Arab J Sci Eng
DOI 10.1007/s13369-015-1800-0

RESEARCH ARTICLE - COMPUTER ENGINEERING AND COMPUTER SCIENCE

FUDT: A Fuzzy Uncertain Decision Tree Algorithm for Classification of Uncertain Data

S. Meenakshi[1] · **V. Venkatachalam**[2]

Received: 4 December 2014 / Accepted: 23 July 2015

Abstract The classifications of uncertain data turned into one of the dreary procedures in the data mining domain. The uncertain data have tuples with distinctive probability distribution, which helps to find similar class of tuples. When we consider an uncertain data, the feature vector will not be a single valued but a function. In this paper, we proposed fuzzy entropy and similarity measure to characterize the uncertain data through binary decision tree algorithm. Fuzzy entropy is used to find the best split point for the decision tree to handle the uncertain data. Similarity measure is used to make the better decision for the uncertain data with high accuracy. Initially, fuzzy entropy for each feature vector is calculated to select the best feature vector. Then, best split is selected from the selected feature vector. With the help of trained uncertain data, the binary tree starts to grow. Once the split point is selected, then the constructed decision tree is evaluated by the testing phase of uncertain data. The testing data are subjected to the trained decision tree to obtain the classified data. The experimental analyses are made to evaluate the performance of the proposed FUDT approach. Proposed FUDT algorithm is compared with the existing classification algorithm UDT in terms of accuracy and running time. The experimental analysis finalizes that our FUDT algorithm outperforms the existing UDT algorithm.

Keywords Uncertain data · Classification of uncertain data · Decision tree · Binary decision tree · Fuzzy entropy

✉ S. Meenakshi
meenakshi0976@gmail.com

1 Department of Master of Computer Applications, The Kavery Engineering College, Mecheri, Salem 636 453, India

2 The Kavery Engineering College, Mecheri, Salem, India

Published online: 06 August 2015

1 Introduction

In recent years, there is a wave of interest in creating methods for maintaining and mining uncertain data. The main reason for uncertain data in many applications is due to the limitation in equipment such as unreliable sensors, use of imputation, interpolation or extrapolation techniques to estimate position of moving objects, partial or uncertain responses in surveys and many more. Recent work on uncertain data mining consists of probability theory that has often adopted as a formal framework for representing data uncertainty. Usually, an object is represented as a probability density function (pdf) over the attribute space rather than a single point, as usually assumed when uncertainty is neglected. Mining technique for such type of data includes clustering algorithm, density estimation technique, outlier detection, support vector classification and decision trees. [1–6].

Decision tree is a simple, still broadly used method for classification and prediction modeling. It divides data into terminal nodes, where each terminal node appoints a class label. The non-terminal node in decision tree includes root and other internal node, which contains attribute test condition, in order to separate records with different characteristics. The partitioning process stops when the subsets cannot be divided any more by using the predefined criteria. The decision tree is used in several domains such as in database marketing, and also it can be used to segment groups of customer and develop customer profiles to help marketers produce targeted promotion that achieve high response rates [7–9]. Classification over uncertain data has much attention due to their inherent uncertainties of data in many real-world applications such as sensor network monitoring and object identification. Several factors, which generate uncertainty, include data collection error, measurement, data sampling error, absolute source, network latency and transmission

error. For instance, in a moving object databases, because of the limited resources, it is impossible for database server to know about exact positions of all objects at every instance of time. There are two types of uncertainty: measurement or sampling error. The measurement error is obtained from fuzziness of GPS device where as in the sampling error uncertainty obtains from the frequency of moving objects. Hence, it is very important to analyze uncertain data effectively and efficiently [10–13].

The classical methods of classification mainly depend on probability theory framework. This type of technique calculates the probability assignments of objects in different specific classes. The final classification of an object is mainly determined by the class committed with the highest probability value. In the classification of uncertain data, different classes can partly overlap, and objects in the overlapped regions are hard to correctly classify into particular class due to insufficient attributes information. Hence, probability theory framework is usually not the best technique to characterize such uncertainty and imprecision. Recently, new technique such as belief function (BF) is introduced in Dempster–Shafer theory (DST), which is widely used to model the uncertain information for data clustering, data classification, image processing and information fusion [14–17]. Since standard classification techniques are not suitable to handle the imperfection in data, a common way to deal with this is to ignore such data. This leads to loss of information, and the model obtained is not a faithful representation of reality. The study of naive possibility classifiers is motivated by the simplicity and the performances of Naïve Bayesian classifiers (NBC) and by making use of possibility theory to handle poor data. In spite of the fact that possibility distribution is useful for representing imperfect knowledge, there are only few works that use possibility theory for classification and most of existing Naïve classifiers deals with categorical attributes only [18–24].

In this paper, we present an approach for classifying the uncertain data through binary decision tree. To make the decision tree effective, here, we propose a method as fuzzy entropy that helps to select the best split point; also, we present an effective significant measure, which helps the decision tree to work effectively. Initially, we separate the uncertain decision tree into two main parts, the first one is training phase and the second one is testing phase. The training part of the uncertain data helps to construct the decision tree with the aid of fuzzy entropy value. To select the best split point, here we select the best feature vector from the fuzzy entropy value, from which, we select the best split point. The length and accuracy of the decision tree are defined by the proposed significant measure. If significant measure of the every data points in the leaf node belongs to a single class, then there is no need to increase the length of the decision tree, else, the selection of best split point through fuzzy entropy is continued with the number of data points present in the leaf node (partition). Once the construction of binary decision tree for the uncertain data is finished, the constructed binary is evaluated through the testing phase of the uncertain data.

The rest of the paper is organized as follows: the second section describes about recent researches on uncertain data. The third section plots motivation behind the approach, and the fourth section plots the basic algorithms used in the proposed approach. The fifth and sixth sections include the detailed description about the proposed fuzzy uncertain decision tree algorithm for classification of uncertain data. In the seventh, we plot the performance and comparative analysis, and with section eighth, we conclude the paper.

