

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/283802637>

Possibilistic Very Fast Decision Tree for Uncertain Data Streams

Article · January 2015

DOI: 10.1007/978-3-319-19857-6_18

CITATIONS

0

READS

28

2 authors, including:



[Mohamed Salah Gouider](#)

Université de Tunis

32 PUBLICATIONS 45 CITATIONS

SEE PROFILE

Possibilistic Very Fast Decision Tree for Uncertain Data Streams

Mohamed Hamroun and Mohamed Salah Gouider

Abstract This paper addresses the classification problem with imperfect Data Streams. More precisely, it extends standard CVFDT to handle uncertainty in both building and classification procedures. Uncertainty here is represented by possibility distributions. The first part investigates the issue of building decision trees from Data Streams with uncertain attribute values by developing a non-specificity based information gain as the attribute selection measure which, in our case, is more appropriate than the standard selection measure based on Shannon entropy. The extended approach so-called Possibilistic Very Fast Decision Tree for Uncertain Data Streams (Poss-CVFDT) offers a more flexible building procedure. The second part addresses the classification phase. More specifically, it investigates the issue of predicting the class value of new instances presented with certain and/or uncertain attribute values.

Keywords Classification • Uncertainty • Possibility theory • Non specificity • Data streams • Decision tree

1 Introduction

Data Stream Classification represents an important task in machine learning and data mining applications. It consists in inducing a classifier from a set of historical examples with known class values and then using the induced classifier to predict the class value (the category) of new objects given known the values of their attributes (features). Classification is widely used in many real world applications including pharmacology, medicine, marketing, etc. However, classification with data streams is a very challenging task. Therefore, effective algorithms need to be designed in order to take into account temporal locality and the concept drift of the data.

M. Hamroun (✉) · M.S. Gouider
Institut Supérieur de Gestion, Laboratoire SOIE, Tunisia, Africa
e-mail: mohamed.hamroun@gmail.com

M.S. Gouider
e-mail: ms.gouider@yahoo.fr

To address these issues, many algorithms have been proposed, mainly including various algorithms based on ensemble approach and decision tree, e.g., [1] proposed algorithms to learn very fast decision trees (VFDT), [2] proposed a novel algorithm called concept-adapting Very Fast Decision Tree (CVFDT) which can learn decision trees incrementally with bounded memory usage, high processing speed, and detecting evolving concepts. In real world applications, the massive amounts of data are inseparably connected with imperfection. In fact, data can be imprecise or uncertain or even missing. These imperfections might result from using unreliable information sources. Standard classifiers, generally, ignore such imperfect data by rejecting them or replacing each imperfect data item by an arbitrary certain and precise value or by a statistical value such as a median, mode or a mean. This is not a good practice because it alters the real observed data. Consequently, ordinary Data Stream classification techniques such as CVFDT should be adequately adapted to take care of this problem.

Our idea is to treat different levels of uncertainty using possibility theory which is a non-classical theory of uncertainty [3]. More precisely, we will handle samples whose attribute values are given in the form of possibility distributions. We also adapt the attribute selection measure, used in the building phase, to the possibilistic framework by using a non-specificity based criterion instead of the Shannon entropy. In addition, we introduce a new sliding model based on samples's timestamp in order to improve the ability of CVFDT to cope with concept drift issue. Such possibilistic decision tree will be referred to by Poss-CVFDT. The paper is organized as follows: Sect. 2 presents a summary of related works. Section 3 proposes an extension of CVFDT, namely the Poss-CVFDT approach. This section defines the building procedure, then, it describes the method that we propose for the classification process. Before concluding, Sect. 4 presents and analyzes experimental results carried out on modified versions of commonly used data sets obtained from the U.C.I. machine learning repository.

