World Engineering & Applied Sciences Journal 7 (2): 74-84, 2016
ISSN 2079-2204

DOI: 10.5829/idosi.weasj.2016.7.2.22587

Decision Tree Approach for Classifying Uncertain Data

[1]K. Soundararajan and [2]S. Sureshkumar

[1]Department of Computer Science and Engineering,
Manonmaniam Sundaranar University, Tirunelveli, India
[2]Vivekanadha College of Engineering for Women, India

Abstract: Decision tree is powerful and popular tool for classification and prediction in uncertainty data. This study proposes a decision tree based classification system for uncertain data. The uncertain data means lack of certainty. Data uncertainty comes by different parameters including sensor error, network latency measurements precision limitation and multiple repeated measurements.It is found that decision tree classifier gives more accurate result if take "complete information" of data set is taken. In this paper, the traditional decision tree algorithm is modified by combining firefly and weighted entropy measure. Results obtained from three UCI repositories demonstrate that t he proposed measure results in decision trees that are more compact with classification accuracy that is comparable to that obtained using popular nose splitting measure. The simulation result demonstrates that the proposed system gives better results for uncertain data and it is computationally efficient in terms of accuracy.

Key words: Decision tree · Weighted entropy · Uncertain data · Firefly algorithm · Optimal split point

INTRODUCTION

From large amounts of data the data mining refers to extorting or mining knowledge. In data mining the classification of large data set is a vital problem [1].Classification is one of the most significant data mining methods. Group/class membership is calculated for data instances in data mining [2]. Traditional machine learning algorithms frequently presume that the data values are precise or specific. However, in various emerging applications, the data is intrinsically. Sampling faults and instrument faults are both sources of indecision and data are classically symbolized by probability distributions rather than by deterministic values. In several real world applications uncertain data is ever-present, such as environmental monitoring, sensor network, market study and medical diagnosis [3]. Many factors supply to the uncertainty. It may be cause by indistinctness measurements, network latencies, data staling and decision faults. In categorical attributes and numerical attributes uncertainty can take place [3, 4]. When the value of a numerical attribute is indecisive; the attribute is called an indecisive numerical attribute (UNA) [5]. As data uncertainty is ever-present have to propose a data mining algorithm for uncertain data.

In addition, dissimilar models have been suggested for classification of uncertain data such as Decision tree [6], neural networks [7], Bayesian belief networks, Fuzzy set [8], rule based classification algorithm [9], naïve Bayes algorithm [10] and Genetic models. Besides, Clustering of uncertain data has newly put together interests from researchers. This is constrained by the requirement of using clustering techniques to data that are unsure in nature and a need of clustering algorithms that can tackle the uncertainty. When trying to group the location of objects tracked by means of GPS, the fault may affect the clustering result [11]. For uncertain data broadly used classifier is decision tree classifier. Decision trees are well known as they are practical and uncomplicated to understand [6]. A decision tree is a flow-chart-like hierarchical tree structure is contains three basic elements: decision nodes related to attributes, edges or branches which match up to the different feasible attribute values. The third component is leaves together with objects that classically belong to the similar class or that are very alike [12,13]. In previous decades more than a few decision tree based classifiers are proposed to the classifying the uncertain data. The famous C4.5 classification algorithm [14], ID.3 [13], CART and UDT employed to the uncertain data classification. On the other hand, probability vector

Corresponding Author: K. Soundararajan, Department of Computer Science and Engineering,
Manonmaniam Sundaranar University, Tirunelveli, India.

and probability density function to symbolize uncertain categorical attribute and uncertain numerical attribute correspondingly.

The greater part of decision tree is made up of two main procedures: the building (induction) and the classification (inference) procedures. Rules can as well be extorted from decision trees without difficulty. When decision tree is created with decision tree classification algorithm sometimes it takes place that it produces some discarded & meaningless rules as it develops deeper, it is called as over fitting [15]. This can be evaded by only regarding those characteristics which will have big contribution in forming the specific rule. At precise level, the growth of decision tree is stopped, in order that, the rule created provides enhanced classification. There are two kinds of pruning methods, first is pre-pruning [15, 16], i.e. while building the decision tree keep on verifying whether tree is over fitting based on dissimilar measures like Laplace error [15], MDL [17] length, cost etc and second method is post pruning, in which the tree is built first and next reduction of branches and levels of decision tree is prepared.

The rest of the paper organized as follows: the recent research works is analyzed in section 2; the problem identification is described in the section 3; the proposed work is briefly explained in section 4; the experimental results and discussion are depicted in section 5; and section 6 represents the summary of the work.

Literature Review: Uncertain data mining has been a growing interest in data mining. In addition several data mining methods were proposed by many researches. Ran Wang *et al.* [17] developed learning ELM-Tree from big data based on uncertainty reduction. In their method, the decision tree nodes were dividing with information entropy and ambiguity were employed as uncertainty measures. So as to solve the problem in DT induction because of over partitioning, ELMs were entrenched as leaf nodes, only after the available splits gain ratios tend to lesser than the fixed threshold level. After that, for classified a big data to efficiently decrease the computational time in ELM-Tree model, they applied parallel computation to five components. The result analysis showed that their method reduced the computational time efficiently. But in their method, it cannot be applicable with mixed types of attributes.

