Mining Multiple Fuzzy Frequent Patterns with Compressed List Structures

Jerry Chun-Wei Lin[1,*], Jimmy Ming-Tai Wu[2], Youcef Djenouri[3], Gautam Srivastava[4], and Tzung-Pei Hong[5]
[1]*Western Norway University of Applied Sciences, Bergen, Norway*
[2]*Shandong University of Science & Technology, Shandong, China*
[3]*SINTEF Digital, Mathematics and Cybernetics, Oslo, Norway*
[4]*Brandon University, Brandon, Canada*
[5]*National University of Kaohsiung, Kaohsiung, Taiwan*
jerrylin@ieee.org; wmt@wmt35.idv.tw; youcef.djenouri@sintef.no; SRIVASTAVAG@brandonu.ca
tphong@nuk.edu.tw

***Abstract*—Fuzzy-set theory was invented to represent more meaningful representations of knowledge for human reasoning, which can also be applied and utilized for handling the quantitative database. In this paper, an efficient fuzzy mining (EFM) algorithm is presented to fast discover the multiple fuzzy frequent patterns from quantitative databases under type-2 fuzzy-set theory. A compressed fuzzy-list (CFL)-structure is developed to maintain complete information for rule generation. Two pruning techniques are developed to reduce the search space and speed up mining progress. Several experiments are carried out for the purpose of verifying the efficiency and effectiveness of the designed approach in terms of runtime and the number of examined nodes under different minimum support thresholds and the results indicated the designed EFM achieves the best performance compared to the existing models.**

***Index Terms*—fuzzy-set theory, fuzzy data mining, fuzzy-list structure, pruning strategies.**

I. Introduction

Pattern mining or called Knowledge Discovery in Databases (KDD) [1], [2], [4] has been treated as an important issue in many tasks since it can discover the potential and implicit information from the datasets, and the first fundamental algorithm is called Apriori [1], which is used to find associations of the item(sets) in the databases. Since the Apriori is a level-wise approach, which needs higher computational costs to first generate the candidates then evaluates them level-by-level, an improved algorithm called FP-growth [18] was implemented to improve mining efficiency by compressing the relevant transactions into a tree structure (called FP-tree). For the most works regarding ARM, they mostly focus on mining the FIs or ARs from binary databases, which only considers whether an item(set) appears in the databases. The other import factors such as interestingness, weight, importantness, and quantity are not considered as the major factors in ARM. In real-life domains and applications, an item can be purchased with several amounts in the shopping behaviors, for instance, five bottles of the beer or two boxes of the milk. It is thus not a trivial task to discover the knowledge and information from the quantitative databases. Fuzzy-set theory [15], [28], [40] was thus designed and used in many intelligent systems such as engineering fields, manufacturing, or medical diagnosis since the represented knowledge based on fuzzy-set is more interpretable for human reasoning. Furthermore, it can be used to convert the quantitative value of items into meaningful linguistic terms with the corresponding degrees, which is easier for managers and retails to make efficient decisions. Several algorithms were studied to handle the quantitative database based on the fuzzy-set theory for mining the fuzzy frequent itemsets. Many methods were respectively developed to mine fuzzy frequent patterns based on different structures and pruning strategies to reduce the computational cost [17], [26], [27], [29]. However, the above approaches only consider one linguistic term with the maximal scalar cardinality of an item, thus the discovered information may be incomplete for decision-making. Several algorithm considered the multiple fuzzy frequent itemsets (MFFIs) [20], [21], [30], [31] to derive more complete and sufficient knowledge. Based on this mechanism, more complete rules can be mined and the useful decisions can thus be produced.

The above methods mostly consider the type-1 fuzzy-set theory to discover the required information and knowledge, i.e., ARs or FIs. The algorithms used the conventional type-1 fuzzy-sets still, however, they treated the linguistic term with a discrete value. Mendel and John then designed the type-2 fuzzy-set theory [34] by involving the uncertain factor to mine the required information for decision-making. Chen et al. [8] integrated the type-2 fuzzy-sets model and considered the pattern mining problem to handle the quantitative database based on the level-wise approach. However, this approach still holds the single linguistic term of an item, thus the derived information may still be incomplete. Lin et al. [33] then developed a list-based method for efficiently mining type-2 fuzzy frequent patterns, which can speed up the mining performance compared to the level-wise approach. It does not, however, have the successful pruning methods to reduce the size of the search area; many unpromising candidates are still identified.

In this paper, we present a compressed fuzzy-list (CFL)-

*Corresponding author

978-1-7281-6932-3/20/$31.00 ©2020 IEEE

structure to keep more information for later mining progress. Two effective pruning strategies and an efficient mining (EFM) algorithm have been developed to mine the multiple fuzzy frequent patterns (MFFPs). Experiments are then conducted to show that the designed approach outperforms the level-wise-like and conventional list-based approaches in terms of runtime and number of examined candidates.

II. Literature Review

Association-rule mining (ARM) [1], [2], [4] is a basic methodology for knowledge discovery, which shows the relationships among the itemsets in binary databases. The first algorithm is named Apriori [2] that uses the level-wise approach to discover the numerous association rules (ARs). This approach is progressed by a level-wise approach, thus the computational cost is very high to produce ARs. To solve the limitation of Apriori, the FP-growth [18] was presented to speed up mining performance. Several extensions of frequent itemsets mining (FIM) are then further studied and developed in many different applications and domains [10], [11], [13], [25], and most of them focus on mining the required information from the binary database. In realistic situations, an item may, however, be purchased with several quantities in a transaction [12], [14], [38]. It is thus a non-trivial task to retrieve the information from the quantitative databases since the downward closure (DC) property is required to be maintained for ensuring the correctness and completeness of the discovered knowledge.