2 Related Works

Uncertain Data Streams

Some previous work focused on building classification models on uncertain data examples. All of them assume that the Probability Density Function (PDF) of attribute values are known. [7] developed an Uncertain Decision Tree (UDT) for uncertain data. [6] developed another type of decision tree DTU for uncertain data classification. [8] proposed a neural network method for classifying uncertain data. Since the uncertainty is prevalent in data streams, the research of classification is dedicated to uncertain data streams nowadays. Unfortunately, only a few algorithms are available. [9] proposed two types of ensemble classification algorithms, Static Classifier Ensemble (SCE) and Dynamic Classifier Ensemble (DCE), for mining uncertain data streams. [10] proposed a CVFDT based decision tree named UCVFDT for uncertain

data streams. UCVFDT has the ability to handle examples with uncertain attribute values by adopting the model described by [5] to represent uncertain nominal attribute values. In their work, uncertainty is represented by a probability degree on the set of possible values considered in the classification problem.

3 Poss-CVFDT

3.1 Data Streams Structure

Instead of rejecting instances having uncertain attributes values or adding a null attribute value to such instances, we use the possibility theory. More formally, we propose to represent the uncertainty on the attributes values of instances by a possibility degree on the set of possible values considered in the classification problem. Among the advantages for working under the possibility theory framework, we recall that the two extreme cases, total ignorance and complete knowledge, are easily satisfied. Given a stream data $S = \{S_1, S_2, \dots, S_t, \dots\}$ where S_t is a sample in S , arriving at time T_i for any $i \in \mathbb{N}$, $T_i < T_n$, we denoted by $S_t = \langle X^{uc}, y \rangle$.

- Here $X^{uc} = \{X_1^{uc}, X_2^{uc}, \dots, X_d^{uc}\}$ is a vector of d uncertain categorical attributes. Given a categorical domain $Dom(X_i^{uc}) = \{v_1, v_2, \dots, v_m\}$, X_i^{uc} is characterized by possibility distribution over Dom , where $0 \leq \pi(X_i^{uc} = v_j) \leq 1$
- $Y_k \in C$ denotes the class label of S_t , where C is set of class labels on stream data $C = \{y_1, y_2, \dots, y_k, \dots, y_n\}$.

we build and learn a fast decision tree model $y_k = f(X^{uc})$ from S to classify the unknown samples.

The incoming sample will be passed down to a certain node from the root recursively according to its attribute value. Here we adopt a recursive process to partition the training samples into fractional samples based on [7, 10]. By giving a split attribute at node N , denoted by X_i^{uc} , sample st is divided into a set of samples $\{st_1, st_2, \dots, st_m\}$ by X_i^{uc} where $m = |Dom(X_i^{uc})|$ and st_j is a copy of st except for attribute X_i^{uc} .

3.2 Sliding Window Model

During the classification technique, we emphasis the concept drift issue which change the classifier result over time. The capture of such changes would help in updating the classifier model effectively. The use of an outdated model could lead to very low classification accuracy. That is why we need to focus only on the N recent records. Based on timestamp of samples, a new approach have been proposed to detect and identify outliers in the underlying data [11]. This model can be an interesting task in many sensor network applications.

3.3 Specificity Gain

Classical splitting measures such as Information Gain, Gini Index are not applicable in our case. In this section, based on possibility measures, we define new parameters which take into account the uncertainty encountered in the data flow. An interpretation have been proposed by [12, 13] based on the context of possibility theory assuming that the U-uncertainty measure of nonspecificities of possibility distribution, which is defined as

$$Nonspec(\pi) = \sum_{i=1}^n [(\pi(\omega_{(i)}) - \pi(\omega_{(i+1)})) \log_2 i] + (1 - \pi_{(1)}) \log_2 n \quad (1)$$

can be justified as a proper generalization of Hartley Information [14] to the possibilistic setting. Non Specificity plays the same role to that of Shannon Entropy in probability theory. So we use it to construct a selection measure in the same way as information gain and Information gain ratio are constructed from Shannon Entropy. We calculate a Specificity Gain based on the nonspecificities on the possibility distributions $\pi_{X_i^{uc}}$ on the values of X_i^{uc} , π_C on the set of class labels and $\pi_{CX_i^{uc}}$ the joint distribution of the set of values of X_i^{uc} and the set of classes. The idea so suggests itself to construct these possibility distributions but we have to take into consideration the concept underlying possibility theory. The solution that we propose is the following : We will induce a representative possibility distribution that represents the marginal distribution of the different possibility degrees of the different values of X_i^{uc} and the set of class labels C. These marginal distributions are obtained not by summing values but by taking their maximum. Then, the average is applied as the case:

$$\pi_{X_i^{uc}}(vj) = Avg_{|C|}(\max_{y \in C}(\pi_{CX_i^{uc}}(y, vj))) \quad (2)$$

$$\pi_C(y) = Avg_{|Dom(X_i^{uc})|}(\max_{vj \in Dom(X_i^{uc})}(\pi_{CX_i^{uc}}(y, vj))) \quad (3)$$

