six notations $F_1, F_2, \ldots, F_6$ in Table IV as six features and regard each row in Table IV as a negative example denoted by e_j ($1 \leq j \leq 7$), e.g., $e_1 = (1, 0, 0, 1, 1, 0)$. Define the negative example set $N = \{e_1, e_2, \ldots, e_7\}$, the positive example set $P = \{e^+\}$ in which $e^+ = (0, 0, 0, 0, 0, 0)$, the similarity measure $\text{SM}(0, 0) = \text{SM}(1, 1) = 1$, $\text{SM}(1, 0) = 0$, denote the under_T element by 1 and the non under_T element by 0 ($0 < T < 1$), one can directly verify that Table IV is just $\text{EM}(P, N)$, the extension matrix of P with respect to N. The proof is completed.

IV. Heuristic Algorithms for OFFSS

From Theorem 3, one can find that obtaining a practically exact algorithm for the OFFSS problem is unrealistic. So, we have to look for heuristic algorithms. From Theorem 2, we know that the OFFSS problem is equivalent to a search for a path involving the least columns in $\text{EM}(P, N)$ which is the extension matrix of P with respect to N (P is the positive class and N is the negative class). Definition 10 shows that a path in $\text{EM}(P, N)$ means that a connection of under_T elements which are obtained by selecting one under_T element from each row of $\text{EM}(P, N)$. One can expect that, "the bigger N_1 is, the smaller N_2 is" where N_1 denotes the number of under_T elements in each column of a path and N_2 is the number of columns involved in this path. Hence, an intuitive idea of searching a path involving the least columns is to gradually select the column with the most under_T elements in the extension matrix. In detail, one can select one column with the most under_T elements in the current extension matrix and then remove the rows which include an under_T element in the selected column. This process is repeated when the extension matrix is not empty. The result is expected to have a

TABLE III
SET COVER PROBLEM

R	S_1	S_2	S_3	S_4	S_5	S_6
1	1			1	1	
2			2	2		
3		3			3	3
4	4	4				
5	5	5				5
6				6		6
7	7		7		7	

TABLE IV
CHARACTERISTIC SETS OF A COVER

R	F_1	F_2	F_3	F_4	F_5	F_6
1	1	0	0	1	1	0
2	0	0	1	1	0	0
3	0	1	0	0	1	1
4	1	1	0	0	0	0
5	1	1	0	0	0	1
6	0	0	0	1	0	1
7	1	0	1	0	1	0

smaller number of columns (features). The following heuristic algorithm is formed according to this idea. In fact, this is a kind of greedy algorithm.

Heuristic algorithm

Step 1. Initialization: FS is the entire feature space; S is he feature subset to be searched; P is the given positive class; N is the given negative class; $\mathrm{EM}(P,N)$ is the current extension matrix of P with respect to N; and S is initially set to an empty set.

Step 2. From the current extension matrix $\mathrm{EM}(P,N)$, find a column with the most under_T elements. Use F_j to denote this column, and then replace S with $S \cup \{F_j\}$.

Step 3. From $\mathrm{EM}(P,N)$, remove the rows which include an under_T element in the selected j-th column, and then form a new $\mathrm{EM}(P,N)$ which is regarded as the current extension matrix.

Step 4. If $\mathrm{EM}(P,N)$ is empty, then regard S as the final result [stop]; else, go to Step 2.

Example 4 illustrates clearly the computed process of the above heuristic search algorithm.

A. Example 4

Consider the OFFSS problem of the group of examples given in example 1 (Table I). The extension matrix of P with respect to N has been shown in Fig. 4. From Fig. 4, one can find that the second column has the most under_T elements (the threshold value T is set to 0.25). So the current feature subset S is set to be $\{B\}$. After removing the rows which include an under_T element in the second column, the current extension matrix only includes the first row and the last row of the original matrix. Both the first and third columns are two columns with the most under_T elements, hence, the first feature A or the third feature C is aggregated to S. Consequently, two optimal feature subsets, $S_1 = \{A, B\}$ and $S_2 = \{B, C\}$, are obtained. Intuitively, the feature subset S_1 is better than S_2 due to $r_{11} = 0$ and $r_{13} = 0.25$ in the extension matrix (Fig. 4).

Essentially, the OFFSS problem proposed in this paper is to search for such significant features that the overlapping degree of P and N does not exceed a given threshold. From Definition 4, one can see that the "maximum" degree of overlapping is used. The maximum operation may result in inflexibility of the heuristic algorithm to some extent. Moreover, the Step 4 in the above algorithm does not allow noisy example appearing in P and N where the noisy example refers to such an example which appears simultaneously in P and in N. To illustrate the inflexible case and the noisy case, we consider the following two examples.