In the past decades, the fuzzy-set theory [15], [40] has been utilized in many applications and domains since its interpretable for human reasoning. Srikant et al. [35] introduced the approach for defining ARs by partitioning and transforming the problem into a binary database. Au and Chan [3] designed the F-APACS to mine fuzzy ARs (FARs) by employing linguistic terms to identify the discovered regularities and exceptions. Kuok et al. [23] developed an algorithm to process the quantitative attributes and concluded that fuzzy sets has better capability to handle the numerical values than existing methods. Hong et al. [17] implemented a fuzzy mining algorithm to mine the fuzzy rules that replies on the generate-and-test approach for handling the quantitative databases then proposed a GDF approach [20]. The GDF uses the gradual concept to mine the multiple fuzzy frequent itemsets (MFFIs) that also reduces the size of the processed database gradually; the computational cost can thus be reduced since some unpromising linguistic terms can be also deducted together in the mining progress. Chen et al. [7] developed a fusion model to improve the mining progress of multi-level fuzzy association rules based on a cumulative distribution of probabilities. Watanabe and Fujioka [37] have established the redundancy equivalence and theorems for FARs. Chang et al. [6] developed an ISPFTI algorithm by adopting the fuzzy-set theory to mine sequential patterns (frequent sequences) within fuzzy time intervals. Several algorithms based on the fuzzy-set theory for mining the required information in different applications and domains were then studied and developed in progress [5], [16], [24], [32], [36], [39].

To speed up the generate-and-test methodology for mining the fuzzy frequent itemsets (FFIs), Lin et al. then developed the fuzzy frequent pattern tree (FFP)-tree algorithm [26] to compress the fuzzy 1-itemsets into a tree structure for later mining process. This approach has produced a loose tree structure, thus a compressed fuzzy frequent pattern tree (CFFP)-tree algorithm [27] was proposed to reduce the size of the tree nodes. However, CFFP-tree approach still needs the extra memory usage for the attached array; it sometimes has the memory leakage problem. To solve this problem, the upper-bound fuzzy frequent pattern tree (UBFFP)-tree algorithm [29] was designed to keep a more condense tree structure, thus reducing the memory leakage problem for handling the big datasets.

The above works only reply to the type-1 fuzzy-set theory that does not take the uncertainty into account. The membership functions of type-1 fuzzy-set theory are entirely crisp that is inadequate in realistic applications to manage uncertainty models. To better present, the discovered knowledge with the uncertainty, type-2 fuzzy-set theory [19], [22], [34] was then proposed and developed. To incorporate the type-2 fuzzy-sets with pattern mining, Chen et al. [8] first developed the conventional level-wise (or Apriori-like) approach to mine the fuzzy type-2 frequent patterns level-wisely. This approach requires, however, to generate the amounts of candidates with high time complexity, which is not efficient for the mining task. Also, it uses the maximal scalar cardinality approach to retrieve only a single linguistic term of an item, which may produce insufficient knowledge for decision-making. Lin et al. then presented a list-based approach [33] to maintain the complete information for the mining progress. However, without the efficient pruning strategies and the loose upper-bound value on the unpromising patterns, this approach still has to examine many candidates for deriving the actual fuzzy frequent patterns.

III. Preliminary and Definition

Assume that I is considered as a finite set with m distinct items in the database D such that $I = \{i_1, i_2, \ldots, i_m\}$. The database with quantitative values of the items is considered as D, in which D has n transactions such that $D = \{T_1, T_2, \ldots, T_n\}$, and each T_q contains a unique identifier called the *TID*. Each item i_j in T_q has its purchase amount, which is denoted as: $q(i_j, T_q)$. A k-itemset is denoted as X such that $X = \{i_1, i_2, \ldots, i_k\}$, in which each i_j is the distinct item in X, $X \subseteq I$ and $X \subseteq T_q$. A membership functions used in type-2 fuzzy-set theory is denoted as μ. A threshold is denoted as δ, which is used as the minimum support to verify whether an itemset is considered as the fuzzy frequent pattern. A simple example is illustrated in Table I, which consists of ten transactions and six distinct items, denoted from a to f. Suppose that the minimum support threshold in Table I is set as δ (= 20%), and the type-2 fuzzy-sets used in the example are illustrated in Fig. 1. Here, three linguistic terms called

$Low(L)$, $Middle(M)$ and $High(H)$ are used in the μ. Note that the user can specify the number of the linguistic terms based on different domains, requirements, or applications.

TABLE I: An illustrated quantitative database.

TID	Items with the purchase amounts
T_1	*a*:5, *c*:4, *e*:1
T_2	*a*:3, *e*:1
T_3	*a*:1, *e*:2, *f*:2
T_4	*b*:2, *c*:1, *e*:3
T_5	*a*:4, *b*:5, *c*:5, *d*:3, *e*:3
T_6	*b*:4, *d*:1, *e*:4
T_7	*c*:4, *e*:2
T_8	*b*:4, *e*:4, *f*:3
T_9	*b*:3, *c*:4, *e*:2, *f*:1
T_{10}	*e*:5, *f*:5

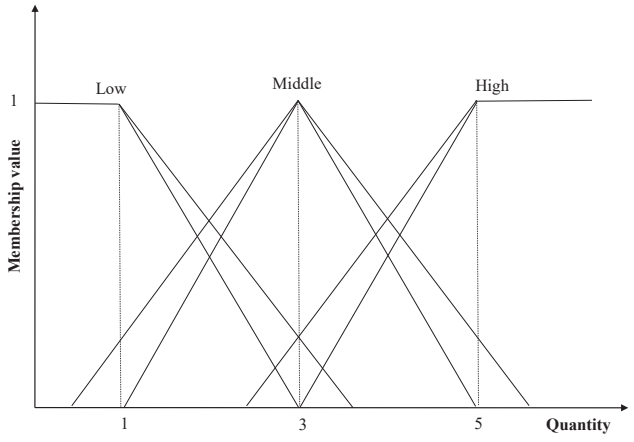

Fig. 1: A membership functions with three linguistic terms.