Classification of uncertain data using decision trees was proposed by Kiran and Venugopal [18]. In their work, to develop a modified LazyDT ensemble, relevance -based boosting style algorithm1 was proposed. They also presented a distance-based pruning technique, to overcome the over-fitting problem for LazyDT, to design simpler and accurate LazyDT. The result analysis showed that their proposed algorithm was highly effective and also used for designing decision tree using traditional algorithm if they were numerous amounts of data tuples.

Naive possibilistic classifiers for imprecise or uncertain numerical data were proposed by Myriam Bounhas *et al.* [19]. They investigated naive possibilistic classifiers performance, which commencement with the presence of uncertainty. They extended possibilistic classifiers by modified the numerical data, so as to deal with uncertainty in data representation. The possibility distribution was used to encode the Gaussian probabilistic distributions. They considered uncertainty as two types, first one was the uncertainty related with the training class set, designed with possibility distribution in excess of class labels and the second one was beneath the form of intervals only, the indistinctness permeating attribute values presented in the testing set signified for continuous data. The uncertainty data about class labels were accommodated by acclimatized the possibilistic classification model. They developed an algorithm with imprecise attribute values and with imprecise attribute values. The experimental report showed that the possibilistic classifiers for handling uncertainty in data were better when compared with existing method.

Chunquan Liang *et al.* [20] was developed learning very fast decision tree from uncertain data streams with positive and unlabeled samples. In their work, the uncertain data was classified by positive and unlabeled samples. They proposed an algorithm specifically puuCVFDT which was dependent on concept-adapting very fast decision tree (CVFDT) algorithm. The analysis output discriminated that unreal and real-life database described the puuCVFDT algorithm had capacity and efficiency to deal the concept drift with uncertainty data beneath positive and unlabeled learning method.

Uncertain canonical correlation analysis for multi-view feature extraction from uncertain data streams was presented by Wen-Ping Li *et al.* [21]. The canonical correlation analysis (CCA) was a renowned method to obtain similar features from a two multivariate data. But, in uncertain data, it was not possible to extract valuable features from the data; because the data existed uncertainty was employed in many applications. They described an uncertain CCA namely UCCA which was used for feature extraction from uncertain multidimensional data sets. The uncertain linear structure

could characterize well in UCCA in the projected space, with the help of information of uncertainty. Their proposed method is then analyzed with many different real datasets and efficiency of the method in corresponding with multi-view classification based on dimensionality reduction.

Lei Xu and Edward Hung [22] were proposed, improving classification accuracy on uncertain data by considering multiple subclasses. The drawbacks of uncertain data classification, where the data positions were uncertain and the probability density functions (pdf) description were studied in the work. In their work, a supervised Uncertain K-means algorithm was proposed along with multiple subclasses (SUMS). The multiple subclasses (SUMS) considered the object in the same class could split into subclasses. And also, they proposed a bounded supervised UK-means to prevent overfitting by means of multiple subclasses (BSUMS). The experimental results demonstrated that the performance of SUMS and BSUMS had better than traditional algorithm.

A Dynamic Distance Estimation using Uncertain Data Stream Clustering in mobile wireless sensor networks was implemented by Qinghua Luo *et al.* [23]. They developed a dynamic communication distance estimation method based on uncertain interval data stream clustering called Dynamic Distance Estimation method using Uncertain Data Stream Clustering (DDEUDSC). The RSSI data statistical information was used to specify the RSSI-D mapping association regarding interval data. After that the data pattern was considered for composed of some successive cluster centers and applied it in their uncertain RSSI data stream clustering algorithm to evaluate the dynamic communication distance. The experimental results showed the proposed method was an efficient method to increased RSSI-D estimation accuracy in RSSI data stream with uncertainty and dynamics characteristic.

A Bayesian classification for uncertain data was developed by Biao Qin *et al.* [24]. A probabilistic and statistical theory was applied on uncertain data in their method. They developed a method to compute the Bayes theorem conditional probabilities. A Bayesian classification algorithm was developed based on that computation of conditional probabilities, applied for uncertain data. The experimental results showed that the Bayesian classification algorithm classifies uncertain data with potentially higher accuracies than the Naive Bayesian method. It also had a more stable performance than the existing extended Naive Bayesian method.

Problem Definition: This section focus on the problem of decision-tree classification on uncertain data. We explain traditional decision trees in shortly. Then, we discuss splitting measurestouncertain datahandle.

Traditional Decision Trees: Decision tree classiffication is a well-studied problem in data mining and artificial intelligence. A decision tree classifies data items by posing a series of questions about the features associated with the items. Decision tree is a flow chart-like structure consisting of internal nodes, leaf nodes and branches. Each question is contained in a node and every internal node points to one child node for each possible answer to its question. The questions thereby form a hierarchy, encoded as a tree. Each internal node of a decision tree represents a test on an unseen test tuple's feature (attribute). The result of the test decides the branch of the internal node that an unseen test tuple should follow. Each leaf node represents a class or a probability distribution of classes. During the testing of an unseen test tuple, a path is traced from the root to a leaf node of the tree. The prediction of the class label of the test tuple is based on the leaf node reached.