2 Literature Review

Qin et al. [7] have improved the traditional decision tree algorithms and extend measures, including entropy and information gain, considering the uncertain data interval and probability distribution function. Both certain and uncertain datasets can be handled by their algorithm. The experiments have established the effectiveness and robustness of the proposed algorithm as well as its satisfactory prediction accuracy. Sun et al. [10] have proposed the classification algorithms based on conventional and optimized ELM to conduct classification over uncertain data. Firstly, as the training data for learning, we view the instances of each uncertain data. Subsequently, the probabilities of uncertain data in any class are computed according to learning results of each instance. Finally, they have implemented the final classification using abound-based approach. To classify over uncertain data in a distributed environment based on OS-ELM and Monte Carlo theory, they have also extended the proposed algorithms.

Bounhas et al. [14] have extended possibility classifiers, which have been recently adapted to numerical data, to cope with uncertainty in data representation. Here, the possibility distributions, which are used, are supposed to encode the family of Gaussian probabilistic distributions that are compatible with the considered dataset. They have considered two types of uncertainty: (i) the uncertainty associated with the class in the training set, which is modeled by a possibility distribution over class labels, and (ii) the imprecision pervading attribute values in the testing set represented under the form of intervals for continuous data. Moreover, the approach takes into account the uncertainty about the estimation of the Gaussian distribution parameters due to the limited amount of data available. They have first adapted the possibility classification model, which are previously proposed for the certain case, in order to accommodate the uncertainty about class labels. Then, they have proposed an algorithm based on the

extension principle to deal with imprecise attribute values. Qin et al. [18] have proposed a novel Bayesian classification for classifying and predicting uncertain datasets. In modern applications, uncertain data are extensively presented such as sensor databases and biometric information systems. Instead of trying to reduce uncertainty and noise from datasets, this paper follows the new paradigm of directly mining uncertain data. To calculate conditional probabilities, they have integrated the uncertain data model with Bayes theorem and proposed new techniques. Besides laying the theoretical foundations for enhancing naive Bayesian classification to process uncertain datasets, they have showed how to put these concepts into practice. Their experimental assessment demonstrates that the classifiers for uncertain data can be efficiently constructed and efficiently classified and expect even highly uncertain data

Farid et al. [25] have introduced two independent hybrid mining algorithms to develop the classification precision rates of decision tree (DT) and naive Bayes (NB) classifiers for the classification of multi-class problems. For solving classification problems in data mining, both DT and NB classifiers are useful, efficient and commonly used. As the presence of noisy contradictory instances in the training set may cause the generated decision tree suffers from over fitting and its accuracy may decrease, in their first proposed hybrid DT algorithm, they have employed a naive Bayes (NB) classifier to remove the noisy troublesome instances from the training set before the DT induction. In addition, it is computationally very expensive for a NB classifier to compute the class conditional independence for a dataset with high dimensional attributes. Therefore, in the second proposed hybrid NB classifier, they have employed a DT induction to select a comparatively more important subset of attributes for the production of naive assumption of class conditional independence.

Mantas et al. [26] have presented an analysis of a procedure to build decision trees based on imprecise probabilities and uncertainty measures called CDT. Based on the Shannon's entropy for precise probabilities, they have compared their procedure with the classic ones. In the method's performance, they have showed that handling imprecision is a key part of obtaining improvements. With higher level of imprecision, this analysis allows us to extend the CDT's procedure to present a new method. In this extension, both the class variable and the input features are manipulated with rough probabilities. Khushaba et al. [27] have proposed a novel feature extraction method based on the utilization of wavelet packet transform (WPT) and the concept of fuzzy entropy. The steps involved in the acts are, in the first step the WPT is employed to generate a wavelet decomposition tree from which many features are extracted. In the second step, a new algorithm to calculate the fuzzy entropy is developed and adopted as a measure of information content to judge on features suitability in classification, by setting a threshold and removing the features that fall under a certain threshold. In the final step, principle component analysis (PCA) is used to decrease the dimensionality of the generated feature set. As an application, to prove its efficiency the new algorithm is employed in multi-function myoelectric control problem.

Tsang et al. [28] have extended the model of decision tree classification to accommodate data tuples having numerical attributes with uncertainty described by arbitrary pdf's. To build decision trees for classifying such data, they have modified classical decision tree algorithms. They have found that when suitable pdf's are used, exploiting data uncertainty leads to decision trees with remarkably higher accuracies. Hence, they advocate that data can be collected and stored with the pdf information intact. Performance is an issue, though, because of the increased amount of information to be processed, as well as the more complicated entropy computations involved. Therefore, they have devised a series of pruning techniques to improve tree construction efficiency. Their algorithms have been experimentally verified as highly effective. Their execution times are of an order of magnitude comparable to classical algorithms. Some of these pruning techniques are generalizations of analogous techniques for handling point-valued data. Other techniques, namely pruning by bounding and endpoint sampling, are novel.

3 Motivation Behind the Approach

The classifications of uncertain data became one of the tedious processes in data mining domain. The uncertain data contain tuples with different probability distributions, and thus, to find similar class of tuples is a complex process. When we consider uncertain data, the feature vector will not be a single value but a function. Recently, Tsang et al. [28] have proposed a decision tree based uncertain data classification. When multi-class data are given to the decision tree, their algorithm has to give repeated calculation to produce the probability distribution matching the class labels; thus, time and memory utilization will be high for the particular algorithm. Inspired from the research, we proposed a modified decision tree algorithm for handling uncertain data. The proposed algorithm is based on a probability distribution function with the proposed method of selecting the best feature vector with its splitting point. The decision tree makes decision based on probability distribution function of the uncertain data with respect to the class. The major change in decision tree algorithm is regarding the selection of best feature vector and split point to construct an effective and faster decision tree [29].