Definition 1: The *i* is an attribute (item) in the database such that $i \subseteq I$, which is also treated as the linguistic variable and its value is the set of fuzzy terms represented as the natural language such that $R_{i1}, R_{i2}, \ldots, R_{ih}$. This fuzzy terms can be transformed by the pre-defined μ (membership functions).

Definition 2: The $v_{i_{T_q}}$ is represented as the quantitative value of i, which shows the quantitative of the item (linguistic variable) i in a transaction T_q.

Definition 3: The $f_{i_{T_q}}$ is considered as the set of fuzzy linguistic terms with their membership degrees (fuzzy values) that was transformed from the quantitative value $v_{i_{T_q}}$ of the linguistic variable i by μ as:

$$f_{i_{T_q}} = \mu_i(v_{i_{T_q}})(= \frac{(fv^{lower}_{i_{T_q}1}, fv^{upper}_{i_{T_q}1})}{R_{i1}} + \cdots + \frac{(fv^{lower}_{i_{T_q}h}, fv^{upper}_{i_{T_q}h})}{R_{ih}}), \tag{1}$$

in which *h* represents as the number of fuzzy terms of *i* transformed by μ, R_{il} shows the *l*-th fuzzy terms of *i*, $fv^{lower}_{i_{T_q}l}$ indicates the lower membership degree (fuzzy value) of $v_{i_{T_q}}$ for *i* in the *l*-th fuzzy terms R_{il}, $fv^{upper}_{i_{T_q}l}$ states the upper membership degree (fuzzy value) of $v_{i_{T_q}}$ for *i* in the *l*-th fuzzy terms R_{il}, $fv^{lower}_{i_{T_q}l} \leq fv^{upper}_{i_{T_q}l}$, and $fv^{lower}_{i_{T_q}l}, fv^{upper}_{i_{T_q}l} \subseteq [0, 1]$.

For the given example in Table I, each transaction in the database is then transformed by the membership functions of Fig. 1. The final results after transformation are shown in Table II.

In 2016, Lin et al. [33] developed a list-based structure to mine the multiple fuzzy frequent patterns based on the type-2 fuzzy-set. However, it does not provide efficient pruning strategies to reduce the size of the search space, thus many unpromising candidates are still examined. Also, the upper-bound values on the candidates are over-estimated. In this paper, to efficiently mine the required multiple fuzzy frequent patterns from the database by considering the membership functions of type-2 fuzzy-set theory, an efficient structure is further developed to keep the complete information, and the efficient pruning strategies should be designed to reduce the size of the search space, thus improving the pattern mining performance.

IV. Proposed Efficient Fuzzy Mining Model

The purchase amount of each item in the database *D* is first transformed as the set of fuzzy linguistic terms with their fuzzy interval values by the pre-defined membership functions. For instance, the original database shown in Table I was then transformed using the membership functions of type-2 fuzzy-set shown in Fig. 1. After that, the results are stated in Table II. Since it is not a trivial task to elaborate the interval fuzzy value in the mining progress, the centroid type-reduction method [8] is then applied to reduce the complexity for mining MFFPs of the interval values. Definition is stated as follows.

Definition 4: The membership degree of a linguistic term R_{il} in a transformed database D' is denoted as $fv^c_{i_{T_q}l}$, and defines as:

$$fv^c_{i_{T_q}l} = \frac{fv^{lower}_{i_{T_q}l} + fv^{upper}_{i_{T_q}l}}{2}. \tag{2}$$

To evaluate whether a pattern is a MFFP, the scalar cardinality of each linguistic term is then summed up for the evaluation. The definition is then given below.

Definition 5: The scalar cardinality of each linguistic term is the summed up value of the transformed membership degrees and can be represented as the support value of a linguistic term as:

$$Sup(R_{jl}) = \sum_{R_{jl} \subseteq T_q \wedge T_q \in D'} fmv^c_{iql}, \tag{3}$$

To discover the complete information of MFFPs, the multiple linguistic terms of an item (set) is considered in the derived knowledge. The strategy called *MultiTerm* is then adopted here to keep the complete information for later mining progress of the developed EFM, which is described below.

Strategy 1 (Multiple terms with scalar cardinality, MultiTerm): To mine more and complete information, each linguistic term R_{in} of an item i, whose scalar cardinality (*Sup*) is no less the predefined minimum support count ($minSup \times |D|$) is considered to be represented of the item. Thus, each linguistic variable may have at least one represented fuzzy term with its membership degree (fuzzy value).

To maintain the downward closure property for building the compressed fuzzy-list (CFL)-structure, the linguistic terms in

TABLE II: Transformed database from Table I.

