A Data-Sensitive Approach for Fuzzy Concept Extraction

Ebtesam El-Hossiny Shemis [1]*   **Ahmed Mohammed Gadallah [1]**   **Hesham Ahmed Hefny [1]**

[1]*Institute of statistical studies and research,*
Cairo University, 5 Ahmed Zewail Street - Orman - Giza, Egypt
*Corresponding author's Email: eptesam.elhossiny@pg.cu.edu.eg

Abstract: Nowadays, extracting fuzzy concepts using fuzzy formal concept analysis (FFCA) is an increasingly important process in many fields including data-mining, information retrieval, and Ontology construction. However, recent studies have heightened the need for more efficient approaches for extracting a reduced count of distinct fuzzy concepts in a reasonable time. This paper aims mainly to address such challenge. Generally, the proposed approach is composed of two main stages to handle quantitative data effectively. Firstly, a data-sensitive fuzzification stage maps the many-valued context to a more consistent fuzzy one. Secondly, an enhanced algorithm generates a reduced count of valuable fuzzy concepts and merges the more similar ones. Accordingly, the proposed approach can efficiently handle very large datasets where the process of extracting all fuzzy concepts is an intractable task. Surprisingly, the proposed approach reduces the overall processing time and complexity when compared with some other previous approaches.

Keywords: Fuzzy formal concept analysis, One-sided fuzzy concepts, Data-sensitive fuzzy set, Fuzzy fix-points, Knowledge extraction.

1. Introduction

Formal concept analysis (FCA) is a robust lattice theory branch that has been inspired in 1980's [1, 2] . It can be defined as "a field of applied mathematics based on concepts and concept hierarchies" [3, 4]. Amazingly, FCA has been successfully involved in many fields and hence many applications such as linguistics, information retrieval [5], economics and much more [6]. In fact, traditional FCA can only handle crisp formal contexts effectively. Such binary context represents a binary relationship between objects and attributes. In consequence, FCA depends mainly on the classical crisp set theory in which an object either belongs totally to a set or not belong at all.

Generally, FCA is used to generate a set of formal concepts, also known as fix-points [1, 2]. Afterward, it draws the corresponding conceptual hierarchy called formal concept lattice. Such concept lattice shows the parent/child relationships between the extracted formal concepts. Moreover,

FCA can further generate a set of attribute implications and associations. In consequence, it best suits association rule mining problem.

Ineffectively, traditional FCA uses a crisp scaling method to handle quantitative context by dividing the attribute domain to a set of disjoint intervals. Unfortunately, the crisp scaling method faces the problem of crisp boundaries and hence not suitable for human-like reasoning [3]. In contrary, fuzzy formal concept analysis (FFCA) has the ability to handle such quantitative contexts more effectively in a flexible human-like manner with the aid of fuzzy set theory. Accordingly, crisp boundaries problem is perfectly handled by the fuzzification process. In consequence, a multi-valued attribute is scaled to a set of linguistic labels with matching degrees in the range of [0, 1] [7].

Generally, the process of extracting the whole set of fuzzy concepts is a time-consuming problem, especially in case of large contexts. Such process is a #P-complete problem [8]. But fortunately, the complexity can be reduced if the relation between

the objects and the attributes is relatively small even with large objects/attributes sets [9].

This paper proposes a more efficient enhanced approach for extracting one-sided-fuzzy concepts from a multi-valued context. In the first stage, it benefits from using data-sensitive fuzzy terms in which a linguistic concept is data-sensitive and has a flexible definition. Moreover, in the second stage, it reduces the processing time associated with extracting the whole set of fuzzy concepts, regarding some other related algorithms. .So, the main contribution of this paper is to enhance fuzzy concept semantics in addition to extracting them in a more reasonable time.

The rest of this paper is organized as follows: Section 2 presents basic FCA notions and definitions. Moreover, FFCA notions and definitions are addressed in Section 3. Subsequently, Section 4 presents some FFCA related works. The proposed data-sensitive approach to generate one-sided fuzzy concepts is introduced in Section 5. Consequently, Section 6 shows the conducted experiments for evaluating the proposed algorithm. Eventually, conclusion and future works are introduced in Section 7.

2. FCA notions and definitions

This section presents some basic definitions and notions related to classical FCA, more definitions and notions are found in [1, 3, 4]. Generally, formal concept analysis seeks mainly to discover underlying clusters of objects and attributes in a specific dataset [10]. Mainly, FCA accepts the input binary formal context that represents a crisp relation I between objects G and their attributes M. In consequence, a formal context can be simply defined as a cross table of object rows, and column attributes (or vice versa). The intersected cell between each object g and attribute m has a cross symbol if $(g, m) \in I$.