4 Decision Tree Algorithm

The dataset D consists of a set of tuples and set of attributes (vectors) $D = \{\{T\}, \{A\}\}$ in which the attributes has two parts such as feature vector and decision vector (class label). Each tuple in the set t_i is correlated through feature vector$v$. The feature vector v is correlated through the tuple t and class label c. The classification method is to compose a model M that directs each feature vector $v = [(a_1, a_2, ..., a_n), c_i]$ to a probability distribution P_x on set of classes C. Consider the test tuple t which has the set of feature vector $t = [(v_1, v_2, ..., v_n), c_i]$, probability distribution P over feature vector predicts the class label c with high accuracy.

With the intention of the above criteria, the decision tree algorithm classifies the unusual types of data. A binary decision tree is constructed in the proposed approach with split function z_i. The split function is derived from the values of selected attribute. The split function is used to construct nodes on decision tree to either left or right. A test $v_i < z_i$ predicts the tuple goes to left or right on the decision tree. This will be associated with the distribution of feature vector v_i. Initially, we train the decision tree with the set of training tuples T,$D \Rightarrow T = [t_1, t_1, ..., t_n]$ where n is the total number of tuples in the dataset. Each tuple in the set T is associated with a feature vector v. The value v is associated with the attribute corresponding to the tuple and a class label c. Here, $t \Rightarrow [(a_1, a_2, ..., a_n), c_i]$represents a tuple and with class label. With a class label c_i of a given test tuple $t_{test} \Rightarrow [(a_1, a_2, ..., a_n), c = ?]$, we traverse the tree starting from the root node until a leaf node is reached. When we visit an internal node n, we execute the test $v_{test} \leq z_n$and proceed to the left child or the right child accordingly. Eventually, we reach a leaf node m. The probability distribution Pm associated with m gives the probabilities that t_{test} belongs to each class label $c \in C$. For a single result, we return the class label $c \in C$ that maximizes Pm(c).

5 Uncertain Data and Uncertain Data Handling

The main objective of the proposed approach is concerned with the classification of uncertain data using the decision tree algorithm. As per the discussion in the above section, a feature vector is usually represented by a single value. On the other hand, in uncertain data, the feature vector is represented in a specific range or is known as probability distribution function (pdf). The pdf is defined over range [p, q] over the data and is in a closed form.

The process would be implemented numerically by storing a set of sample points, x, which is a values in the range [p, q] and will be stored on a set S. The value x is associated with the value $f_i(x)$, which is the pdf. The sample points created are used to effectively approximating $f_i(x)$ by a discrete distribution with s possible values. Considering the decision tree scenario to handle the uncertain data, a decision tree in our uncertainty model is of the point data model. The difference lies in the way that the tree is employed to classify test tuples with no class labels. Similar to the training tuples, a test tuple t_{test} contains uncertain attributes. Therefore, the feature vector is thus a vector of pdf's $v = [f_1, f_2, ..., f_n]$. A classification model is thus a function M that maps such a feature vector to a probability distribution P over C. The probabilities for P are calculated as follows. During these calculations, we associate each tuple t_i weight w in the range 0 and 1. The idea interprets the conditional probability of the tuple under test with weight x to identify the class label. The class label that possesses highest probability for tuple will be the class of that particular tuple. Numerous approaches have been proposed for handling and classifying the uncertain data. Here, we proposed an adaptive averaging-based method to classify the uncertain data.

6 Proposed Fuzzy Uncertain Decision Tree Algorithm for Classification of Uncertain Data

The uncertain database contains many attributes, and each tuple has set of data points with respect to the attribute since construction of decision tree with the uncertain data is not an easy task. To solve this problem, in this paper we develop a method which helps to construct the decision tree, and we train the decision tree through the training data of the database with the significant measure. The result of the significant measure from the decision tree represents class of the uncertain data tuple. The construction of decision tree and calculation of the significant measure are given in the following sections.

6.1 Construction of Decision Tree

In this section, we are constructing the decision tree for the uncertain data. The initial step of the method is to select the of best feature vector from the training set of uncertain data tuple. Here, the 80 % of the data tuples are taken for the training data from which the fuzzy entropy of the feature vector and the split point is calculated for selecting the best feature vector and its split point and 20 % of the data tuples are used for testing the decision tree.

6.1.1 Selection of Feature Vector and Split Point

Consider that the uncertain database has a set of tuples and attributes including the class label $D = \{\{T\}, \{A\}\}$, and each tuple has set of data points $T = \{dp\}$ with respect to the attributes. There are many splitting points available from which the root node of the decision tree (first split point)

is selected in order. In this paper, initially we find the best attribute (feature vector) through the fuzzy entropy calculation. Once the calculation of fuzzy entropy of the every feature vector gets finished, the feature vector that has minimum fuzzy entropy is considered as best attribute among them. The next phase is to select the split point from the feature attributes, which is taken using fuzzy entropy. Similar to that of selecting the feature vector, the best of the split point is selected using minimum value of fuzzy entropy. The steps to calculate the fuzzy entropy for the attribute and for the split point are presented below.

Initially, we calculate the fuzzy entropy of the each split point for every attribute with respect to the number of classes present in the uncertain database. The following Eqs. 1 and 2 represent the fuzzy entropy value of the split point in which Eq. (1) represents the match degree of the split point with respect to the class c SP_{ij}^{c} where the value of i represents the split point, the value j represents the attribute a. With the aid of Eq. (1), match degree for every class of each split point is calculated, which is also utilized for calculating the fuzzy entropy value of the split point $F\left(SP_{ij}\right)$ which is represented in Eq. (2). With the aid of the fuzzy entropy value of the split, the fuzzy entropy of the feature vector is calculated. It is represented in Eq. (3).

$$D\left(sp_{ij}^{c}\right) = \frac{\text{No. of dp} \in c \leq sp_{ij}}{\sum_{c=1}^{C} \text{No. of dp} \in c \leq sp_{ij}} \tag{1}$$

$$F\left(SP_{ij}\right) = \sum_{c=1}^{C} sp_c \tag{2}$$

$$F\left(V_j\right) = \sum_{i=1}^{k} sp_{ij} \tag{3}$$

With the intention of selecting the best feature vector (attribute), the above three Eqs. (1, 2, 3) are utilized and the feature vector that has the minimum fuzzy entropy value is elected as the best one. From the selected feature vector, the split point that has the minimum fuzzy entropy is considered as the root node of the decision tree.