TID	Transformed fuzzy linguistic terms
T_1	$\frac{(0,0.25)}{a.M}+\frac{(1,1)}{a.H}$, $\frac{(0.5,0.63)}{c.M}+\frac{(0.5,0.63)}{c.H}$, $\frac{(1,1)}{e.L}+\frac{(0,0.25)}{e.M}$
T_2	$\frac{(0,0.25)}{a.L}+\frac{(1,1)}{a.M}+\frac{(0,0.25)}{a.H}$, $\frac{(1,1)}{e.L}+\frac{(0,0.25)}{e.M}$
T_3	$\frac{(1,1)}{a.L}+\frac{(0,0.25)}{a.M}$, $\frac{(0.5,0.63)}{e.L}+\frac{(0.5,0.63)}{e.M}$, $\frac{(0.5,0.63)}{f.L}+\frac{(0.5,0.63)}{f.M}$
T_4	$\frac{(0.5,0.63)}{a.M}+\frac{(0.5,0.63)}{a.H}$, $\frac{(0,0.25)}{b.M}+\frac{(1,1)}{b.H}$, $\frac{(1,1)}{c.L}+\frac{(0,0.25)}{c.M}$, $\frac{(0,0.25)}{e.L}+\frac{(1,1)}{e.M}+\frac{(0,0.25)}{e.H}$
T_5	$\frac{(0.5,0.63)}{a.M}+\frac{(0.5,0.63)}{a.H}$, $\frac{(0,0.25)}{b.M}+\frac{(1,1)}{b.H}$, $\frac{(0,0.25)}{c.M}+\frac{(1,1)}{c.H}$, $\frac{(0,0.25)}{e.M}+\frac{(1,1)}{d.H}$, $\frac{(0,0.25)}{e.L}+\frac{(1,1)}{e.M}+\frac{(0,0.25)}{e.H}$
T_6	$\frac{(0.67,0.75)}{b.M}+\frac{(0.33,0.5)}{b.H}$, $\frac{(1,1)}{d.L}+\frac{(0,0.25)}{d.M}$, $\frac{(0.5,0.63)}{e.M}+\frac{(0.5,0.63)}{e.H}$
T_7	$\frac{(0.5,0.63)}{e.L}+\frac{(0.5,0.63)}{e.M}$, $\frac{(0.5,0.63)}{c.M}+\frac{(0.5,0.63)}{c.H}$
T8	$\frac{(0.67,0.75)}{b.M}+\frac{(0.33,0.5)}{b.H}$, $\frac{(0.5,0.63)}{e.M}+\frac{(0.5,0.63)}{e.H}$, $\frac{(0,0.25)}{f.L}+\frac{(1,1)}{f.M}+\frac{(0,0.25)}{f.H}$
T_9	$\frac{(0.33,0.5)}{b.L}+\frac{(0.67,0.75)}{b.M}$, $\frac{(0.5,0.63)}{c.M}+\frac{(0.5,0.63)}{c.H}$, $\frac{(0.5,0.63)}{e.L}+\frac{(0.5,0.63)}{e.M}$, $\frac{(1,1)}{f.L}+\frac{(0,0.25)}{f.M}$
T_{10}	$\frac{(0,0.25)}{e.M}+\frac{(1,1)}{e.H}$, $\frac{(0,0.25)}{f.M}+\frac{(1,1)}{f.H}$

the transactions are sorted in ascending order by *ASCorder* strategy, which is described below.

Strategy 2 (Sort in ascending order, ASCorder): Each linguistic term of transactions in the transformed database D' is then sorted in ascending order of their support value, and denoted as $\prec$ which can be used for later processing of CFL-structure construction phase.

The revised and sorted transactions are indicated in Table III.

TABLE III: The sorted database.

TID	Linguistic terms
T_1	$\frac{0.56}{c.H}, \frac{1}{e.L}, \frac{0.13}{e.M}$
T_2	$\frac{1}{e.L}, \frac{0.13}{e.M}$
T_3	$\frac{0.56}{e.L}, \frac{0.56}{e.M}$
T_4	$\frac{0.13}{b.M}, \frac{0.13}{e.H}, \frac{0.13}{e.L}, \frac{1}{e.M}$
T_5	$\frac{0.13}{b.M}, \frac{0.13}{e.H}, \frac{1}{c.H}, \frac{0.13}{e.L}, \frac{1}{e.M}$
T_6	$\frac{0.71}{b.M}, \frac{0.56}{e.H}, \frac{0.56}{e.M}$
T_7	$\frac{0.56}{c.H}, \frac{0.56}{e.L}, \frac{0.56}{e.M}$
T_8	$\frac{0.71}{b.M}, \frac{0.56}{e.H}, \frac{0.56}{e.M}$
T_9	$\frac{0.71}{b.M}, \frac{0.56}{c.H}, \frac{0.56}{e.L}, \frac{0.56}{e.M}$
T_{10}	$\frac{1}{e.H}, \frac{0.13}{e.M}$

After the original database is revised and sorted, the algorithm is processed to construct the CFL-structure. Each remaining 1-itemset is used to construct its relevant CFL-structure for maintaining the complete information. Properties of the CFL-structure are given below.

Definition 6: Assume that *X* is considered as the set of the linguistic terms and *T* is set as a transaction such that $X \subseteq T$. Thus, the remaining set for all linguistic terms in *T* after *X* is denoted as T/X.

The definition of the developed CFL-structure is then described below.

Definition 7: Each element in the CFL-structure of X has three attributes (ordered) as: tid, fmv, and $rmrfv$.

- *tid* shows that the term *X* is in a transaction *T*.
- *fmv* shows the fuzzy membership value of *X* in a transaction *T*.
- *rmrfv* shows the relative maximum remaining fuzzy membership value after *X* in a transaction*T*, which is the minimum value between $rmrfv(X,T)$ and $fmv(X,T)$.

Here, Sup is defined as the sum up value of fmv in the CFL-list structure, and $rSup$ is the sum up value of $rmrfv$ in the CFL-list structure. From the above definition, the new developed CFL-structure is shown in Fig. 2. For instance in Fig. 2, the fuzzy term $\{b.M\}$ appears in transactions T_4, T_5, T_6, T_8, and T_9, and its elements are (4, 0.13, 0.13), (5, 0.13, 0.13), (6, 0.71, 0.56), (8, 0.71, 0.56) and (9, 0.71, 0.56), respectively. The Sup and $rSup$ are 0.239 and 0.194. In this example, the Sup is greater than the $minSup$ (= 0.2) that means the $\{b.M\}$ is considered as the MFFP. However, since its $rSup$ is less than 0.2, it is not necessary to explore the extensions of $\{b.M\}$; the size of the search space can thus be greatly deducted. The construction algorithm of the CFL-structure is then stated in Algorithm 1.