We define the *Specificity gain* as:

$$S_{gain}(X_i^{uc}) = Nonspec(\pi_C) + Nonspec(\pi_{X_i^{uc}}) - Nonspec(\pi_{CX_i^{uc}}) \quad (4)$$

We select for test, the attribute which yields the highest S_{gain} .

In order to collect statical data to S_{gain} , we only make a single pass over set Samples SN. So we propose, at each node of the nodes of the dynamic decision tree, for each possible value vj of each attribute $X_i^{uc} \in X^{uc}$, for each class yk , to associate it with a vector called Node Feature(NF) which is composed by four parameters $(PS_{ijk}, N_{ijk}, N(S), t_s)$. Here,

- PS_{ijk} denotes the sum of possibility degrees of each coming sample in yk that $X_i^{uc} = v_j$.
- $N(S)$ defines the number of the samples contained in the NF.
- N_{ijk} denotes the ratio of the sum of possibility degrees of each sample in yk that $X_i^{uc} = v_j$, $N_{ijk} = \frac{PS_{ijk}}{N(S)}$.
- t_s indicate the time of the recent sample incorporated in the NF.

→ for each new coming sample S^{new} , we only need to update PS_{ijk} , N_{ijk} , $N(S)$ as follows:

$$PS_{ijk} = PS_{ijk} + \pi(X_{i(S^{new})}^{uc} = v_j) \quad (5)$$

$$N(S) = N(S) + 1 \quad (6)$$

$$N_{ijk} = \frac{PS_{ijk}}{N(S)} \quad (7)$$

→ for removing sample S^{old} from a node, we only need to update PS_{ijk} , N_{ijk} , $N(S)$ as follows:

$$PS_{ijk} = PS_{ijk} - \pi(X_{i(S^{old})}^{uc} = v_j) \quad (8)$$

$$N(S) = N(S) - 1 \quad (9)$$

$$N_{ijk} = \frac{PS_{ijk}}{N(S)} \quad (10)$$

3.4 Poss-CVFDT Building

Based on [2], Algorithm 1 defines a pseudo code of the Poss-CVFDT building steps. In this study we use S_{gain} as $G(\cdot)$, X^{uc} denotes a vector of uncertain attributes, \overline{NF}_{ijk} : the NF used to collect statical data for computing specificity gain. t_s defines the arrival time of each sample used to enhance the ability of Poss-CVFDT to detect outdated samples.

Algorithm 1 illustrates the process of Poss-CVFDT learning in four steps : At the beginning, as the classical technique CVFDT, our algorithm does some initializations and then process each sample (X^{uc}, y) to the leaves (1–7). Next from line 8 to 15 each sample arrived is associated with a timestamp. Then it added to the sliding window. Old samples are forgotten from the tree as well as from the window. We detect and identify outdated samples according to their arrival time. The step 16 is for maintaining the tree (Algorithm 2). Finally, from line 17 to 20 we check the split validity of an internal node periodically.

Algorithm 2 details the growing of the uncertain decision tree. The node Feature's parameters are maintained between lines 3 and 7. From line 8 to line 18, a sample is split into a set of fractional samples. From line 19 to 29, the specificity gain is computed using the node feature at leaf nodes. Based on Hoeffding bound, split attribute is chose and the leaf node is split into an internal node.