B. Example 5

Consider the examples given in Table V where two qualifiers "Very" and "More-or-less" are defined as

$$\text{Very}\,(A(x)) = (A(x))^2 \text{ and More-or-less}\,(A(x)) = \sqrt{A(x)}$$

for any term with membership function $A(x)$. The similarity measure between two terms A and B is defined as the equation shown at the bottom of the page. One can directly compute the extension matrix of P with respect to N, which is shown in Fig. 5.

By setting $T = 0.40$, it is easy to see from Fig. 5 that, except for the last column, each column of the extension matrix has six under_T elements. The maximal number of under_T elements is reached at three columns simultaneously. The aforementioned heuristic algorithm does not know which column should be selected.

C. Example 6

Consider the examples given in Table VI where only one positive example exists. The last negative example which is identical to the positive example is possibly noisy. The extension matrix is shown in Fig. 6. According to the above heuristic algorithm, the selection process of feature subset is described as Empty $\rightarrow \{A\} \rightarrow \{A, B\}$. The remaining extension matrix

$$\mathrm{SM}(A,B) = \begin{cases} 0.85, & \text{if } B = \text{Very}(A) \text{ or } B = \text{More-or-less}(A) \\ \mathrm{Max}_{x \in X}(\min(A(x), B(x)))^2, & \text{otherwise} \end{cases}.$$

TABLE V
GROUP OF EXAMPLES WITH QUALIFIERS

Case.	Feature A	Feature B	Feature C	Feature D	Class
1	Mid	Small	Small	Mid	Positive
2	Very Big	Mid	More-or-less Big	More-or-less Small	Positive
3	Very Small	Big	Mid	Mid	Positive
4	Small	Small	Small	Big	Negative
5	Big	Big	Mid	Mid	Negative
6	Mid	Mid	Big	Mid	Negative

$$\begin{bmatrix} 0.25 & 1.00 & 1.00 & 0.25 \\ 0.25 & 0.00 & 0.25 & 1.00 \\ 1.00 & 0.25 & 0.00 & 1.00 \\ 0.00 & 0.25 & 0.00 & 0.00 \\ 0.85 & 0.25 & 0.38 & 0.38 \\ 0.15 & 1.00 & 0.85 & 0.38 \\ 0.85 & 0.00 & 0.25 & 0.25 \\ 0.00 & 1.00 & 1.00 & 1.00 \\ 0.15 & 0.25 & 0.25 & 1.00 \end{bmatrix}_{9\times 4}$$

Fig. 5. Extension matrix $\mathrm{EM}(P, N)$ corresponding to Table V.

$\mathrm{EM}(P, N)$ includes only the last row in which there exists no under_T elements (the threshold value T does not exceed 1), so $\mathrm{EM}(P, N)$ cannot become empty.

To overcome the shortcomings as shown in Examples 5 and 6, we revise the previous heuristic algorithm as shown here.

Revised heuristic algorithm

Step 1. Initialization is same as the original heuristic algorithm.

Step 2. From the current extension matrix $\mathrm{EM}(P, N)$, find a column with the most under_T elements. Use F_j to denote this column, and then replace S with $S \cup \{F_j\}$. If there is more than one column with the most under_T elements, select one column such that the sum of its under_T elements is minimum.

Step 3. Same as the original heuristic algorithm.

Step 4. If the number of under_T elements of the remaining extension matrix $\mathrm{EM}(P, N)$ is less than a given small number (threshold value), then regard S as the final result and regard the remaining examples as noise [stop]; else go to Step 2.

By using the revised heuristic algorithm to handle the above Examples 4, 5, and 6, one can obtain the results 1) in Example 4, $S_1 = \{A, B\}$ is the first optimal feature subset and $S_2 = \{B, C\}$ is the second one; 2) in Example 5, $S_1 = \{A, B\}$ is the first optimal feature subset and $S_2 = \{A, C\}$ is the second one; and 3) in Example 6, $S_1 = \{A, B\}$ is the only optimal feature subset and the last negative example is regarded as noise.

TABLE VI
GROUP OF EXAMPLES WITH NOISE

Case.	Feature A	Feature B	Feature C	Feature D	Class
1	Mid	Mid	Mid	Mid	Positive
2	Big	Mid	Mid	Mid	Negative
3	Mid	Big	Mid	Mid	Negative
4	Big	Mid	Big	Mid	Negative
5	Big	Mid	Mid	Big	Negative
6	Mid	Mid	Mid	Mid	Negative

$$\begin{bmatrix} 0.25 & 1.00 & 1.00 & 1.00 \\ 1.00 & 0.25 & 1.00 & 1.00 \\ 0.25 & 1.00 & 0.25 & 1.00 \\ 0.25 & 1.00 & 1.00 & 0.25 \\ 1.00 & 1.00 & 1.00 & 1.00 \end{bmatrix}_{5\times 4}$$

Fig. 6. Extension matrix $\mathrm{EM}(P, N)$ corresponding to Table VI.