In general, decision tree is a popular classiffication model in machine learning, artificial intelligence and data mining because they are: (1) Practical, with a wide range of applications, (2) Simple and easy to understand and (3) Rules can be extracted and executed manually. Traditionally, numerical values are handled as precise and de?nite point-values. In many applications, however, data uncertainty arises naturally because of: (i) Measurement errors due to equipment limitations, (ii) Data staleness due to transmission bandwidth and (iii) Repeated measurements. Currently, research on data classi?cation mainly focuses on certain data, in which precise and de?nite value is usually assumed. However, data with uncertainty is quite natural in real-world application due to various causes, including imprecise measurement, repeated sampling and network errors [24-30].

Splitting Criteria: A number of methods [30, 31] have been proposed to construct decision trees. These algorithms generally use the recursive-partitioning algorithm and its input requires a set of training examples, a splitting rule and a stopping rule. Partitioning of the tree is determined by the splitting rule and the stopping rule determines if the examples in the training set can be split further. If a split is still possible, the examples in the training set are partitioned into subsets by performing a set of statistical tests defined by the splitting rule.

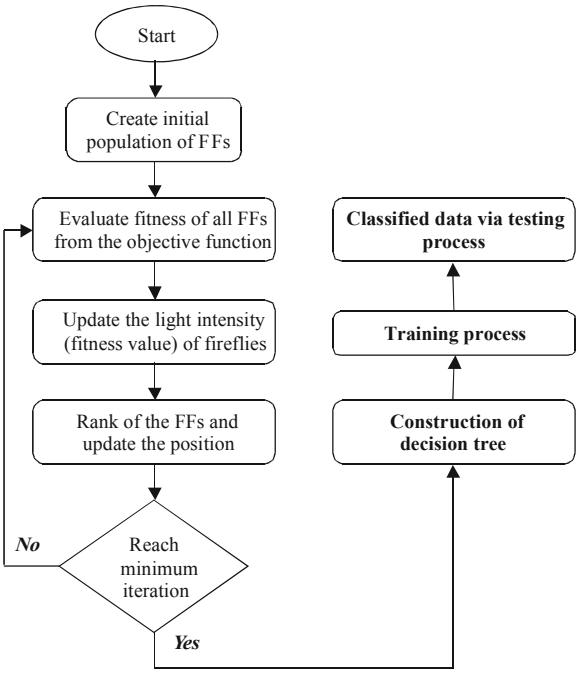

Fig. 1: Overview of the proposed system

The test that results in the best split is selected and applied to the training set, which divides the training set into subsets. This procedure is recursively repeated for each subset until no more splitting is possible. From above consideration, an efficient splitting measure is urgently needed to solve the decision tree classification problem.

Proposed Decision Tree Classification System on Uncertain Data: Classification became a successful data mining technique for uncertain data. Decision tree is powerful and popular tool for classification and prediction.In this paper, we study the problem of constructing decision tree classifiers on data with uncertain numerical attributes. Our goal are (i) to find optimal split point for building decision trees from uncertain data using firefly algorithm. (ii) to design weighted entropy based fitness function in firefly algorithm for building decision trees. The overall diagram of the proposed system is presented in Figure 1.

Decision Tree Construction: In this section, we explain about the method to construct the decision tree for the uncertainty data. In decision tree construction, the goodness of a split is quantified by an impurity measure. One possible function to measure impurity is entropy. Entropy (E) is an information based measure and it is based only on the proportions of tuples of each class in the raining dataset. Entropy is taken as dispersion measure because it is predominantly used for constructing decision trees. In this study, an efficient measure namely weighted entropy (WE) is used, which captures the distribution and correlation information of uncertain data. The proposed measure is designed to reduce the number of distinct classes that would result in each sub-tree after an optimal split via firefly algorithm. After presorting the attribute values (along with class labels), the measure is calculated for each attribute at every successive midpoint of distinct attribute values. The attribute that has minimum measure value is chosen as the splitting attribute at the corresponding split value.

$$W(y) = \sum_{i=1}^{m} w(y_i) H_x(y_i) \tag{1}$$

where,

$$w(y_i) \to 2\left(1 - \frac{1}{1+\exp(-H_x(y_i))}\right)$$

Let the set of uncertain data records before the split be denoted by R.In most of the cases, entropy (E) finds the best split and balanced node sizes after split in such a way that both left and right nodes are as much pure as possible. In this paper, en efficient measure [28] is used namely weighted entropy (HE) combining with firefly algorithm (FA).

Selecting Optimalsplit-point for Splitting via Firefly Algorithm: Once optimal split point is chosen using firefly algorithm, decision tree is constructed based on the minimum weighted entropy value. The intention of the firefly algorithm is to come up with the optimal point selection *R* such that equation (4) is minimized. Selecting optimal split point is an essential task in decision tree. Proper selection of split point gives more accurate results. To formulate this optimization, we develop an optimization based selection namely firefly algorithm (FA) to optimize the initial data selection. Firefly algorithm (FA) is a met heuristic algorithm to solve optimization problems. It was introduced by Xin-She Yang at Cambridge University [29]. The algorithm is inspired by the flashing behaviour of fireflies at night. The flow diagram of the firefly algorithm (FA) is illustrated in Figure 1. Selecting initial optimal attribute is estimated using firefly algorithm to improve the classification performance.