Algorithm 1 illustrates the process of Poss-CVFDT learning:

Algorithm1: The learning algorithm for Poss-CVFDT

Inputs

S a stream of samples;
 X^{UC} an uncertain categorical attribute vector;
 $\bar{N}F_{ijk}$ a Node feature : a vector associated to each node ;
 $G(.)$ Specificity gain for split evaluation;
W the size of the Window ,based time;
 T_i the arrival time of each sample ;
 N_{min} the number between checks for growth;
 $N(s)$ defines the number of the samples contained in the node Feature;
f the number between checks for drift;

Outputs

HT a decision tree for uncertain samples

```

1 Begin
2 Let HT be a tree with a single leaf  $l$  , the root;
3 Let ALT(1) be an initially empty set of alternate trees for  $l$ ;
4 For each each class  $y_k$  do
5   For each possible value  $v_j$  of each attribute  $X_i^{uc} \in X^{uc}$ 
6     Let  $N_{ijk} = 0$  ,  $PS_{ijk}=0$  ,  $N(s) =0$ 
//Initialize the Node Feature  $\bar{N}F_{ijk}(l)$ 
7   End For
8 End For
9 For each sample  $(X^{uc} , y)$  in S do
10 Sort  $(X^{uc} , y)$  into a set of leaves L using HT of any node  $(X^{uc} , y)$ 
passes through
11  $t_s$  indicate the time of the recent sample incorporated in
the node Feature  $t_s = T_n$ 
12 Add  $((X^{uc} , y), t(s))$  to the beginning of W
13 Let  $((X_w^{uc} , y_w, t(s))$  be the last element of W.
14 Forget Samples (HT,  $((X_w^{uc} ; y^w), t(s))$ ) where  $T_i < t(S) - W$ 

15 Let W with  $((X_w^{uc}; y_w); t(s))$  removed
16 Poss-CVFDTGROW(HT,G,  $(X^{uc} , y), n_{min}$  )
// refer to Algorithm 2
17 If there have been f samples since the last checking of alternate trees
18 CheckSplitValidity(HT)
19 Return HT
20 End.
```

Algorithm 2 details the growing procedure of the uncertain decision tree.

Algorithm2: Poss-CVFDTGROW(**HT**,**G**, (X^{uc} , **y**), n_{min})

```

1 Let  $l$  be the root of HT
2 Let ALT(1) be an initially empty set of alternate trees for 1;
3 For each class  $y_k$  do
4   For each possible value  $v_j$  of each attribute  $X_i^{uc} \in X^{uc}$ 
5     Maintain the  $\bar{NF}_{ijk}$  following formula (5) , (6) , (7)
6   End For
7 End For
8 For each  $T_{ALT}$  in ALT(1) do
9   Poss-CVFDTGROW( $T_{ALT}$ ,G, ( $X^{uc}$  , y),  $n_{min}$  )
10 End For
11 Label  $l$  with the majority class among the samples seen so far at  $l$ 
12 If  $l$  is not a leaf
13   Split (  $X^{uc}$  , y) into a set of fractional samples FS
14   For each sample  $S_j$  in FS do
15     Let  $l_j$  be the branch child for  $S_j$ 
16     Poss-CVFDTGROW( $l_j$ ,G,  $S_j$  ,  $n_{min}$  )
17   End For
18 End IF
19 Else If  $N(S) > n_{min}$ 
20   Compute  $G(X)$  for each attribute  $X_i^{uc} \in X^{uc}$  based on  $\bar{NF}_{ijk}(l)$  and formula (4)
21   Let  $X_a^{uc}$  ,  $X_b^{uc}$  be the attribute with highest and second-highest  $\bar{G}$ 
22   Compute  $\epsilon$ 
23   If  $\Delta \bar{G}_j = \bar{G}(X_a) - \bar{G}(X_b) > \epsilon$ 
24     Replace  $l$  by an internal node that splits on  $X_a^{uc}$ 
25     For each class  $y_k$  and each  $v_j$  possible value of each attribute  $X_i^{uc} \in X^{uc}$ 
26       Initialize  $\bar{NF}_{ijk}(l_j)$ 
27     End For
28   End If
29 End IF
30 Return HT

```

3.5 Prediction with Poss-CVFDT

Once a Poss-CVFDT is constructed, it can be used for predicting class types. The prediction process starts from the root node, the test condition is applied at each node in the tree and the appropriate branch is followed based on the outcome of the test. When the test sample S is certain, the process is quite straightforward since the test result will lead to one single branch without ambiguity. When the test is on an uncertain attribute, the prediction algorithm proceeds as follows:

Given an uncertain test sample st, $\langle X^{uc}, ? \rangle$, If the test condition is on a uncertain categorical attribute $X_i^{uc} \in X^{uc}$ and $\text{Dom}(X_i^{uc}) = \{v_1, v_2, \dots, v_m\}$ a categorical domain which characterized by a possibility distribution where $0 \leq \pi(X_i^{uc} = v_j) \leq 1$.