Table 1 represents the sample feature vectors and its corresponding fuzzy entropy value, which is calculated by utilizing Eqs. 1, 2 and 3. From Table 1, it can be understood that the feature vector V_3 which is (1.2) has the minimum fuzzy entropy value, when compared to other fuzzy entropy value, and it is selected as the best feature vector. The selected feature vector consists of many split points from which the root node is selected. The split point which consists of minimum root node is considered as the root node. Table 2 represents the fuzzy entropy of the sample split points from the selected feature vector. The selected feature vector V_3 has many split points, from which the minimum value is selected. Table 1 shows that the sp_5 has the minimum value, which is

Table 1 Sample fuzzy entropy for feature vector of the uncertain database

Feature vector	Fuzzy entropy
V_1	2.3
V_2	1.5
V_3	1.2
V_4	2.5
V_5	2.54
V_6	1.7

Table 2 Sample fuzzy entropy for split points of the uncertain database

Split point	Fuzzy entropy
sp_1	0.3
sp_2	0.75
sp_3	0.55
sp_4	0.65
sp_5	0.25
sp_6	0.42

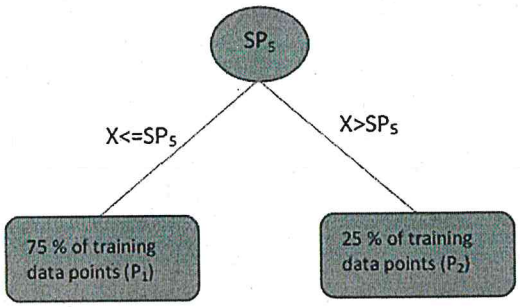

Fig. 1 Representation of decision tree with the selected split point

equal to 0.25. Thus, sp_5 is selected as the root node for the decision tree since the fuzzy entropy value is minimum compared to that of other split points.

Figure 1 shows that the complete training data are fully applied to the split point (root node of the decision tree), and 75 % of the training data points satisfy the condition $x \leq sp_5$ and the 25 % of training data satisfy the condition $x > sp_5$. In order to increase the length of the decision tree, calculate the significant measure for each class of all the partitions (P_i) and significant measure represents the probability value of each class. If all the data in a particular partition (P_i) belong to a single class, there is no need for any further partition. Else, again select the best feature vector and its split point for that particular partition through the above Eqs. 1, 2 and 3. Similarly, increase the length for all the partitions using the given three equations. The calculation of the significant measure is described in the following section.

6.1.2 Calculation of Significant Measure

The proposed significant measure represents the probability of each class in the partition of the leaf node. Using the proposed significant measure, the accuracy of the decision tree is more compared to that of the existing method. The following equation represents the calculation of the proposed significant measure.

$$\mathrm{SM}(c)_i = N(\mathrm{dp})_i \times P(c)_i \times \left[\frac{1}{\sqrt{2\pi\sigma^2(c)_i}} \exp\left(\frac{-\left(N(\mathrm{dp})_i - \mu(c)_i\right)^2}{2\sigma^2(c)_i}\right)\right] \quad (4)$$

In Eq. (4), $\mathrm{SM}(c)_i$ represents the significant measure of the leaf node i, which belongs to the class $(c)_i$ and $N(\mathrm{dp})_i$ represents the number of data points in the leaf node i, of the class c. The symbol $P(c)_i$ represents the probability of the class c in the leaf node i, and $\sigma(c)_i$ represents the average variance of the selected feature vector in the leaf node i that belongs to class c and $\mu(c)_i$ represents that average mean value of the selected feature vector in the leaf node i of the class c.

Figure 2 represents the decision tree with selected split point and the proposed similarity measure. The length of the decision tree depends on the similarity measure of the leaf node with respect to each class. If every data point of the leaf node belongs to any single class, then there is no need to extend the length of that leaf node, else the selection of split point process is repeated for the data present in the leaf node. Consider the above Fig. 2, every data points in (leaf node) partition P2 belong to the class c1; hence, there is no need to increase the length of the decision tree, whereas in case of partition one, each class has a significant measure value, since the partition one is detached into another partition through the selection of next split point, which is done by the above Eqs. 1, 2 and 3. This process is repeated until every data point of the leaf node belongs to single class.

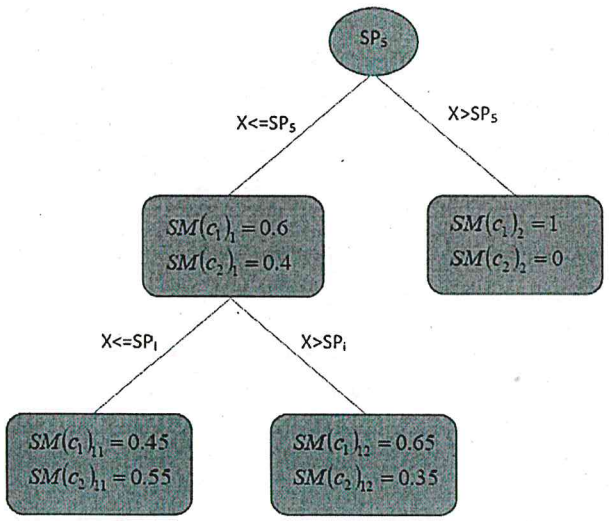

Fig. 2 Representation of decision tree with selected split point and proposed similarity measure

6.2 Evaluation of Decision Tree Through Uncertain Testing Data

Once we trained the decision tree through the set of uncertain data tuples, we evaluate the decision tree through the remaining set of testing data. For each tuple, the decision tree produces the significant measure in each leaf node with respect to each class. With the aid of those measures, in this paper, for each class we calculate the significant measure through the following Eq. 5.

$$\mathrm{SM}(c_i) = \sum_{i,j=1}^{l} \mathrm{SM}(c_i)_{ij} \quad (5)$$

After the calculation of above Eq. 5, the significant measure that consists of the maximum value will hold the class of the test tuple.