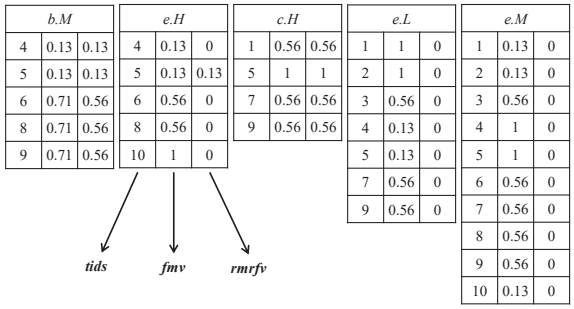

Fig. 2: A built CFL-structure.

After a CFLs-structures being generated, a pruning strategy will be taken to reduce the space searching, which uses the $Supt$ and $rSup$ of such a list X to decide whether to search the extension of X. The strategy is described as Lemma 1.

Lemma 1: For an termset X, if $Sup(X)$ or $rSup(X)$ is less than the minimum support threshold, then any supersets (extension) of X is not multiple fuzzy frequent pattern and should be pruned.

The proof of Lemma 1 is shown below.

Proof 1: $\forall$ transaction $T \supseteq X'$,
$\because X'$ is an extension of X, $(X' - X) = (X'/X)$, we can obtain that $X \subseteq X' \subseteq T \Rightarrow (X'/X) \subseteq (T/X)$,
$\therefore fmv(X',T) = fmv(X,T) \cup fmv((X' - X),T) = min(fmv(X,T), fmv(X'/X,T)) \leq fmv(X,T)$ and $min(fmv(X,T), fmv(X'/X,T)) \leq fmv(X'/X,T) =$

Algorithm 1: Construction of the 1-pattern CFL-structure.

Input: D', a revised and sorted dataset.
Output: the CFLs-structures for 1-patterns and large 1-patterns L'.

for *each linguistic term* t_{jn} *of item* j **do**
  if $Sup(t_{jn}) \geq minSup$ **then**
    put t_{jn} into L', and keep L' as Sup-ascending order;
for *each linguistic term* t_{jn} *of* L' *in each* T *of* D' **do**
  add element (tid, fmv of t_{jn} in T, $rmrfv$ of t_{jn} in T) to t_{jn}-CFL-structure;
  $CFLs = CFLs \bigcup t_{jn}$-CFL-structure;
return L', constructed *CFLs* ;

$rmrfv(X,T)$.

Suppose that $X.tids$ denotes the set of *tids* of X,
$\because X \subseteq X' \Rightarrow X'.tids \subseteq X.tids$,
$\therefore \frac{\sum_{id(T)\in X'.tids} fmv(X',T)}{N} \leq \frac{\sum_{id(T)\in X.tids} fmv(X,T)}{N} \Rightarrow Sup(X) < minSup.$

Furthermore, we can obtain that $\frac{\sum_{id(T)\in X'.tids} rmrfv(X',T)}{N} \leq \frac{\sum_{id(T)\in X.tids} rmrfv(X,T)}{N} \Rightarrow rSup(X) < minSup.$

From the given example, the search space for mining the required MFFPs is based on the enumeration tree, which is shown in Fig. 3.

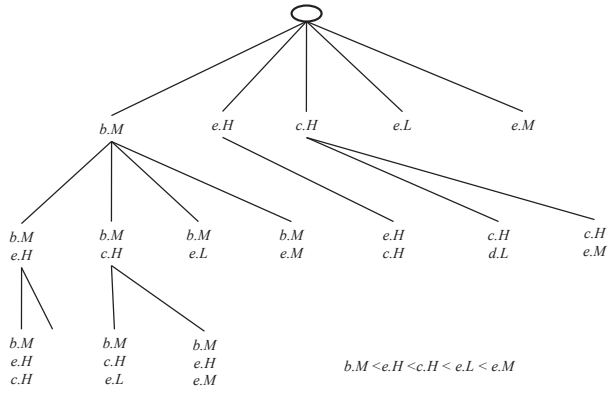

Fig. 3: The size of search space in the running example.

To perform and generate the k-itemsets($k \geq 2$), the terms of P_x and P_y are used to generate the CFL-structure, forming as P_{xy}. The fuzzy terms are first examined to determine whether the valid $P_{xy}.CFL$ is generated. If P_x and P_y appear in the same transactions (TIDs), the simple join operation is then performed to calculate the fmv of each transaction T. Furthermore, the minimum operation is also adopted to find the remaining $rmrfv$ of the P_{xy} in T. This process is then described below.

- $E_{xy}.tid$=$E_x.tid$(or $E_y.tid$).
- $E_{xy}.fmv$=$min(E_x.tid, E_y.tid)$.
- $E_{xy}.rmrfv$ = $min(E_x.rmrfv, E_y.rmrfv)$.

Here, we can note that if the sum of fmv is no larger than the pre-defined minimum support count, it is not considered as the MFFP and the supersets will be discarded and ignored, directly without any further exploration. This progress is then executed recursively until no candidates can be generated.

After the CFL-structure is generated, we then present another pruning strategy to reduce the size of the search space by using the Sup and $rSup$ of such a list X to decide whether to search the extension of X. The strategy is described as Lemma 2.

Lemma 2: For a termset X, if $Sup(X)$ or relative remaining support $rSup(X)$ is less than the minimum support threshold, then any supersets (extension) of X is not a F2FP and should be discarded.