For the leaf node of Poss-CVFDT, each class $yk \in C$ has a possibility degree $\pi_C(yk)$ which is the possibility degree for an instance to be in class yk if it falls in this leaf node. $\pi_C(yk)$ is computed based on the node feature and more especilly on N_{ijk} in the node. Assume path L from the root to a leaf node contains t tests, and the data are classified into one class yk in the end, When predicting the class type for a sample S with uncertain attributes, it is possible that the process takes multiple paths. Suppose there are m paths taken in total, then the possibility degree of yk can be computed as the fraction of the total of possibility degree obtained at each path and the number of paths (m) : $\pi_C(yk) = \frac{\sum_{i=1}^m \pi_{yk}^i}{m}$.

Finally, the sample will be predicted to be of class yk which has the largest possibility degree $\pi_C(yk)$ among the set of class lables, where $k=1,2,\dots,|C|$ and $\text{possidist} = \{\pi_C(y_1), \pi_C(y_2), \pi_C(y_3), \dots, \pi_C(|C|)\}$ a possibility distribution over classes at a node leaf.

4 Exprimnt Study

In this section, we present the experimental results of the proposed decision tree algorithm Poss-CVFDT. A collection containing 3 real-world benchmark datasets were assembled from the UCI Repository. The evaluation of our Poss-CVFDT classifier was performed based mainly on three evaluation criterias, namely, the classification accuracy expressed by the percentage of correct classification (PCC), F1 Measure and the running time (t).

4.1 Artificial Uncertainty Creation in the Training Set

Due to a lack of real uncertain datasets, we introduce synthetic uncertainty into the datasets. We treated uncertainty in our approach by assigning a possibility distribution to attribute values. These possibility degrees are created artificially (Table 1).

Example:

- Education: Bachelors (B), Masters (M), Doctorate (D).
- Occupation: Technique support (TS), Craft repair (CR), Sales (S).
- Native_country: United States (US), Cambodia (C), England (E).

Table 1 The classical training set

S	Education	Occupation	Native_country
X1	B	TS	C
X2	D	CR	E

Table 2. details the new training set with uncertain attribute values.

Table 2 Training set based on possibility distributions

S	Education			Occupation			Native_country		
	B	M	D	TS	CR	S	US	C	E
X1	1	0	0	0.1	1	0.7	1	0.2	0.9
X2	0.5	1	0.6	1	0.7	0.2	1	0.3	0.4

4.2 Simulations on the Real Data Sets

We have modified UCI databases by introducing uncertainty in the attributes values of their instances as presented in Table 3.

Table 3 Description of datasets

Datasets	# Instances	# Attributes	# Classes
Solar-Flare	1389	10	3
Car evaluation	1728	6	4
Nursery	12960	8	5

4.3 Experimental Results: Poss-CVFDT VERSUS UCVFDT

We compare in this section the results obtained by our proposed approach and those obtained by the UCVFDT.

1. **Comparison of Classification Accuracy and F1 Measure:** Comparing two different methods which work under two different frameworks seems to be not imperative. This section presents different results carried out from testing Poss-CVFDT and UCVFDT. We consider P as the percentage of attributes with uncertain values from the training set defined such that: $0 < P < 50$ or $50 < P < 100$. The percentage of P is useful to test the behavior of our classifier and its robustness in dealing with uncertainty. These uncertainty percentages represent the percentages of generated uncertain objects of one given dataset. For example, if we

fixe $P > 50\%$, it means that, for a given database which contains 10 attributes, more than 5 will be generated with uncertainty. These values of P allow us to generate the uncertain databases. We take g as the possibility degree defined for the different values of an attribute: $0 < g < 0,5$ or $0,5 < g \leq 1$.