It is worth noting that, in the process of implementation of the revised heuristic algorithm, the extension matrix $\mathrm{EM}(P, N)$ does not need to be really generated in memory and only the number of under_T elements needs to be aggregated. It shows that the algorithm has no much computational effort and space consumption that implies the implementation is easy and cheap. Another benefit may be that the proposed heuristic algorithm does not like GA and is not time consuming. In addition, one point needed to be shown is that the OFFSS problem described in this paper will degenerate to the crisp case proposed in [6] if all features are restricted to nominal values.

V. EXPERIMENTS AND COMPARISONS

In this section, we investigate applications of OFFSS to fuzzy decision tree induction, and compare the performance of OFFSS with three selected feature selection methods by experiments.

A. Selected Three Feature Selection Methods

We select three types of feature selection methods in comparison with our OFFSS. The three are neuro-fuzzy method, neural network output sensitivity-based method, and mutual information-based method, respectively.

Neuro-fuzzy approaches [e.g., [20], [21], [38], and [7]] are usually based on an overall feature evaluation index (OFEI). Each class is considered as a fuzzy subset. In this paper we select the [7] definition on OFEI which is given for the qth feature by

$$\mathrm{OFEI}q = \frac{\sum_{j,k=1, j\neq k}^{Q} H_{qjk}}{\sum_{j=1}^{Q} H_{qj}}$$

where Q is the number of classes, H_{qj} is the value of classification entropy of the qth feature with respect to the jth class, and H_{qjk} is the value with respect to the j-th and the kth classes. It is easy to see that the lower the value of OFEI, the better the feature is.

Neural network output sensitivity-based approaches use a feature quality index (FQI) for each feature q and then the

features can be sorted according to FQI_q. After training a feed-forward neural network, the FQI for the qth feature refers usually to the value of output' sensitivity to the q-th feature perturbation. The method to evaluate the value of FQI can be different. For example, Zurada in [42] and Engelbrecht in [8] used partial derivatives of the output with respect to the input to define the sensitivity measure and compute its value by Taylor approximate expansion. Yeung in [39] used the variance of the output error with respect to the input perturbation to define the sensitivity measure and Zeng in [41] used the expected value of output error with respect to the input change. De in [7] defined the FQI as follows. For each training data point x_i, the q-th component is set to zero. If $x_i^{(q)}$ denotes the modified point, then except for the qth component the other components of x_i and $x_i^{(q)}$ are the same. Let o_i and $o_i^{(q)}$ denote the output vectors obtained from the neural network with respect to x_i and $x_i^{(q)}$, respectively. If the qth feature is not salient, the difference between o_i and $o_i^{(q)}$ should be small. Therefore, the FQI is defined as

$$FQI_q = \sum \left\| o_i - o_i^{(q)} \right\|^2.$$

The important feature should correspond to big FQI. This paper selects the aforementioned De [7] method to compute the FQI for comparison.

Mutual information-based feature selector [2] and [22] is universally accepted as a promising method to feature selection. This method considers mainly the dependence between features and selects some features with high independence. It can be briefly formulated as follows.

1) Let F denote the initial set of n features and S be empty.
2) Compute the mutual information $I(C,f)$, for each feature $f \in F$.
3) Find the feature f that maximizes $I(C,f)$, set $F \leftarrow F - \{f\}$, $S \leftarrow \{f\}$.
4) Repeat until $|S| = k$: a) For all pairs (f,s), $f \in F$, $s \in S$, compute $I(f,s)$. b) Choose feature f as the one that maximizes $I(c,f) - \beta \sum_{s \in S} I(f,s)$, set $F \leftarrow F - \{f\}$, $S \leftarrow S \cup \{f\}$.
5) Output the set S containing the selected features.

In our experiments, the parameter β in step 4) is assumed 0.5.

B. Databases Used

We select five databases for comparing our OFFSS with the approaches mentioned in Section V.A. The five databases, i.e., Iris [34], MPG [34], Pima [34], Breast cancer [34], and Sleep state [23], are briefly summarized here.