Proof 2: $\because X \subseteq X' \Rightarrow X'.tids \subseteq X.tids$,
$\therefore Sup(X') = \frac{sum_{id(T)\in X.tids} fmv(X',T)}{N} = \frac{\sum_{id(T)\in X'.tids} min(fmv(X,T), fmv(X'/X,T)}{N} \leq \frac{sum_{if(T)\in X'.tids} min(fmv(X,T), rmfv(X,T)}{N} = \frac{\sum_{id(T)\in Q'} fmv(X,T) + \sum_{id(T)\in Q''} rfmv(X,T)}{N} = rSup(X) \leq minSup.$

Note that suppose $Q' \cup Q'' = X'.tids$ and $Q' \cap Q'' = , T \in Q', fmv(X,T) < rmfv(X,T)$, and $T \in Q', fmv(X,T) \geq rmfv(X,T)$.

Algorithm 2: Developed EFM algorithm.

Input: $CFLs$, the built CFL-structure.
Output: $MFFPs$, the set of multiple fuzzyfrequent patterns.

for *each list* X *in* $CFLs$ **do**
  if $Sup(X) \geq minSup$ **then**
    add items of X into $MFFPs$;
    if $rSup(X) \geq minSup$ **then**
      $exCFLs \leftarrow null$;
      for *each CFL-structure* Y *after* X*in* $CFLs$ **do**
        $exCFLs \leftarrow exCFLs + Constrcut(X,Y)$;
      EFM($exCFLs$);
return $F2FPs$.

The developed EFM algorithm is then shown in Algorithm 2. First, the algorithm begins with the initially constructed CFL-structures, and for each termset (such as X), the $Sup(X)$ is firstly compared with the $minSup$ to examine whether X is frequent. After that, the relative remaining support value of X, called $rSup(X)$, is then utilized to decide whether the extensions of X should be explored. After the extensions of the termset X is constructed, the algorithm

is processed again for next *k*-itemsets until all the required MFFPs are determined.

V. Experimental Evaluation

In this section, the developed EFM is then performed compared to the level-wise algorithm [8] and list-based approach [33] in several datasets. The algorithms were implemented in JAVA language, performing on a PC with Intel Core i5-3470 @ 3.20GHz and 4 GB main RAM. All the implemented algorithms are performed on 32-bit Microsoft Windows 7 operating system. Three real-life [9] chess, mushroom and foodmart were conducted for the experiments. The characteristics of the conducted datasets are shown in Table IV.

TABLE IV: Characteristics of used datasets.

Dataset	#|D|	#|I|	AvgLen	MaxLen	Type
chess	3196	75	37	37	dense
mushroom	8,124	119	23	23	dense
foodmart	21,556	1559	4	11	sparse

The purchase amount of each item in the quantitative database is first transformed according to the defined type-2 membership functions. In the experiments, the linguistic 2-terms shown in Fig. 4 is used to show the performance of the designed model. Note that the linguistic terms can be defined by the user's preference.

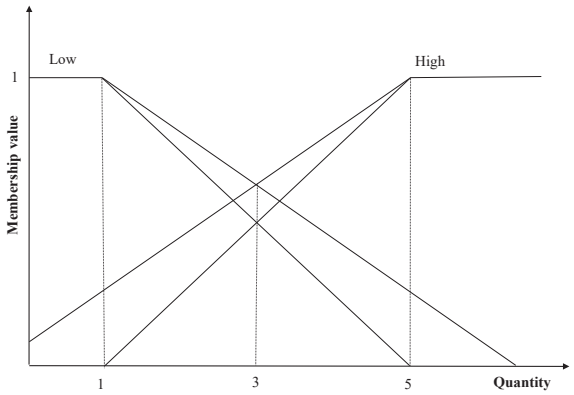

Fig. 4: The membership function of linguistic 2-terms.

A. Execution time

The execution time of the compared algorithms for 2-terms membership functions is first illustrated in Fig. 5 at different minimum support thresholds. It can be seen from the above results that the developed EFM algorithm has better execution time than the conventional level-wise and the state-of-the-art list-based algorithm for mining MFFPs with fuzzy linguistic 2-terms for all conducted datasets. From the observation of the above results, it can be seen that the execution time decreases along with the increase of the minimum support threshold. This is acceptable since as the increasing of minimum support threshold, the number of MFFPs decreases since fewer patterns satisfy the condition with a higher threshold. From the results, we can thus observe that the designed EFM needs fewer computations than the compared approaches.

B. Number of examined nodes

In this section, the number of examined nodes in the search space of the enumeration tree for the three compared algorithms are then determined. Results under linguistic 2-terms membership functions are then stated in Fig. 6. It can be easily observed that the designed EFM has generated fewer nodes for examination in the search space compared to the other two approaches. Thanks to the advantage of the designed two pruning strategies, they are effective to reduce some unpromising candidates for examination in the search space of the MFFPs.

VI. Conclusions and Future Works

In this paper, an efficient fuzzy mining (EFM) algorithm is presented to discover the set of multiple fuzzy frequent patterns (MFFPs) based on the type-2 fuzzy-set theory. A compressed fuzzy-list (CFL) is also maintained for storing the satisfied fuzzy frequent itemsets that reduces the conventional limitation of multiple database scans. Two effective pruning strategies are also designed to early reduce the unpromising candidates, thus the search space to find the required MFFPs can be deducted. Experiments are then performed to conduct six datasets regarding varied minimum thresholds to verify the performance of the designed EFM method compared to the previous two works in terms of execution time and the number of examined nodes of the search space. Furthermore, we will then explore the more condense structure and tighter upper-bound values on the patterns to speed up mining efficiency. It is also a big challenge to maintain sufficient information for incremental mining in the dynamic database or efficiently synthesizing the discovered knowledge (i.e., MFFPs) from different branches.