Pseudo code

UDB - Uncertain database
dp - Data points
dt - Data tuples
SM (c) - Significant measure of a class c
SP - Split point
V - Feature vector
P - Partition

Begin

1. Read UDB
2. UDB = training UDB + testing UDB
3. For training UDB
4. Call FUDT
5. For testing UDB
6. Call ***test***
End

Subroutine:FUDT

1. for each feature vector
2. for each split point
3. for each class
4. Calculate degree, $D\left(sp_{ij}^{c}\right) = \frac{\text{No.of dp}\in c \le \text{sp}_{ij}}{\sum_{c=1}^{C} \text{No.of dp}\in c \le \text{sp}_{ij}}$
5. End for

6. Calculate fuzzy entropy, $F\left(\mathrm{SP}_{ij}\right) = \sum_{c=1}^{C} \mathrm{sp}_c$

7. end for

8. Calculate fuzzy entropy, $F\left(V_j\right) = \sum_{i=1}^{k} \mathrm{sp}_{ij}$

9. end for

10. min fuzzy entropy $\rightarrow$ *best feature vector* (V_j)

11. min fuzzy entropy from (V_j) $\rightarrow$ *Store best split point* (SP_{ij})

12. (SP_{ij}) $\rightarrow$ *root node*

13. Read training set of UDB

14. If $(dp <= SP_{ij})$

$dp \rightarrow$ *left side* (P_i)

Else

$dp \rightarrow$ *right side* (P_{i+1})

For each partition

For each class

Calculate significant measure, $\mathrm{SM}(c)_i = N(\mathrm{dp})_i \times P(c)_i \times \left[\frac{1}{\sqrt{2\pi\sigma^2(c)_i}} \exp\left(\frac{-(N(\mathrm{dp})_i - \mu(c)_i)^2}{2\sigma^2(c)_i}\right)\right]$

End for

If SM$\in$*>1 class*

Go to step 1 to 11

Assign next split point

Go to step 13

Else

Stop

Subroutine: test

1. for each tuple t_i

2. for each data point

3. Split acc. toSP in FUDT

4. End for

5. for each class

6. Calculate significant measure $\mathrm{SM}(c_i) = \sum_{i,j=1}^{l} \mathrm{SM}(c_i)_{ij}$

7. End for

8. End for

9. Max $\mathrm{SM}(C_i) \rightarrow t_i$

7 Experimental Result and Discussion

The experimental analysis discusses about the performance of the proposed decision tree for classifying the uncertain data. The performance is analyzed based on the modified method using the decision tree algorithm. The proposed approach uses four dataset for the processing of uncertain data classification, which are mainly iris dataset [29], liver disorder dataset [30], breast cancer [31] and echocardiogram dataset [32].

7.1 Performance Evaluation Based on Accuracy

In this section, we evaluate our proposed fuzzy uncertain decision tree algorithm based on accuracy with the four dataset mentioned in the above.

The above Fig. 3 represents the comparison of accuracy of the proposed algorithm with the existing algorithm for the iris dataset. The accuracy of the proposed algorithm FUDT is greater than the existing algorithm UDT. Figure 4 represents the comparison of accuracy of the proposed algorithm with the existing algorithm for the liver disorder database. From Fig. 4, we conclude the thing as we proved our proposed FUDT algorithm has better performance than the existing UDT algorithm in terms of accuracy. The above Fig. 5 repre-