Solution Representation: For optimal attribute selection in decision tree construction, one of the most significant issues is how to symbolize a solution. The solution

representation ties up with the firefly algorithm performance. We define one firefly (solution) as a possible solution in the population. The initial population of fireflies is constructed randomly for firefly algorithm. The initial population of size Y is defined as:

$$Y = A_d \quad (d = 1,2,...,n) \tag{2}$$

where, n is the number of fireflies.

The initialized continuous position values are generated by the follow formula:

$$u_k^* = u_{\min} + (u_{\max} - u_{\min})^* r \tag{3}$$

where, $x_{\min} = 0$, $x_{\max} = 1$ and r is a uniform random number between 0 and 1.

Fitness Evaluation: Fitness function is defined based on our objective. In our work, an optimization formula is derived in equation (1), which is derived based on the minimizing the objective function.

$$W(y) = \min \sum_{i=1}^{m} w(y_i) H_x(y_i) \tag{4}$$

where,
$H_x(y_i)$ → the entropy
$w(y_i)$ → the weight of the entropy of each attribute

Firefly Updation: The movement of the firefly (FF) p, when attracted to another more attractive (brighter) firefly q, is determined by.

$$u'_p = u_p + \gamma(r) * (u_p - u_q) + \varphi \ (rand - \tfrac{1}{2}) \tag{5}$$

The second term in equation (5) is due to attraction, the third term introduces randomization with 'φ' being the randomization parameter and "rand" is a random number generated uniformly distributed between 0 and 1.

Attractiveness, $\gamma(r) = \gamma_0 e^{-\theta r^m}, \qquad m \geq 1$ (6)

where, r is the distance between two fireflies, γ_0 is the initial attractiveness of firefly and θ is a absorption coefficient.

Distance, $r_{pq} = \left\| u_p - u_q \right\| = \sqrt{\sum_{k=1}^{d} (u_{p,s} - u_{q,s})^2}$ (7)

where, $u_{p,s}$ is the s^{th} component of the spatial coordinate of the p^{th} firefly and d is the total number of dimensions. Also $q \in \{1,2,...,F_n\}$ is randomly chosen index. Although q is determined randomly, it has to be different from p. Here F_n is the number of fireflies.

The procedure for optimal attribute selection in decision tree using firefly algorithm as follows:

- Generate an initial population of fireflies randomly (described in solution representation section).
- Evaluate the fitness of each firefly in the population.
- Create a new population by replacing the updation equation (5) until the new population is complete.
- Using the newly generated population for the further sum of the algorithm.
- If the test condition is satisfied, stop and return the best solution in the current population.
- Repeat step 3 until the target is met.
- Finally obtain the optimal split point O_s to decision tree.

Training of UDT: In our model, a dataset consists of UD_r training tuples, t_1, t_2,...,t_d and k numerical (real-valued) feature attributes, A_1, A_2,...A_k. Each tuple t_i is associated with feature vector $V_i = (f_{i,1}, f_{i,2},...,f_{i,k})$ and a class label $c_i \in C$. In this section, we study binary decision tree with tests on numerical attributes. Each internal node n of a decision tree is associated with an attribute n and a split point $z_n \in dom\ (A_j)$, giving a binary test $v_{0,jn} \leq z_n$. An internal node has two children, which are labeled "left" and "right", respectively. Each leaf node m in the decision tree is associated with discrete probability distribution P_m over C. For each $c \in C$, $P_m(c)$ gives a probability reflecting how likely a tuple assigned to leaf node m would have a class label of c. For each internal node n (including the root node), to determine ϕ_n, we first check the attribute A_{jn} and split point z_n of node n. A data item with value uncertainty is usually represented by a probability density function (pdf) over a finite and bounded region of possible values. Since the pdf of t_x under attribute A_{jn} spans the interval $[a_{x,jn}, b_{x,jn}]$ we compute the left andright probability. For each leaf, the discrete probability distribution P_m is computed as follows:

$$P(A) = \frac{Number\ of\ ins\tan ce \in A}{P_L(n)} \tag{8}$$

$$P(B) = \frac{Number\ of\ ins\tan ce \in B}{P_L(n)} \tag{9}$$

$$P_L(n) = \frac{Number\ of\ ins\tan ce \in Left\ leaf}{Total\ number\ of\ ins\tan ce} \tag{10}$$

Table 1: Sample uncertainty database

ID	Attribute	Class
t_1	-1→ 8	
	+10→3	A
t_2	-10→1	
	-1→8	A
t_3	-1 → 5	
	+1 → 1	
	+10 → 2	A
t_4	-10 → 5	
	-1 → 1	
	+1 → 13	B
t_5	0 → 1	
	+1 → 30	
	+10 → 4	B
t_6	-10 → 3	
	+1 → 4	B

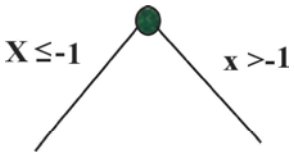

Fig. 2: Decision tree construction at first iteration

For example, suppose the sample uncertainty database is tabulated in Table 1. Given the database of size 6 × 3, tuple t_2 contains two data elements (-10 and -1), where the repeating number of data element (-10) would be 1 and the repeating number of data element (-1) would be 8 and the class label of the corresponding tuple would be *A*. Now, the weighted entropy [28] value, a well-known measure, is used to determine the appropriateness of a current node in constructing a Decision tree (DT).