1) Iris dataset: This is a well-known benchmark dataset which is widely used to test a learning algorithm in the field of machine learning. This dataset has 150 examples which are classified into three classes, i.e., Setosa, Versicolor and Virginical. Each example is characterized by four numerical features which are sepal length (SL), sepal width (SW), petal length (PL), and petal width (PW). Five linguistic terms, i.e., very small (VSM), small (SM), medium (MED), large (LRG) and very large (VLRG), are used to fuzzify every feature.
2) Mile per gallon (MPG) dataset: This is another bench-mark dataset which comes from a nonlinear regression model where several features (input variables) are used to predict another feature (output variable). The MPG problem has six input variables which are no. of cylinders (discrete), Displacement (continuous), Horsepower (continuous), Weight (continuous), Acceleration (continuous) and Year-model (discrete). The output variable is the fuel consumption in MPG. After removing examples with missing values, the data set is reduced to 392 entries. One purpose of the research on this problem is to select several important input variables (to find the degree of importance of inputs with respect to the output).
3) Pima diabetes dataset: The Pima Indian Diabetes dataset contains 768 examples. Each example representing a patient who may show signs of diabetes is described by eight features which are: a) number of times pregnant, b) plasma glucose concentration, c) diastolic blood pressure, d) triceps skin fold thickness, e) two-hour serum insulin, f) body mass index, g) diabetes pedigree function, and h) age. There are 500 examples from patients who do not have diabetes and 268 examples from patients who are known to have diabetes.
4) Breast cancer diagnosis problem: The University of Wisconsin Breast Cancer data set consists of 699 patterns which are classified two classes, 458 benign examples and 241 malignant examples. Each example is described by nine features: a) clump thickness, b) uniformity of cell size, c) uniformity of cell shape, d) marginal adhesion, e) single epithelial cell size, f) bare nuclei, g) bland chromatin, h) normal nucleoli, and i) mitoses. In the dataset, the values of the sixth feature of 16 examples are missing. We neglect the 16 examples when conducting experiments.
5) Sleep state dataset: This dataset describes different states about human's sleep contains 1236 examples with eleven attributes and is initially classified to six classes.

C. Feature Subset Selection

We first use both our proposed OFFSS and the three methods mentioned in Section V-A to select feature subsets based on the each dataset. Since our OFFSS is with respect to two-class problem, we need to transfer more than two class problems to two-class problems. We illustrate this transformation via the Iris dataset. The similarity measure between two linguistic terms is defined as

$$\mathrm{SM}(A,B) = \mathrm{Max}_{x \in X} \left(\min(A(x), B(x))\right)^2$$

for A, $B \in \{\mathrm{VSM}, \mathrm{SM}, \mathrm{MED}, \mathrm{LRG}, \mathrm{VLRG}\}$. The computed process and the result for given threshold $T = 0.25$ are listed as follows.

Step 1. $Setosa \cup Versicolor \Rightarrow P$, $Virginica \Rightarrow N$. The process of feature selection is
Empty_set $\rightarrow \{PW\} \rightarrow \{\mathrm{PW}, \mathrm{PL}\}$
Step 2. Setosa $\Rightarrow P$, Versicolor $\Rightarrow N$. The process of feature selection is
Empty_set $\rightarrow \{\mathrm{PL}\}$.
Step 3. The optimal feature subset for the classification task is determined to be $S = \{\mathrm{PL}, \mathrm{PW}\}$.

For MPG dataset, since the output of this problem is continuous, a discretization should be done. Output values are roughly categorized into three classes in this paper, that is, Class 1: $\{\mathrm{MPG} \leq 18\}$, Class 2: $\{18 < \mathrm{MPG} \leq 30\}$, and Class 3: $\{30 < \mathrm{MPG}\}$. The definition of similarity measure between two fuzzy numbers are the same as that in the above Iris classification problem. For given threshold $T = 0.25$, the computed process and the result are listed here.

Step 1. Class_1 $\cup$ Class_2 $\Rightarrow P$, Class_3 $\Rightarrow N$. The process of feature selection is shown as:

Empty_set
$\rightarrow$ {Year_model}
$\rightarrow$ {Year_model, Displacement}
$\rightarrow$ {Year_model, Displacement, Acceleration}
$\rightarrow$ {Year_model, Displacement, Acceleration, Weight}

Step 2. Class_1 $\Rightarrow P$, Class_2 $\Rightarrow N$. The process of feature selection is shown as:

Empty_set
$\rightarrow$ {Displacement}
$\rightarrow$ {Diplacement, Year_model, }
$\rightarrow$ {Diplacement, Year_mode, Acceleration}
$\rightarrow$ {Displacement, Year_model, Acceleration, Weight}

Step 3. The optimal feature subset for the classification task is determined to be S = {Year_model, Displacement, Acceleration, Weight}.

Moreover, for the Sleep state dataset, the third class (688 examples) is considered as the negative class and the other five classes are regarded as the positive class. The feature subset selection results given by difference approaches are listed in Table VIII where the number of features is given in advance.

D. Application to Fuzzy Decision Tree Induction

Each selected dataset is first split into two parts, the training and testing sets by randomly choosing examples. For all datasets 80% of the examples are randomly selected as the training set and the remainder as the testing set. We would like to use fuzzy decision tree induction to check the performance difference between before and after feature selection, and to compare the performance of our proposed OFFSS with the other three approaches mentioned in Section V.A.