As seen in Figs. 1 and 2, both Accuracy and F1 of Poss-CVFDT are higher than that of UCVFDT. We can see that our approach achieves encouraging results when $P < 50$ and $g \in [0,0.5]$. As shown in Figs. 3 and 4, when the degree of uncertainty increases $P > 50$, both accuracy and F1 of Poss-CVFDT decline slowly. We can also observe that, both accuracy and F1 of Poss-CVFDT are quite comparable to those of UCVFDT. This demonstrates the robustness of our approach against the uncertainty is due to the possibility theory which handle the imprecise data efficiently by treating all the extreme cases.

- 2. **Running Time:** Figure 5 illustrates the comparison between UCVFDT and our approach in term of processing time. It is obvious that these encouraging results are obtained due to the sliding model which processes only the most recent records by discarding the obsolete ones. This mechanism allows obtaining good results in few times and with high accuracy.

Fig. 1 PCC with $P < 50$ and $g < 0.5$

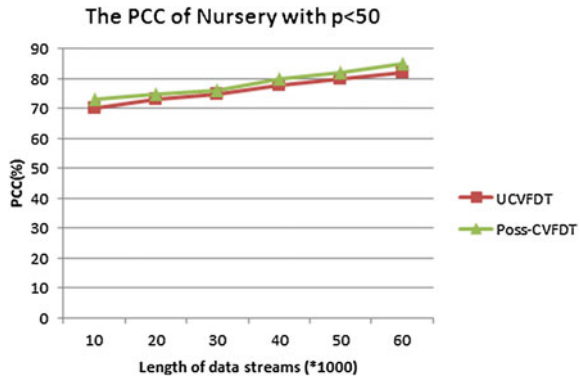


Fig. 2 F1 with $P < 50$ and $g < 0.5$

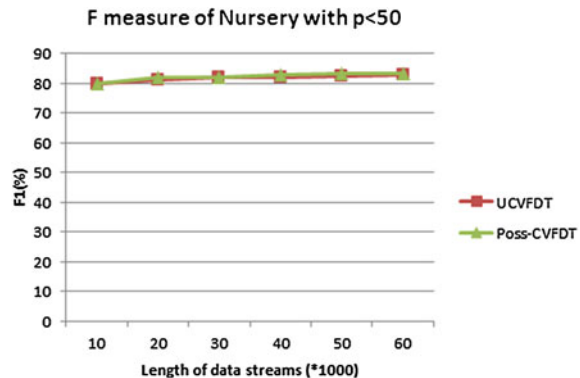


Fig. 3 PCC with $P < 50$ and $g < 0.5$

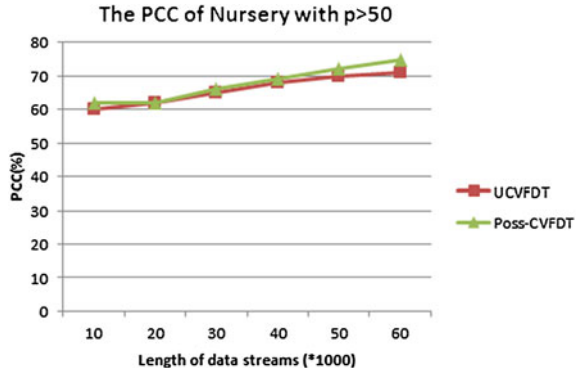


Fig. 4 F1 with $P < 50$ and $g < 0.5$

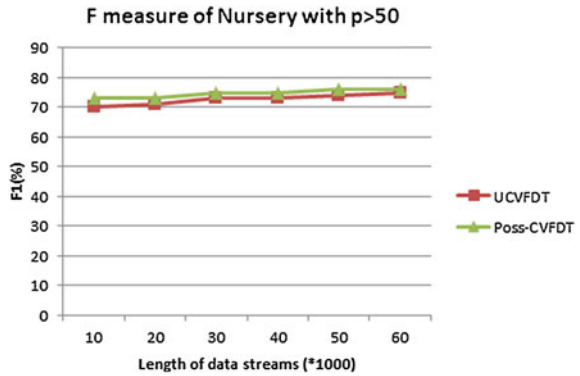
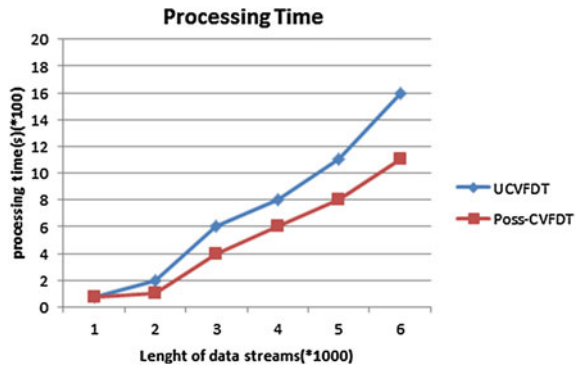