With the intention of selecting the optimal feature vector (attribute), firefly algorithm (FA) (detailed in section 4.2) is utilized and the feature vector has minimum weighted entropy value which is elected as the best one. From the selected optimal split feature vector, the split point that has minimum weighted entropy (using equation 1) is considered as root node of the decision tree. Once iteration is completed via firefly algorithm.

Using weighted entropy in equation 1, we take the minimum value (based on fitness function) as optimal one. Since the first split point is available (-1). In the left side of the split point (-1) and there is only one split point is available (-10). Since we add that split point (-10) directly on the left side. But, in the right side, there are two split points are available, such as (0 and +1). Since, we select the any one among them. For that further, the calculation is needed to select best among 0 and +1. The initial decision tree would be constructed at first iteration as follows:

The same process is repeated until completion of the final data point using firefly algorithm based on the minimum fitness function for constructing decision tree. Once decision data is taken from the dataset, which data point cannot use next time in the decision tree. Therefore, the dimension of solution encoding process will be changed for each iteration.The algorithm discontinues its execution only if maximum number of iterations is achieved and the solution which is holding the best fitness value is selected until stop the iteration. The final decision tree would be represented as following Figure 3 for training dataset.

Testing of UDT: 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 UD_{ts}. In the testing phase of the decision tree, an unknown test tuple is given to the trained decision tree. The test tuple will be in the similar format like a training tuple but with the class label field empty or unknown. Similar to the training tuples, a test tuple t_0 contains uncertain attributes. Its feature vector is thus a vector of pdf's $f_{0,1}$, $f_{0,2}$,...,$f_{0,k}$. The test tuple t_{test} is given to the decision tree algorithm and the split function plot it into either left of right. Then correspond probability is compared with the probability of test tuple. As per the obtained probability value, the test tuple is plotted to the corresponding class. In similar way we can calculate and classify any kind of uncertain data with unknown class labels. If a single class label is desired as the result, we select the class label with the highest probability as the final result.

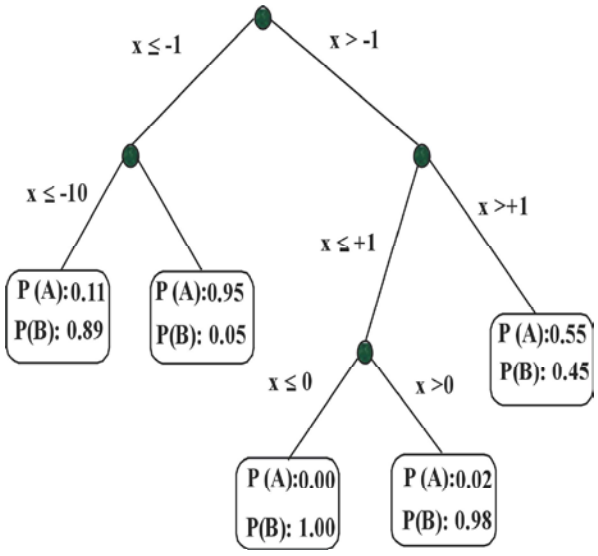

Fig. 3: Trained uncertain database

RESULT AND DISCUSSION

We have offered the results of our suggested methodology and have examined their presentation in this part. The proposed uncertain data classification is implemented in the JAVA program and the classification is experimented with the different dataset and the result is compare with different techniques. The proposed approach uses three dataset for the processing uncertain data classification, which are mainly iris dataset [25], liver disorder dataset [26] and echocardiogram dataset [27]. The suggested uncertain data classification is executed in Java using JDK 1.6 and a series of experiments were performed on a PC with an i5 processor and 4GB of main memory.

Dataset Description: We have generated three types of real world medical datasets such as iris, liver and echo that are taken from UCI machine learning repository.

Iris Dataset: The iris dataset consists of 150 data points distributed over 3 clusters. Each cluster consists of 50 points. This data set represents different categories of irises characterized by four feature values. It has three classes Setosa, Versicolor and Virginica. It is known that two classes (versicolor and virginica) have a large amount of overlap while the class setosa is linearly separable from the other two.

Liver Disorder Dataset: The liver disorder dataset consists of 345 instance and 7 attributes. Here 145 instances are present in class 1 and 200 instances are present in class 2. There are six continuous attributes as dependent attributes and one attribute is class attribute that has value of 1 or 2.

Echocardiogram D ataset: The echocardiogram dataset consist of 132 instance and 12 attributes. The dataset taken from the heart disease patient, some are still alive and some are not. The survival and still-alive variables, when taken together, point out whether a patient survived for at least one year following the heart attack.

Evaluation Matrix: The evaluation of proposed uncertain data classification using three different datasets are carried out using the following metric as suggested by below equations.