TABLE VII
GROUP OF EXAMPLES WITH FIVE FUZZY-VALUED FEATURES

Case.	Feature A	Feature B	Feature C	Feature D	Feature E	Class
1	VLRG	LRG	LRG	LRG	LGR	P
2	LRG	LRG	LRG	LRG	LRG	P
3	MED	SM	MED	LRG	MED	P
4	VLRG	MED	LRG	LRG	LRG	P
5	LRG	SM	SM	LRG	LRG	P
6	SM	SM	SM	LRG	SM	N
7	VLRG	LRG	VSM	MED	MED	N
8	MED	SM	SM	LRG	SM	N
9	SM	SM	MED	MED	MED	N
10	SM	MED	LRG	MED	MED	N

TABLE VIII
FEATURE SUBSET SELECTION BY DIFFERENT APPROACHES

	Iris	MPG	Pima	Breast Cancer	Sleep state
OFEI	{4, 3}	{4, 5, 6, 2}	{2, 3, 6}	{6, 1, 3, 2}	{9, 6, 5, 10, 8, 1}
FQI	{4, 3}	{4, 6, 3, 2}	{8, 2, 1}	{6, 1, 8, 3}	{5, 6, 9, 8, 3, 10}
MIFS	{4, 3}	{4, 6, 2, 1}	{2, 6, 8}	{6, 3, 2, 7}	{5, 1, 9, 10, 8, 3}
OFFSS	{4, 3}	{6, 2, 5, 4}	{2, 6, 7}	{6, 3, 1, 2}	{5, 7, 4, 10, 1, 9}

Fuzzy decision tree induction is an important way of learning from fuzzy examples. Since the generation of optimal (fuzzy) decision tree has been proved to be NP-hard problem(s) [12], [36], it is unrealistic to find an exact algorithm for the optimal tree. It forces people to generate relatively better decision trees via using heuristic information. One popular and powerful heuristic for generating crisp decision trees is called ID3. The earlier version of ID3, which is based on minimum information entropy to select expanded attributes, was proposed by Quinlan [25] in 1986. Fuzzy ID3 is a fuzzy version of crisp ID3, which has been suggested by some authors [37], [31], [14]. The fuzzy decision tree induction based on ID3 can be summarized as follows.

1) Fuzzifying the training data: For each feature of each dataset, we use five linguistic terms which are VSM, SM, MED, LRG, and VLRG (Fig. 7). It is worth noting that, for any x (a real value of any feature), one linguistic term F ($F \in \{\mathrm{VSM}, \mathrm{SM}, \mathrm{MED}, \mathrm{LRG}, \mathrm{VLRG}\}$) can be selected such that $F(x) \geq 0.5$. One may argue the number of used linguistic terms. In fact one can conduct the experiments using different number of linguistic terms (say, SM, MED, and LRG) and obtain a similar result. This paper aims to investigate the feature selection and does not investigate in detail the used linguistic terms.
2) Training: Based on the training dataset, we can generate a fuzzy decision tree. (Of course, one can use Quinlan's SEE5 software to generate a crisp decision tree directly from the continuous training data, but in this paper we generate a fuzzy decision tree in terms of our implemented fuzzy ID3 algorithm). By changing each path from root to leaf to a fuzzy production rule, we can get a set of fuzzy rules. The training will be conducted on the training dataset both before and after the feature selection.
3) Testing: Based on a specified fuzzy reasoning mechanism, we can test the generated fuzzy rules. Here, the reasoning mechanism is specified to be max–min operations. For details, one can refer to [40]. The testing will be conducted on training/testing datasets both before and after the feature selection.

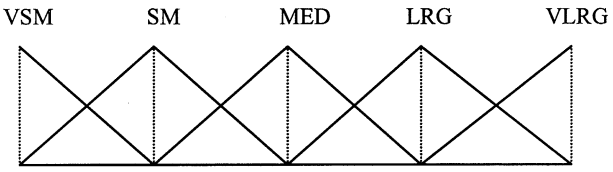

Fig. 7. Five membership functions.

We repeat steps 1)–3) ten times and record the average for training and testing with respect to the selected five datasets. The result (before and after feature selection) is listed in Table IX.

E. Comparison and Analysis

For Iris dataset, all methods obtain the same result. That is, for the classification task, the two important attributes (features) are PL and PW. Only these two features are used, several fuzzy rules with satisfactory accuracy can be extracted.

For the MPG problem, some researchers had investigated the input selection. For example, in [15], the author presented a quick and straightforward way of input selection for neuro-fuzzy modeling and tested his algorithm using MPG problem. After the complicated computation and analysis, the author gave a result that, for the continuous output, the attributes Weight and Year-model are the two most important input variables.