Fig. 5 Execution time



5 Conclusion

In this paper, we propose a new classification method adopted to an uncertain framework named the possibilistic CVFDT method based on the possibility theory and the classical CVFDT. The Poss-CVFDT aim to cope with objects described by possibility distributions. We proposed a new method to compute the measure of selection which called Specificity gain. In Fact, We have introduced a new sliding model based on samples's timestamp In order to improve the ability of Poss-CVFDT to cope with the concept drift issue. We have performed experimentations on UCI databases in order to evaluate the performance of our possibilistic CVFDT method and the UCVFDT approach.

References

1. Domingos, P., Hulten, G.: Mining high-speed data streams. In: Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD00), 7180 (2000)
2. Hulten, G., Spencer, L., Domingos, P.: Mining time-changing data streams. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD01), 97106 (2001)
3. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets Syst.* **1**, 328 (1978)
4. Higashi, M., Klir, G.J.: Measures of uncertainty and information based on possibility distributions. *Int. J. General Syst.* 4358 (1982)
5. Qin, B., Xia, Y., Li, F.: DTU: a decision tree for uncertain data. In: Advances in Knowledge Discovery and Data Mining. (PAKDD09), pp. 4–15 (2009)
6. Qin, B., Xia, Y., Prabhakar, S., Tu, Y.C.: A rule-based classification algorithm for uncertain data. In: IEEE International Conference on Data Engineering, USA. (ICDE09), pp. 1633–1640 (2009)
7. Tsang, S., Kao, B., Yip, K.Y., Ho, W.-S., Lee, S.D.: Decision trees for uncertain data. In: IEEE International Conference on Data Engineering 2009. (ICDE09), pp. 441–444 (2009)
8. Ge, J.A., Xia, Y., Nadungodage, C.H.: A neural network for uncertain data classification. Proceedings of the 14th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Hyderabad, India, pp. 449–460 (2010)
9. Pan, S., Wu, k., Zhang, Y., Li, X.: Classifier ensemble for uncertain data stream classification. In: Proceedings of the 14th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Shenzhen, China, pp. 488–495 (2010)
10. Liang, C., Zhang, Y., Song, Q.: Decision tree for dynamic and uncertain data streams. *JMLR: Workshop Conf. Proc.* *13*, 209–224 (2010)
11. Ghanem, T.M., Hammad, A.M., Mokbel, M.F., Aref, W.G., Elmagarmid, A.K.: Incremental evaluation of sliding-window queries over data streams. *IEEE Trans. Knowl. Data Eng.* **19**(1), 57–72 (2007)
12. Borgelt, C., Kruse, R.: Operations and evaluation measures for learning possibilistic graphical models. *Artif. Intell.* **148**, 385–418 (2003)
13. Borgelt, C., Gebhardt, J., Kruse, R.: Concepts for probabilistic and possibilistic induction of decision trees on real world data. In: Proceedings of the 4th European Congress on Intelligent Techniques and Soft Computing (EUFIT96), Aachen, Germany, vol. 3, Verlag Mainz, Aachen, pp. 1556–1560 (1996)
14. Hartley, R.V.L.: Transmission of information. *Bell Syst. Tech. J.* **7**, 535–563 (1928)

15. Higashi, M., Klir, G.J.: Measures of uncertainty and information based on possibility distributions. *Int. J. Gen. Syst.* **9**, 43–58 (1982)
16. Aggarwal, C.: Data streams: models and algorithms. *Advances in database systems*. (2007) Proceedings of the 24th International Conference on Data Engineering, pp. 150–159 (2008)
17. Noga, A., Yossi, M., Mario, S.: The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.* **58**(1), 137–147 (1999)