References

[1] R. Agrawal, T. Imieliński, and A. Swami, *Mining association rules between sets of items in large databases*, ACM SIGMOD Record, 1993, pp. 207–216.

[2] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," The International Conference on Very Large Databases, pp. 487–499, 1994.

[3] W. H. Au and K. C.C. Chan, "An effective algorithm for discovering fuzzy rules in relational databases," IEEE International Conference on Fuzzy Systems, pp. 1314–1319, 1998.

[4] M. S. Chen, J. Han, and P. S. Yu, "Data mining: An overview from a database perspective," IEEE Transactions on Knowledge and Data Engineering, vol. 6, pp. 866–883, 1996.

[5] C. Li, B. Yan, M. Tang, J. Yi, and X. Zhang, "Data driven hybrid fuzzy model for short-term traffic flow prediction," Journal of Intelligent & Fuzzy Systems, vol. 35, pp. 6525–6536, 2018.

[6] C. I. Chang, H. E. Chueh, and Y. C. Luo, *An integrated sequential patterns mining with fuzzy time-intervals*, The International Conference on Systems and Informatics, 2012, pp. 2294–2298.

[7] J. S. Chen, F. G. Chen, and J. Y Wang, "Enhance the multi-level fuzzy association rules based on cumulative probability distribution approach," The ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp. 89–94, 2012.

[8] C. H. Chen, T. P. Hong, and Y. Li, "Fuzzy association rule mining with type-2 membership functions," Lecture Notes in Computer Science, pp. 128–134, 2015.

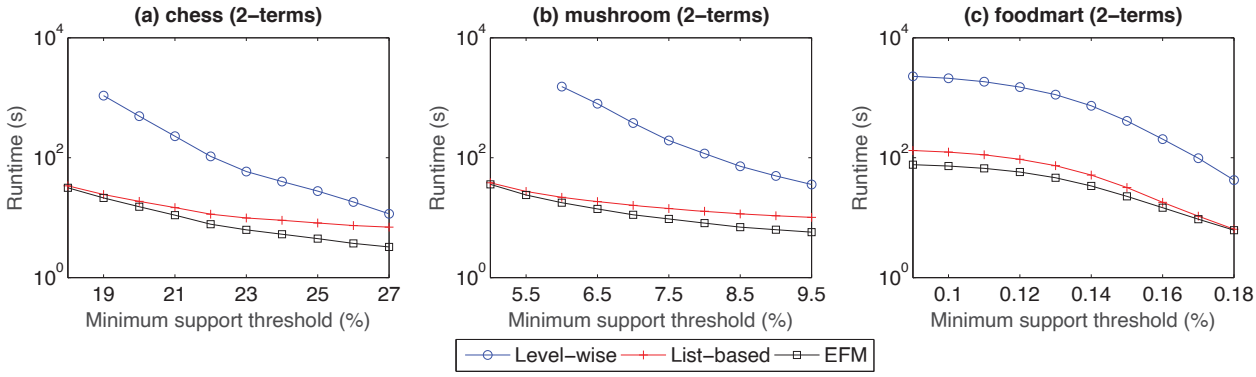

Fig. 5: Execution time comparisons with 2-terms membership functions.

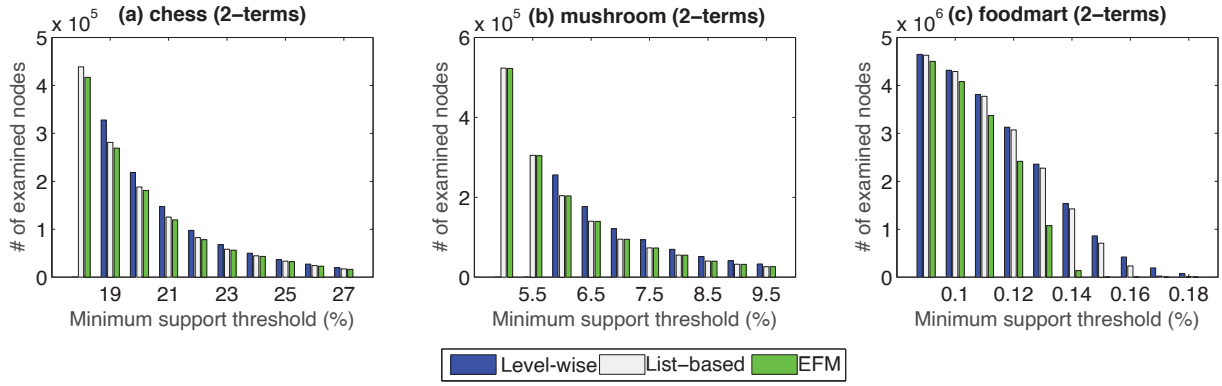

Fig. 6: Comparisons for the number of nodes under linguistic 2-terms membership functions.

[9] P. Fournier-Viger, C. W. Lin, A. Gomariz, T. Gueniche, A. Soltani, Z. Deng, and H. T. Lam, *The SPMF open-source data mining library version 2*, The European Conference on Principles of Data Mining and Knowledge Discovery, 2016, pp. 36–40.

[10] P. Fournier-Viger, C. W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas, "A survey of sequential pattern mining," Data Science and Pattern Recognition, vol. 1, pp. 54–77, 2017.

[11] W. Gan, J. C. W. Lin, P. Fournier-Viger, H. C. Chao, T. P. Hong, and H. Fujita, "A survey of incremental high-utility itemset mining," Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 8, e1242, 2018.