One problem appearing in the above selection process is whether the feature entering the feature subset earlier is more important than the feature entering later. We do not think so. The heuristic does not provide such information. Our obtained result has a few differences in comparison with the results in [15]. It is due to the fact that the MPG problem was discussed in [15] as continuous output but our discussion about this problem is based on a roughly discrete classification of outputs. It is easy to see that our selected optimal feature subset includes the two features Weight and Year-model which are considered as the two most important input variables by many investigators. However, the information offered only by these two features for our roughly discrete classification are not enough to separate the three classes, although they are the two most important input variables of this problem.

From Table VIII, one can see that the feature subsets selected by using the four methods are different (except for the Iris dataset).

From Table IX, we can obtain the following comparative results. After feature selection, our proposed OFFSS is slightly superior to the other three methods in terms of testing accuracy. It may be due to the fact that the OFFSS can select the approximately optimal feature subset. Although the four methods has no obvious difference in training accuracy and for each method the selected features have the almost same performance as all features, the computational complexity of selection of OFFSS is much less than the other three.

F. Some Notes

Usually, fuzzy ID3 algorithm uses partial features of feature space which is enough to complete the generation of decision trees. Generally speaking, the fewer features to be used in the algorithm are, the better the generated decision trees. Therefore, we expect that the decision tree generated by using optimal feature subset selected by the heuristic algorithm proposed in this paper is superior to the decision tree without optimal feature subset. Example 7 shows that it is possible to generate a relatively better decision tree after carrying out the feature subset selection.

TABLE IX
PERFORMANCE COMPARISON AMONG DIFFERENT APPROACHES

Method	Training/testing	Iris	MPG	Pima	Breast Cancer	Sleep state
	All features: Training	0.9659	0.8266	0.7519	0.9884	0.9526
	Testing	0.9701	0.6974	0.7239	0.9222	0.9123
OFEI	Selected features: Training	0.9625	0.754	0.7423	0.9879	0.9356
	Testing	0.9533	0.7076	0.7423	0.9279	0.9239
FQI	Selected features: Training	0.9625	0.7459	0.7216	0.9950	0.9443
	Testing	0.9533	0.7241	0.7267	0.9277	0.9287
MIFS	Selected features: Training	0.9625	0.8119	0.7447	0.9878	0.9401
	Testing	0.9533	0.7279	0.7476	0.9172	0.9312
OFFSS	Selected features: Training	0.9625	0.7928	0.7427	0.9885	0.9428
	Testing	0.9533	0.7559	0.7561	0.9358	0.9532

Example 7: Consider Table VII (adopted from [6] with fuzzification) where the membership functions of the five terms {VSM, SM, MED, LGR, and VLGR} are shown in Fig. 7. The similarity measure between two terms is defined in Section II. Using our revised heuristic algorithm proposed in Section IV for the given threshold $T = 0.25$, one can obtain the optimal feature subset $\{D, E\}$. It is easy to verify that there are three optimal feature subsets, they are $\{A, C\}$, $\{D, E\}$ and $\{A, E\}$. Using fuzzy ID3 algorithm on the entire feature space $\{A, B, C, D, E\}$ and the selected feature subset $\{D, E\}$, one can obtain two decision trees as shown in Figs. 8 and 9. From the view point of minimum number of leaves, one can see that the decision tree with OFFSS (Fig. 9) is superior the decision tree without OFFSS (Fig. 8).

From Example 7, we know that our heuristic algorithm for OFFSS does not find all optimal feature subsets. The group of examples shown in Table VII has three total optimal feature subsets for the given threshold $T = 0.25$ ($\{A, C\}$, $\{D, E\}$ and $\{A, E\}$). Our heuristic only finds the second one but fuzzy ID3 algorithm on the entire feature space uses the third one. A problem is whether or not the expanded features in fuzzy ID3 always constitute an optimal feature subset. Example 8 gives us a negative answer.

Example 8: Consider Table V where the membership functions of the three terms {Small, Mid, and Big} are shown in Fig. 1. The similarity measure between two terms is defined in Example 5. It is easy to check that, for the given threshold $T = 0.25$, $\{A, B\}$ and $\{A, C\}$ are the only two optimal feature subset. However, the expanded attribute at the root is selected as the attribute D which has not been included in any optimal feature subset. The attribute selection of fuzzy ID3 at the root is based on the following computation of fuzzy entropy:

$$\begin{aligned}\text{Entropy}(A) &= 0.690\\ \text{Entropy}(B) &= \text{Entropy}(C) = 0.693\\ \text{Entropy}(D) &= 0.638.\end{aligned}$$

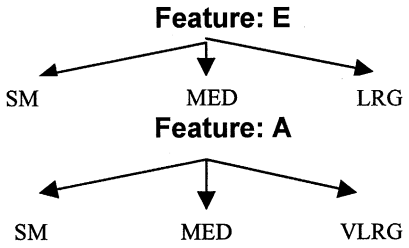

Fig. 8. Fuzzy decision tree on feature space $\{A, B, C, D, E\}$.