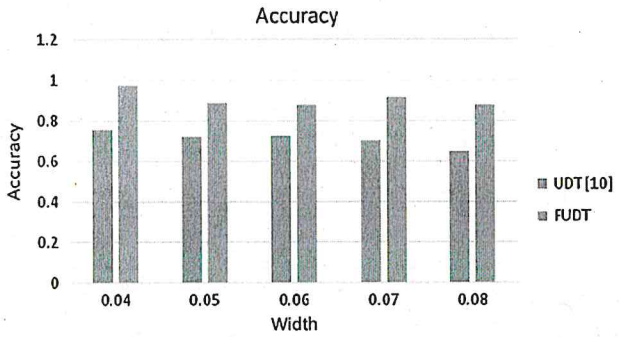

Fig. 3 Accuracy and width for iris dataset

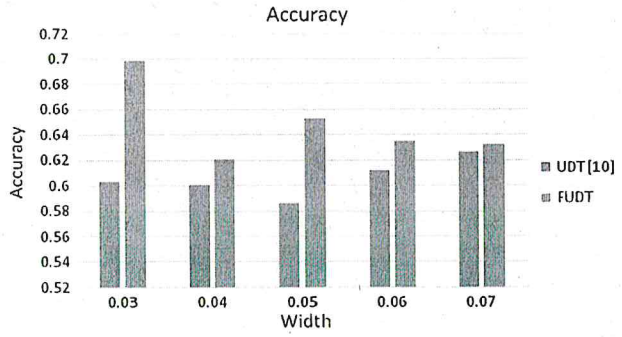

Fig. 4 Accuracy and width for liver disorder dataset

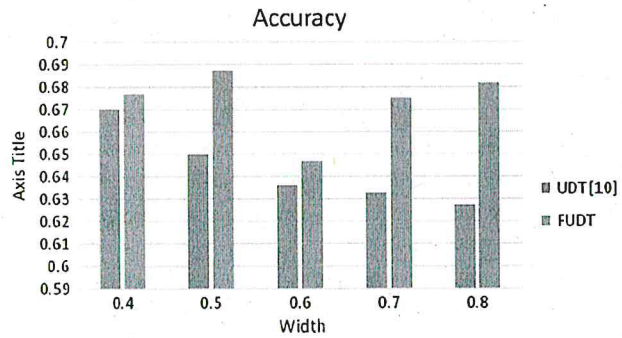

Fig. 5 Accuracy and width for breast cancer dataset

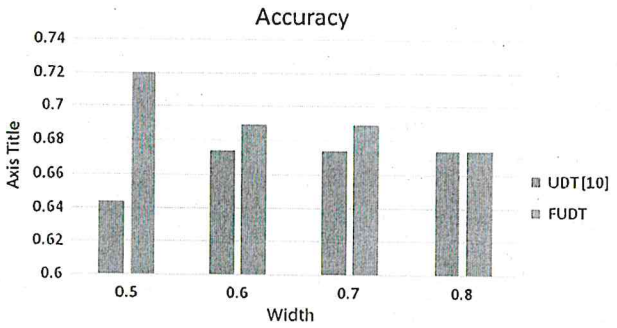

Fig. 6 Accuracy and width for echocardiogram dataset

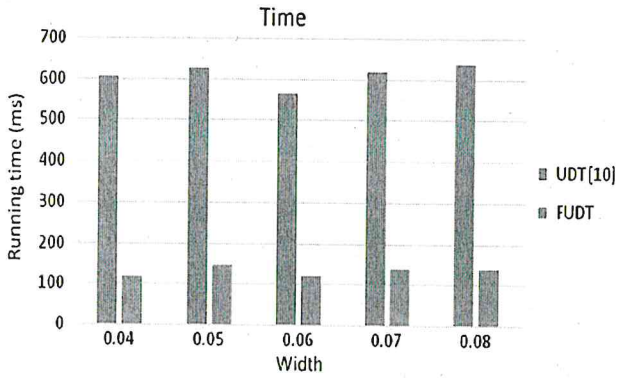

Fig. 7 Running time with that of width for iris dataset

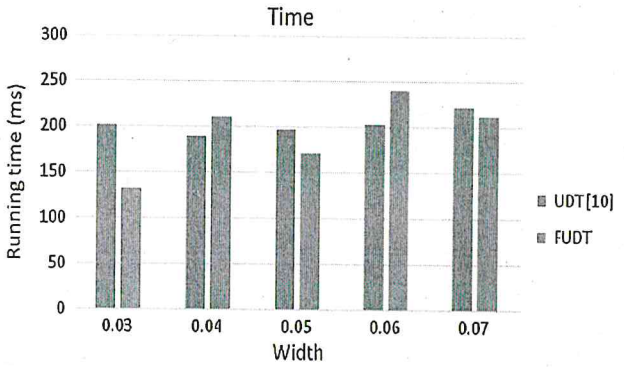

Fig. 8 Running time with that of width for liver disorder database

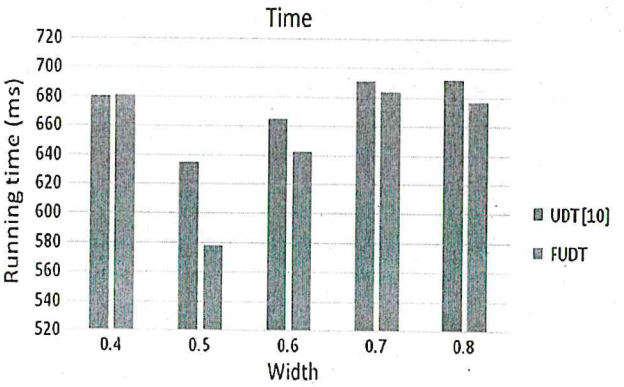

Fig. 9 running time with that of width for breast cancer database

sents the comparison of accuracy of the proposed algorithm with the existing algorithm, which concludes that our proposed algorithm performed well than the existing algorithm in terms of the accuracy. The above Fig. 6 is the representation of the classification accuracy of the proposed algorithm and the classification accuracy of the existing algorithm from where our proposed algorithm classified the uncertain data with good accuracy than the existing algorithm. The reason behind this is the proposed algorithm constructed the decision tree with the proposed fuzzy entropy as a split point chooser and proposed significant measure.

7.2 Comparison with Other Reported Results

While comparing the results proposed by Tsang [7], they have suggested the traditional decision tree algorithms and the extended measures including entropy and information gain techniques for uncertain data. By combining fuzzy theory with an entropy-based measure in our decision free classifier, our proposed method offers an efficient accuracy value which is high compared with the entropy method which is suggested by Smith [7]. The hybridization of fuzzy theory and general entropy eliminates the complication of other entropy methods. The average accuracy rate for iris obtained by Smith [7] is 96.13 %. In our proposed work, we have obtained 98.02 %, while varying the width.