[12] W. Gan, J. C. W. Lin, P. Fournier-Viger, H. C. Chao, and P. S. Yu, "HUOPM: High-utility occupancy pattern mining," IEEE Transactions on Cybernetics, pp. 1–14, 2019.

[13] W. Gan, J. C. W. Lin, P. Fournier-Viger, H. C. Chao, and P. S. Yu, "A survey of parallel sequential pattern mining," ACM Transactions on Knowledge Discovery from Data, vol. 13, Article 25, 2019.

[14] W. Gan, J. C. W. Lin, P. Fournier-Viger, H. C. Chao, V. S. Tseng, and P. S. Yu, "A survey of utility-oriented pattern mining," IEEE Transactions on Knowledge and Data Engineering, 2019.

[15] J. Holland, "Adaptation in natural and artificial systems," Cambridge, MA: MIT Press, 1975.

[16] J. Han and Y. Fu, "Discovery of multiple-level association rules from large databases," The International Conference on Very Large Data Bases, pp. 420–431, 1995.

[17] T. P. Hong, C. S. Kuo, and S. C. Chi, "Mining association rules from quantitative data," Intelligent Data Analysis, vol. 3, pp. 363–376, 1999.

[18] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent Patterns without candidate generation: a frequent-pattern tree approach,", Data Mining & Knowledge Discovery, vol. 8, 53–87, 2004.

[19] H. Hagras, *Type-2 fuzzy logic controllers: A way forward for fuzzy systems in real world environments*, Lecture Notes in Computer Science, 2008, pp. 181–200.

[20] T. P. Hong, G. C. Lan, Y. H. Lin, and S. T. Pan, "An effective gradual data-reduction strategy for fuzzy itemset mining," International Journal of Fuzzy Systems, vol. 15(2), pp. 170–181, 2013.

[21] T. P. Hong, C. W. Lin, and T. C. Lin, "The MFFP-tree fuzzy mining algorithm to discover complete linguistci frequent itemsets," Computational Intelligence, vol. 30, pp. 145–166, 2014.

[22] N. N. Karnik and J. M. Mendel, "Introduction to type-2 fuzzy logic systems," International Conference on Fuzzy Systems, pp. 915–920, 1998.

[23] C. M. Kuok, A. Fu, and M. H. Wong, "Mining fuzzy association rules in databases," ACM SIGMOD Record, vol. 27, pp. 41–46, 1998.

[24] S. Kar and M. M. J. Kabir, "Comparative analysis of mining fuzzy association rule using genetic algorithm," The International Conference on Electrical, Computer and Communication Engineering, pp. 1–5, 2019

[25] C. W. Lin, T. P. Hong, and W. H. Lu, *The Pre-FUFP algorithm for incremental mining*, Expert Systems with Applications, **36** (2009), 9498–9505.

[26] C. W. Lin, T. P. Hong, and W. H. Lu, "Linguistic data mining with fuzzy FP-trees," Expert Systems with Applications, vol. 37, pp. 4560–4567, 2010.

[27] C. W. Lin, T. P. Hong, and W. H. Lu, "An efficient tree-based fuzzy data mining approach," International Journal of Fuzzy Systems, vol. 12, pp. 150–157, 2010.

[28] C. W. Lin and T. P Hong, *A survey of fuzzy web mining*, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, vol. 3, pp. 190–199, 2013.

[29] C. W. Lin and T. P. Hong, "Mining fuzzy frequent itemsets based on UBFFP trees," Journal of Intelligent and Fuzzy Systems, vol. 27, pp. 535–548, 2014.

[30] J. C. W. Lin, T. P. Hong, and T. C. Lin, "A CMFFP-tree algorithm to mine complete multiple fuzzy frequent itemsets," Applied Soft Computing, vol. 28, pp. 431–439, 2015.

[31] J. C. W. Lin, T. P. Hong, T. C. Lin, and S. T. Pan, "An UBMFFP tree for mining multiple fuzzy frequent itemsets," International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 23, pp. 861–879, 2015.

[32] J. C. W. Lin, T. Li, P. Fournier-Viger, and T. P. Hong, "A fast algorithm for mining fuzzy frequent itemsets," Journal of Intelligent & Fuzzy Systems, vol. 29, pp. 2373–2379, 2015.
[33] J. C. W. Lin, X. Lv, P. Fournier-Viger, T. Y. Wu, and T. P. Hong, "Efficient mining of fuzzy frequent itemsets with type-2 membership functions," The Asian Conference on Intelligent Information and Database Systems, pp. 191–200, 2016.
[34] J. M. Mendel, and R. I. B. John, "Type-2 fuzzy sets made simple," IEEE Transactions on Fuzzy Systems, vol. 10, pp. 117–127, 2002.
[35] R. Srikant and R. Agrawal, "Mining quantitative association rules in large relational tables," The SIGMOD International Conference on Management of Data, pp. 1–12, 1996.
[36] D. K. Srivastava, B. Roychoudhury, and H. V. Samalia, "Fuzzy association rule mining for economic development indicators," International Journal of Intelligent Enterprise, vol. 6(1), pp. 3–18, 2019.
[37] T. Watanabe and R. Fujioka, "Fuzzy association rules mining algorithm based on equivalence redundancy of items," IEEE International Conference on Systems, Man, and Cybernetics, pp. 1960–1965, 2012.
[38] J. M. T. Wu, J. C. W. Lin. and A. Tamrakar, "High-utility itemset mining with effective pruning strategies," ACM Transactions on Knowledge Discovery from Data, vol. 13, Article 58, 2019.
[39] L. Wang, Q. Ma, and J. Meng, "Incremental fuzzy association rule mining for classification and regression," IEEE Access, vol. 7, pp. 121095–121110, 2019.
[40] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338–353, 1965.