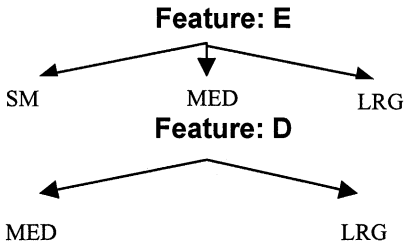

Fig. 9. Fuzzy decision tree on feature subset $\{D, E\}$.

Many researchers had pointed out that the overloaded number of features would seriously affect the quality of inductive learning and the accuracy on extracted rules and irrelevant features would enlarge the noise of the training set [5], [16]. Therefore, we expect that the number of features used in the learning phase can be reduced via optimal feature subset selection such that the performance of learning can be improved. The testing on hand-written number recognition reported in [6] verified this expectation under the crisp environment.

VI. Conclusion

This paper investigates a problem of OFFSS. The computational complexity of OFFSS is proved to be NP-hard; OFFSS is shown to be equivalent to a search for a path in fuzzy extension matrix; a heuristic algorithm for OFFSS is given; and the feasibility and simplicity of the proposed algorithm are demonstrated by applications of OFFSS to fuzzy decision tree induction and by comparison with three different feature selection techniques developed recently.

References

[1] P. W. Baim, "A method for attribute selection in inductive learning systems," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, pp. 88–896, June 1988.

[2] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Trans. Neural Networks*, vol. 5, pp. 537–550, July 1994.

[3] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Art. Intell.*, vol. 97, no. 1-2, pp. 245–271, Dec. 1997.

[4] F. Z. Bril, D. E. Brown, and N. W. Worthy, "Fast genetic selection of features for neural network classifiers," *IEEE Trans. Neural Networks*, vol. 3, pp. 324–328, Mar. 1992.

[5] R. Caruana and D. Fratage, "Greedy attribute selection ," in *Machine Learning: Proc. 11th Int. Conf.*, San Francisco, CA, 1994, pp. 283–288.

[6] B. Chen and J. R. Hong, "The problem of finding optimal subset of features," *Chinese J. Comput.*, vol. 20, no. 2, pp. 133–138, 1997.

[7] R. K. De, N. R. Pal, and S. K. Pal, "Feature analysis: Neural network and fuzzy set theoretic approaches," *Pattern Recogn.*, vol. 30, no. 10, pp. 1579–1579, 1997.

[8] A. P. Engelbrecht, "A new pruning heuristic based on variance analysis of sensitivity information," *IEEE Trans. Neural Networks*, vol. 12, pp. 1386–1399, Nov. 2001.

[9] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. New York: Academic, 1972.

[10] J. R. Hong, "AE1: An extension matrix approximate method for the general covering problem," *Comput. Inform. Sci.*, vol. 14, no. 6, pp. 421–437, 1985.

[11] ——, "Extension matrix theory of learning from examples," *Chinese J. Comput.*, vol. 14, no. 6, pp. 401–410, 1991.

[12] L. Hyafil and R. L. Rivest, "Constructing optimal binary decision trees is NP-complete," *Inform. Processing Lett.*, vol. 5, no. 1, pp. 15–17, 1976.

[13] L. K. Hyung, Y. S. Song, and K. M. Lee, "Similarity measure between sets and between elements," *Fuzzy Sets Syst.*, vol. 62, pp. 291–293, 1994.

[14] H. Ichihashi, T. Shirai, K. Nagasaka, and T. Miyoshi, "Neuro-fuzzy ID3," *Fuzzy Sets Syst.*, vol. 81, pp. 157–167, 1996.

[15] J.-S. R. Jang, "Input selection for ANFIS learning," *Proc. IEEE Int. Conf. Fuzzy Systems*, vol. 2, pp. 1493–1499, 1996.

[16] G. H. John, R. Kohavi, and K. Pfloger, "Irrelevant features and the subset selection problem," in *Machine Learning: Proc. 11th Int. Conf.*, San Francisco, CA, 1994, pp. 121–129.

[17] D. S. Johnson, "Approximation algorithms for combinational problems," *Comput. Syst. Sci.*, vol. 9, no. 3, pp. 38–48, 1973.

[18] I. T. Joliffe, *Principal Component Analysis*. New York: Springer-Verlag, 1986.

[19] C. W. D. Justin and R. J. Victor, "Feature subset selection with a simulated annealing data mining algorithm," *J. Intell. Inform. Syst.*, vol. 9, pp. 57–81, 1997.

[20] R. Krishnapuram and J. Lee, "Fuzzy connective-based hierarchical aggregation networks for decision making," *Fuzzy Sets Syst.*, vol. 46, no. 1, pp. 11–27, 1992.