7.3 Performance Evaluation Based on Running Time

The above Fig. 7 represents the comparison of running time of the proposed algorithm with the existing algorithm for the iris dataset. The execution time for the proposed algorithm FUDT is lesser than the existing algorithm UDT since we proved our proposed FUDT algorithm suitable for the iris dataset in terms of running time. Figure 8 represents the comparison of running time of the proposed algorithm with the existing algorithm for the liver disorder database. From Fig. 8, at most our proposed FUDT algorithm needs very less running time than the existing UDT algorithm. For the width value 0.4 and 0.6, our proposed algorithm takes some more time than the existing algorithm. The above Fig. 9 represents the comparison of running time of the proposed algorithm with the existing algorithm, which concludes that our proposed algorithm requires very less running time than the existing algorithm in terms of the running time. The above Fig. 10 is the representation of the execution time of the proposed FUDT algorithm with the existing UDT algorithm. The running time required for the existing UDT algorithm is less than

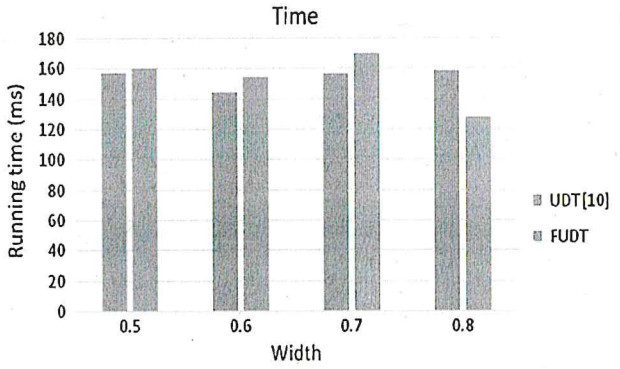

Fig. 10 Running time with that of width for echocardiogram database

the proposed FUDT algorithm for the width value 0.5, 0.6, 0.7. The reason behind this is the proposed algorithm constructed the decision tree along with the calculation of the proposed fuzzy entropy for the each split in every attribute in the uncertain dataset, and the proposed algorithm calculates the significant measure for every class in every leaf node. The above calculation requires more time than the existing since our proposed algorithm does not take much time to classify the uncertain data.

8 Conclusion

In this paper, we have presented an approach for classifying the uncertain data through the binary decision tree with the fuzzy entropy and the proposed significant measure. The fuzzy entropy method is used to select the split points from the training set of uncertain data. The fuzzy entropy method selects the best attributes with the aid of available split point from each attribute. After the fuzzy entropy process, we select the best feature vector which has the minimum fuzzy entropy value. Then, compute the best split point from the selected best feature vector. The length and class of the tuple from the decision tree are decided by the proposed significant measure. After the construction of decision tree, we have evaluated the testing data of the uncertain data. Here, we evaluate our proposed FUDT algorithm with existing UDT algorithm with four different real datasets, and we proved the efficiency of the proposed FUDT algorithm is better than existing UDT algorithm in terms of running time and accuracy.

K-fold cross-validation

K-fold cross-validation is a common technique for estimating the performance of a classifier. Given a set of m training values, a single run of k-fold cross-validation proceeds as follows:

- Initially arrange the training values in a random order.
- Slit the training values into k folds. (k chunks of approximately m/k examples each.)
- For i = 1, ..., k:
- Train the classifier using all the values which do not belong to Fold i.
- Test the classifier on all the values in Fold i.
- Compute n_i, the number of examples in Fold i that were wrongly classified.
- To obtain an accurate estimate to the accuracy of a classifier, k-fold cross-validation is run several times, each with a different random arrangement.

Figures 11, 12, 13 and 14 depict the comparison analysis of accuracy and K-fold validation for four datasets such as iris, liver disorder, breast cancer and echocardiogram datasets; we have taken $K = 10$ so that it can be divided into 5 subsamples; at every iteration, a single sample is taken as training data and remaining are taken as subsample. The process repeats until all the subsamples are utilized as training data. Considering Figs. 11,12 and 14, the accuracy is high at $K = 2$ and at Fig. 13, the accuracy is high at $K = 10$.