Accuracy: The Accuracy is the proportion of true results, either true positive or true negative, in a population. It measures the degree of veracity of a diagnostic test on a condition.

The accuracy can be described by the following equation.

$$A = \frac{T_p + T_n}{T_p + F_p + F_n + T_n} \tag{11}$$

where,

$T_r \rightarrow$ True positive
$T_n \rightarrow$ True negative
$F_p \rightarrow$ False positive
$F_n \rightarrow$ False negative

Performance Evaluation: In this section we evaluate our proposed uncertain data classification based on accuracy with the three datasets mentioned in the above.

Discussion: The methods proposed by Smith Tsang *et al*. [6] and Firefly algorithm with Entropy (FA with E) methods are the best known among existing scheme for uncertain data classification. Furthermore, they characterize the split point performance of the dataset. Therefore, we have chosen to compare the performance of our proposed algorithm Firefly algorithm with Weighted Entropy (FA with WE) against that of these ones. Here, two comparisons are made. (i) A set of experiments is carried out to study the effects on the width of the probability density function's (pdf's) domain as a percentage of the width of an attribute's domain. (ii) Another set of experiments is carried out to study the variation of data size on three datasets. First set of experiment result is presented in Figure 4, 6 and 8. From the first set of result, one can observe that FA with E approach and our proposed approach yield the best performances followed by Smith Tsang *et al*. [6]. Here our proposed approach is slightly better than the FA with E. The above Figure 1 represents effectiveness of width for the liver dataset. When analyzing Figure 4, our proposed approach namely, FA with WE achieves the maximum accuracy of 75.36% which is better than all the other approaches. The Figure 6 represents the effectiveness of accuracy of the proposed algorithm with the existing algorithm for the echo dataset for varying width. When the width is 0.7 the proposed approach FA with WE achieves the maximum accuracy of 97% but FA with E