[21] R. Krishnapuram and F. C.-H. Rhee, "Fuzzy rule generation method for high-level computer vision," *Fuzzy Sets Syst.*, vol. 60, pp. 245–258, 1993.

[22] N. Kwak and C.-H. Choi, "Input feature selection for classification problems," *IEEE Trans. Neural Networks*, vol. 13, pp. 143–159, Jan. 2002.

[23] R. S. Michalski, I. Mozetic, and J. R. Hong, "The multipurpose incremental learning systems," in *Proc. 5th National Conf. Artificial Intelligence*, M. Revist, Ed., Philaddelphia, PA, 1986, pp. 1041–1045.

[24] C. P. Pappis and N. I. Karacapilidis, "A comparative assessment of measures of similarity of fuzzy sets," *Fuzzy Sets Syst.*, vol. 56, pp. 171–174, 1993.

[25] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, pp. 81–106, 1986.

[26] T. W. Rauber and A. S. Steiger-Gargao, "Feature selection of categorical attributes based on contingency table analysis," presented at the 5th Portuguese Conf. Pattern Recognition, Porto, Portugal, 1993.

[27] M. L. Raymer *et al.*, "Dimensionality reduction using genetic algorithms raymer," *IEEE Trans. Evol. Comput.*, vol. 4, pp. 164–171, July 2000.

[28] M. R. Rezaee *et al.*, "Fuzzy feature selection," *Pattern Recogn.*, vol. 32, pp. 2011–2019, 1999.

[29] R. Setiono and H. Liu, "Neural-network feature selector," *IEEE Trans. Neural Networks*, vol. 8, pp. 654–662, May 1997.

[30] M. Setnes and U. Kaymak, "Fuzzy target selection in direct marketing," *Proc. IEEE/IAFE/INFORMS Conf. Computational Intelligence Financial Engineering (CIFEr)*, pp. 92–97, Mar. 1998.

[31] T. Tani and M. Sakoda, "Fuzzy modeling by ID3 algorithm and its application to prediction of outlet temperature," *Proc. IEEE Int. Conf. Fuzzy Systems*, pp. 923–930, Mar. 1992.

[32] M. Setnes and U. Kaymak, "Fuzzy modeling of client preference from large data sets: An application to target selection in direct marketing," *IEEE Trans. Fuzzy Syst.*, vol. 9, pp. 153–163, Feb. 2001.

[33] I. B. Turksen and Z. Zhong, "An approximate analogical reasoning approach based on similarity measures," *IEEE Trans. Syst., Man, Cybern.*, vol. 18, pp. 1049–1056, Nov./Dec. 1988.

[34] UCI Repository of Machine Learning Databases and Domain Theories. [Online]ftp://ftp.ics.uci.edu/pub/machine-learning-databases/

[35] W.-J. Wang, "New similarity measures on fuzzy sets and on elements," *Fuzzy Sets Syst.*, vol. 85, pp. 305–309, 1997.

[36] X. Z. Wang, B. Chen, G. L. Qian, and F. Ye, "On the optimization of fuzzy decision trees fuzzy sets and systems," *Fuzzy Sets Syst.*, vol. 112, no. 2, pp. 117–125, 2000.

[37] R. Weber, "Fuzzy-ID3: A class of methods for automatic knowledge acquisition," in *Proc. 2nd Int. Conf. Fuzzy Logic Neural Networks*, Iizuka, Japan, July 17–22, 1992, pp. 265–268.

[38] R. R. Yager and L. A. Zadeh, *Fuzzy Sets, Neural Networks, and Soft-computing*. New York: Van Nostrand-Reinhold, 1994.

[39] D. Yeung and X. Sun, "Using function approximation for sensitivity analysis of MLP," *IEEE Trans. Neural Networks*, vol. 13, pp. 34–44, Jan. 2002.
[40] Y. Yuan and M. J. Shaw, "Induction of fuzzy decision trees," *Fuzzy Sets Syst.*, vol. 69, pp. 125–139, 1995.
[41] X. Zeng and D. Yeung, "Sensitivity analysis of multilayer perceptron to input and weight perturbations," *IEEE Trans. Neural Networks*, vol. 12, pp. 1358–1366, Nov. 2001.
[42] J. M. Zurada, A. Malinowski, and S. Usui, "Perturbation method for deleting redundant inputs of perceptron networks," *Neurocomputing*, vol. 14, pp. 177–193, 1997.
[43] R. Zwick, E. Carlstein, and D. V. Budescu, "Measurement of similarity between fuzzy concepts: A comparative analysis," *Approx. Reason.*, vol. 1, pp. 221–241, 1987.

E. C. C. Tsang, photograph and biography not available at the time of publication.

D. S. Yeung, photograph and biography not available at the time of publication.

X. Z. Wang, photograph and biography not available at the time of publication.