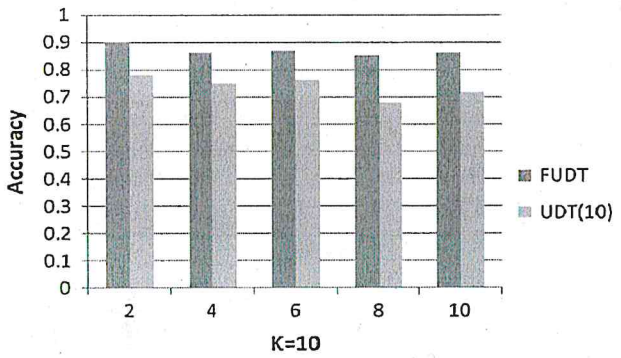

Fig. 11 Accuracy and K-fold validation for iris dataset

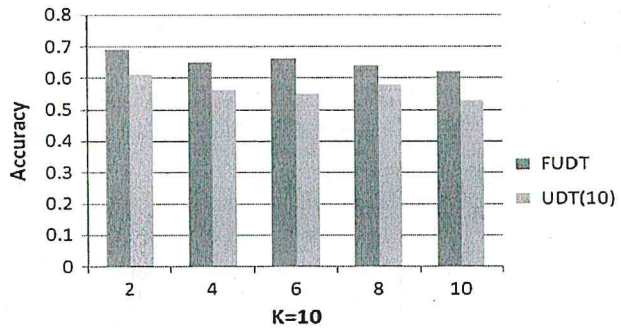

Fig. 12 Accuracy and K-fold validation for liver disorder dataset

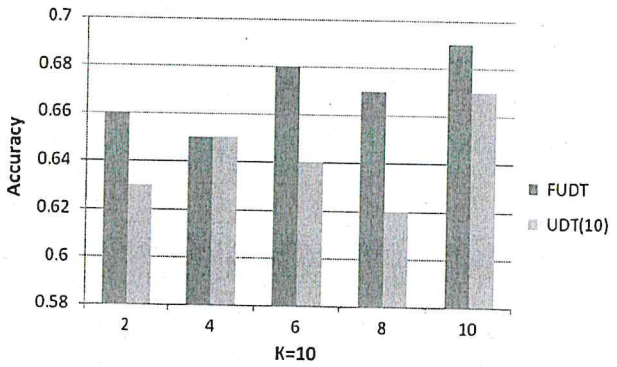

Fig. 13 Accuracy and K-fold validation for breast cancer dataset

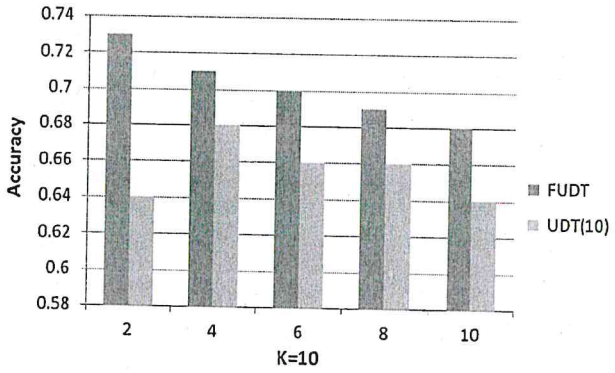

Fig. 14 Accuracy and K-fold validation for echocardiogram dataset

References

1. Denoeux, T.: Maximum likelihood estimation from uncertain data in the belief function framework. IEEE Trans. Knowl. Data Eng. **649**, 119–130 (2013)
2. Charu, C.; Aggarwal.; Yu, P.S.: A survey of uncertain data algorithms and applications. IEEE Trans. Knowl. Data Eng. **21**(5), 609–623 (2009)
3. Barbara, D.; Garcia-Molina, H.; Porter, D.: The management of probabilistic data. IEEE Trans. Knowl. Data Eng. **4**(5), 487–502 (1992)
4. Puente, J.; Fuente, D.; Priore P.; Pino, R.: Abc classification with uncertain data. A fuzzy model vs. a probabilistic model. Appl. Artif. Intell. **16**(6), 443–456 (2002)
5. Cheng, R.; Kalashnikov, D.; Prabhakar, S.: Evaluating probabilistic queries over imprecise data. In: Proceedings of the ACM SIGMOD International Conference on Management of Data (2003)
6. Chau, M.; Cheng, R.; Kao, B.: Uncertain at a mining: a new research direction. In: Proceedings of the Workshop on the Sciences of the Artificial, Hualien, pp. 7–8 (2005)
7. Qin.; Biao.; Xia, Y.; Li, F.: DTU: a decision tree for uncertain data. Adv. Knowl. Discov. Data Mining 4–15 (2009)
8. Choudhary, V.; Jain, P.: Classification: a decision tree for uncertain data using CDF. Int. J. Eng. Res. Appl. **3**(1), 1501–1506 (2013)
9. Appriou, A.: Uncertain data aggregation in classification and tracking processes. Aggreg. Fusion Imperf. Inf. Stud. Fuzz. Soft Comput. **12**, 231–260 (1998)
10. Sun, Y.; Yuan, Y.; Wang, G.: Extreme learning machine for classification over uncertain data. Neurocomputing **128**, 500–506 (2013)
11. Quinlan, J.R.: Probabilistic decision trees. Mach. Learn. **1**, 81–106 (1990)
12. Lobo, O.O.; Numao, M.: Ordered estimation of missing values. PAKDD **239**, 499–503, (1999)
13. Hawarah, L.; Simonet, A.; Simonet, M.: Dealing with missing values in a probabilistic decision tree during classification. In: The Second International Workshop on Mining Complex Data, pp. 325–329 (2006)
14. Bounhas, M.; et al.: Naive possibilistic classifiers for imprecise or uncertain numerical data. Fuzzy Sets Syst. **239**, 137–156 (2013)
15. Angryk, R.A.: Similarity-driven defuzzification of fuzzy tuples for entropy-based data classification purposes. IEEE Int. Conf. Fuzzy Syst. **99**, 414–422 (2006)
16. Kumar, A.; Dadhwal, V.K.: Entropy-based fuzzy classification parameter optimization using uncertainty variation across spatial resolution. J. Ind. Soc. Remote Sens. **38**(2), 179–192 (2010)
17. Qin, B.; et al.: A novel Bayesian classification for uncertain data. Knowl. Based Syst. **24**, 1151–1158 (2011)
18. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufman, Burlington (1993)
19. Cohen, W.W.: Fast effective rule induction. In: Proceeding of the 12th International Conferrence on Machine Learning, pp. 115–123 (1995)
20. Langley, P.; Iba, W.; Thompson, K.: An analysis of Bayesian classifiers. In: National Conference on Artificial Intelligence, pp. 223–228 (1992)
21. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, Berlin (1995)
22. Andrews, R.; Diederich, J.; Tickle, A.: A survey and critique of techniques for extracting rules from trained artificial neural networks. Knowl. Based Syst. **8**(6), 373–389 (1995)
23. Dietterich, T.G.: Ensemble methods in machine learning. Lect. Notes Comput. Sci. **1857**, 1–15 (2000)
24. Farid, D.M.; et al.: Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks. Expert Syst. Appl. **41**, 1937–1946 (2014)
25. Mantas, C.J.; Abellán, J.: Analysis and extension of decision trees based on imprecise probabilities: application on noisy data. Expert Syst. Appl. **41**, 2514–2525 (2014)
26. Khushaba, R.N.; Al-Jumaily, A.; Al-Ani, A.: Novel feature extraction method based on fuzzy entropy and wavelet packet transform for myoelectric control. International Symposium on Communications and Information Technologies (2007)
27. Tsang, S.; et al.: Decision trees for uncertain data. IEEE Trans. Knowl. Data Eng. **23**, 64–78 (2011)
28. Tsang, S.; Kao, B.; Yip, K.Y.; Ho, W.S.; Lee, S.D.: Decision trees for uncertain data. In: Proceeding on International Conference Data Engineering, pp. 441–444, Mar/Apr (2009)
29. Iris dataset https://archive.ics.uci.edu/ml/datasets/Iris
30. Liver disorder dataset https://archive.ics.uci.edu/ml/machine-learning-databases/liver-disorders/bupa.data
31. Breast cancer dataset http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wdbc.names
32. Echocardiogram dataset from https://archive.ics.uci.edu/ml/machine-learning-databases/echocardiogram/echocardiogram.data