Performance evaluation on liver dataset

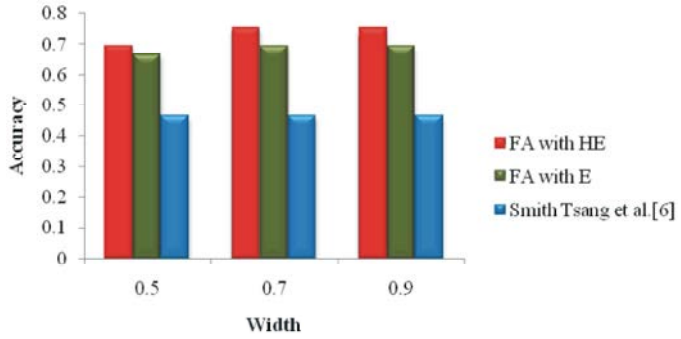

Fig. 4: Effectiveness of width for the liver dataset

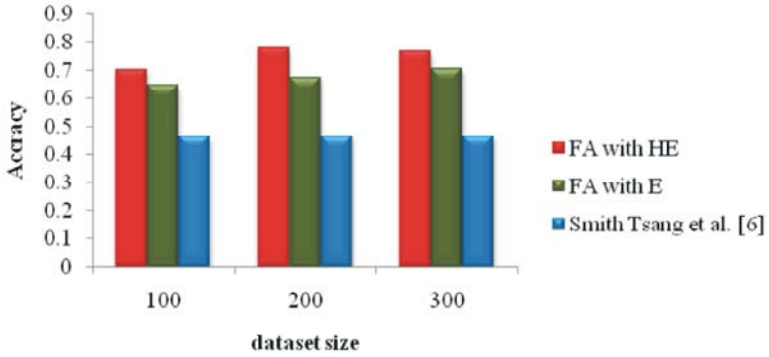

Fig. 5: Evaluation of accuracy for different data size on liver dataset

Performance evaluation on echocardiogram dataset

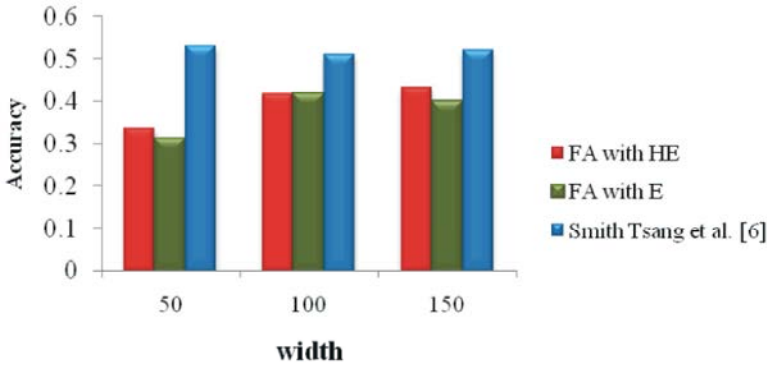

Fig. 6: Effectiveness of width for the echocardiogram dataset

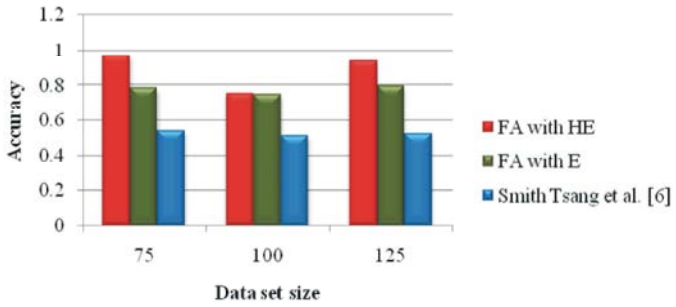

Fig. 7: Evaluation of accuracy for different data size on echocardiogram dataset

Performance evaluation on iris dataset

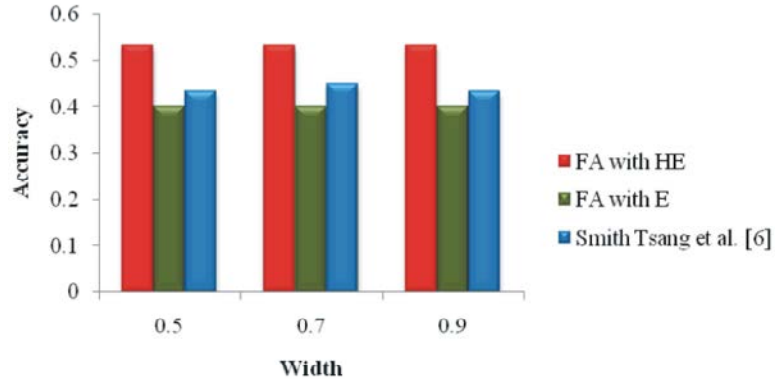

Fig. 8: Effectiveness of width for the iris dataset

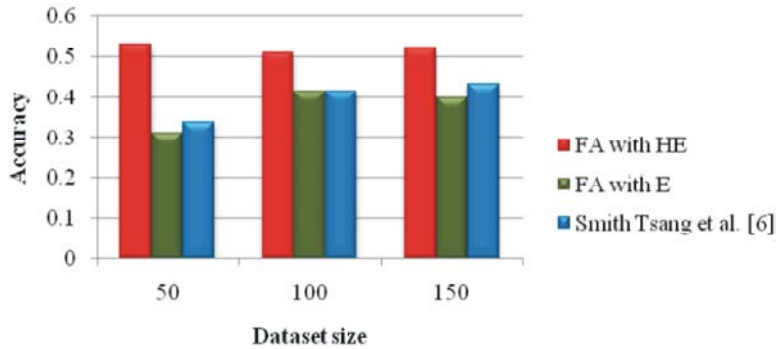

Fig. 9: Evaluation of accuracy for different data size on iris dataset

methods achieves the accuracy of 80% and decision tree using uncertain data classification [6] achieves the accuracy of 55%. From the Figure 6 we conclude the thing is we proved our proposed FA with WE algorithm has better performance than the existing algorithm in terms of accuracy. In Figure 8 represent the effectiveness of width for the iris dataset. When analyzing Figure 9, FA with WE approach achieves the maximum accuracy of 53% which accuracy value is constant for width 0.5, 0.7 and 0.9. Moreover, the Figure 4, 6 and 8 we clearly understand our proposed approach FA with WE is outperformed compare to other two approaches.

Now, we analyze the second set of experiment which is calculating the accuracy of system by varying the size of the dataset. The second experiment result is presented in Figure 5, 7 and 9.

From the result, one can observe that FA with E approach and our proposed approach yield the best performances followed by Smith Tsang *et al*. [6]. Here our proposed approach FA with WE is slightly better than the FA with E. In Fig. 5 illustrates the evaluation of accuracy for different data size on liver dataset. Here the size of the dataset is 200; we obtain the maximum accuracy of 78% for proposed approach which is better than the other approaches. When analyzing the Figure 7, Smith Tsang *et al*. [6] achieves the accuracy of 53%, FA with E achieves the accuracy of 78% and our approach FA with WE achieves the accuracy of 97%. Comparing this accuracy values our approach achieves the maximum values. However in Figure 9, obtain the maximum accuracy of 53% for proposed approach. From the above figures, for the three dataset, our proposed algorithm performed well in terms of accuracy among the three approaches. Since we conclude that, our proposed approach of FA with WE outperformed in terms of accuracy.

CONCLUSION

This systempresented a decision tree based classification system for uncertain data. The uncertain data means lack of certainty. Commonly, data uncertainty comes by different parameters including sensor error, network latency measurements precision limitation and multiple repeated measurements. We found that decision tree classifier gives more accurate result if we take “complete information” of data set. In this paper, we

improved the traditional decision tree algorithm combining firefly and weighted entropy measure. An efficient node splitting scheme was proposed for decision tree construction. Experiment results obtained on three datasets from the UCI repository indicate that the proposed scheme results in decision trees that are more compact.Experimental resultsshown that the proposed system achieved better results for uncertain data based accuracy measure.

REFERENCES

1. Agrawal, R., T. Imielinski and A.N. Swami, 1993. Database mining: A performance perspective, IEEE Transactions on Knowledge and Data Engineering, 5(6): 914-925.
2. Choudhary Varsha and Pranita Jain, 2013. Classification: A Decision Tree for Uncertain Data Using CDF, International Journal of Engineering Research and Applications, 3(1): 1501-1506.
3. Singh Sarvjeet, Chris Mayfield, Sunil Prabhakar, Rahul Shah and Susanne Hambrusch, 2007. Indexing Uncertain Categorical Data, In Proc.of ICDE, pp: 616-625.
4. Qin, B., Y. Xia, S. Prbahakar and Tu Yicheng, 2009. A Rule-based Classification Algorithm for Uncertain Data, The Workshop on Management and Mining of Uncertain Data, pp: 1633-1640.
5. Cheng, R., D. Kalashnikov and S. Prabhakar, 2003. Evaluating probabilistic queries over imprecise data, In: Proceedings of the ACM SIGMOD, pp: 551-562.
6. Tsang Smith, Ben Kao, Kevin Y. Yip, Wai-Shing Ho and Sau Dan Lee, 2011. Decision Trees for Uncertain Data, IEEE Transactions On Knowledge And Data Engineering, 23(1): 64-78.
7. Ge, Jiaqi and Yuni Xia, 2010. UNN: A Neural Network for uncertain data classification, Springer, 6118: 449-460.
8. Suresh, G.V. and O. Srinivasa Reddy, 2011. classification of uncertain data using fuzzy neural networks, world of computer science and information Technology Journal, 1(4).
9. Biao Qin, Yuni Xia, Prabhakar and Yicheng Tu, 2009. A Rule-Based Classification Algorithm for Uncertain Data, in proceedings of IEEE International Conference on.
10. Ren Jiangtao, Sau Dan Lee, Xianlu Chen, Ben Kao, Reynold Cheng and David Cheung, 2009. Naive Bayes Classification of Uncertain Data, in proceedings of ninth IEEE International Conference on.
11. Ngai, W.K., B. Kao, C.K. Chui, R. Cheng and Chau Yip, 2006. Efficient clustering of uncertain data, Springer, Heidelberg, 4065: 436-445.
12. Ilyes Jenhani, Nahla Ben Amor and Zied Elouedi, 2008. Decision trees as possibilistic classifiers, International J. Approximate Reasoning, 48: 784-807.
13. Quinlan, J.R., 1986. Induction of decision trees, Machine Learning, 1(1): 81-106.
14. Ross Quinlan, J., C4.5: Programs for Machine Learning, Morgan Kaufmann, 1993, ISBN 1-55860-238-0.
15. Zhang Yong, 2005. Decision Trees Pruning Algorithm Based on Deficient Data Sets", In Proceedings of the Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies.
16. Arun Poojari, 1999. Data Mining techniques, 150-200.
17. Mehta Manish and Rakesh Agrawal, 1996. SLIQ- A Fast Scalable Classifier for Data Mining, In 5th Intl. Conf. on Extending Database Technology.
18. Siripuri Kiran, M. Venugopal Reddy and P. Niranjan Reddy, 2013. Classification of uncertain data using decision trees, International Journal of Advanced Research in Computer Science and Software Engineering, 3(10): 40-46.
19. Bounhas Myriam, Mohammad Ghasemi, Henri Prade, Mathieu Serrurier and Khaled Mellouli, 2014. Naive possibilistic classifiers for imprecise or uncertain numerical data, Journal on fuzzy sets and systems, 239: 137-156.
20. Liang Chunquan, Yang Zhang, Peng Shi, Zhengguo Hu, 2012. Learning very fast decision tree from uncertain data streams with positive and unlabeled samples, Journal on Information Sciences, 213: 50-67.
21. Wen-Ping Li, Jing Yang and Jian-Pei Zhang, 2015. Uncertain canonical correlation analysis for multi-view feature extraction from uncertain data streams, Journal on Neuro-computing, 149: 1337-1347.
22. Xu, Lei and Edward Hung, 2014. Improving classification accuracy on uncertain data by considering multiple subclasses, Journal on Neuro Computing, 145: 98-107.
23. Luo Qinghua, Xiaozhen Yan, Junbao Li and Yu Peng, 2014. DDEUDSC: A Dynamic Distance Estimation using Uncertain Data Stream Clustering in mobile wireless sensor networks, Journal on Measurement, 55: 423-433.
24. Qina Biao, Yuni Xia, Shan Wanga and Xiaoyong Dua, 2011. A novel Bayesian classification for uncertain data, Journal on Knowledge-based System, 24(8): 1151-1158.

25. Iris dataset https://archive.ics.uci.edu/ml/datasets/Iris.
26. Liver disorder dataset https: // archive.ics.uci.edu/ml/machine-learning-databases/liver-disorders/bupa.data.
27. Echocardiogram dataset from https://archive.ics.uci.edu/ml/machine-learning-databases/echocardiogram/echocardiogram.data.
28. Wu, Shu and Shengrui Wang, 2013. Information-Theoretic Outlier Detection for Large-Scale Categorical Data, Knowledge and Data Engineering, 25(3): 589-602.
29. Yang, X.S., 2008. Nature-inspiredmetaheuristic algorithm, University of Cambridge, United Kingdom: Luniver Press.
30. Breiman, L., J.H. Friedman, R.A. Olshenand and C.J. Stone, 1984. Classification and Regression Trees, Wadsworth International Group, Belmont, California.
31. Quinlan, J.R., 1993. Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, California.