Definition 1: A formal context is a triple $K= (G, M, I)$ that consists of objects G, columns M and a crisp relation between objects and columns $I \subseteq G \times M$ such that if $g \in G$ and $m \in M$ are in relation in I, then it will be represented by the pair $(g, m) \in I$ or $(g\ I\ m)$ and is read as "object g has attribute m".

Given that $A \subseteq G$ and $B \subseteq M$, the derivation of A and B is given by Eq. (1) and Eq. (2) respectively:

$$A \uparrow := \{ m \in M | (g\ I\ m) \,\forall\, g \in A\} \qquad (1)$$

Where $A\uparrow$ (denoted as A') is a set of common attributes in M shared by all objects in A.

$$B \downarrow := \{ g \in G | (g\ I\ m) \,\forall\, m \in B\} \qquad (2)$$

Where $B\downarrow$ is a set of all objects that have relationships with all attributes in the set B.

Definition 2: the pair (A, B) is said to be a formal concept of the formal context $K= (G, M, I)$ if $A \subseteq G$, $B \subseteq M$, $A' = B$ and $B' = A$. Usually, A and B refers to concept extent and intent respectively.

The unordered set of all concepts for context $K= (G, M, I)$ is given by Eq. (3).

$$\mathfrak{B}(G, M, I) = \{(A, B) | A' = B\ and\ B' = A\} \quad (3)$$

Definition 3: Let (A_1, B_1) and (A_2, B_2) be two formal concepts of the formal context $K= (G, M, I)$, then (A_1, B_1) is a sub-concept of (A_2, B_2) if $A_1 \subseteq A_2$ (i.e. $B_2 \subseteq B_1$). Such relation is denoted by $(A_1, B_1) \leq (A_2, B_2) \Longleftrightarrow (A_1 \subseteq A_2) \Longleftrightarrow (B_2 \subseteq B_1)$.

The set of all formal concepts of K (G, M, I) partially ordered by sub/super relation is called *concept lattice* and is denoted by $\underline{B}\ (G, M, I)$.

Classical FCA can only handle crisp binary context effectively which is not the usual case in real-world dataset fields. It suffers from the problem of crisp boundaries where it is difficult to set sharp borders among scaled attribute intervals.

Definition 4: A multi-valued context K=(G, M, V, I) is composed of a set G of objects, a set M of attributes, a set V of attribute values and a ternary-relation I between G, M, and V (i.e., $I \subseteq G \times M \times V$). An element $(g, m, v) \in I$ indicates that the attribute m has the value v for the object g.

3. FFCA notions and definitions

Generally, there exist numerous viewpoints of FFCA and hence various fundamental definitions set. Consequently, FFCA can be categorized, according to the way of involving the fuzzy set theory, into (a) Full-sided FFCA where both sides of fuzzy concepts are fuzzy sets (fuzzy extent and fuzzy intent). Such type of FFCA is extensively introduced in literature in the Bělohlàvek's series of papers [11-14]. (b) One-sided FFCA where only one side of fuzzy concepts is fuzzily defined while the other side is crisply defined (fuzzy extent and crisp intent) or (crisp extent and fuzzy intent). Such category of FFCA is presented in [5, 10, 15-18].

This section introduces basic definitions and notions regarding the one-sided FFCA with fuzzy extents and crisp intents concepts.

Definition 5: A fuzzy formal context K is a triple $(G, M, I= \varphi\ (G \times M))$ where G is a finite object set, M is a finite attribute set and I is a fuzzy set on the

196

domain $G \times M$ representing a relation between objects set and attributes set. Each relation $(g, m) \in I$ has a membership value μ $(g, m) \in [0, 1]$. A threshold interval $[t_1, t_2]$, where $0 \leq t_1 \leq t_2 \leq 1$, can be applied to the fuzzy context to eliminate each relation instance (g, m) whose membership value μ (g, m) is out of such threshold interval.

Effectively, fuzzification transforms the multi-valued context to the corresponding fuzzy one using suitable membership functions. Afterward, the resulting fuzzy context feds into the FFCA algorithm to generate all fuzzy concepts.

Definition 6: A fuzzy concept of the fuzzy context $K(G, M, I)$ is a pair $(A_F = \varphi(A), B)$ where $A_F \subseteq G$ is the fuzzy concept extent, $B \subseteq M$ is the crisp intent, $A' = B$ and $B' = A$.

Definitions of A' and B' vary according to two main points of view. Eq. (4) and Eq. (5) introduces the first viewpoint [16-18].

$$A' := \{b \in B \mid \forall a \in A: \mu_I(a, b) \geq \mu_A(a)\} \quad (4)$$

$$B' := \{\frac{a}{\mu_A(a)} \mid \mu_A(a) = min_{b \in B}(\mu_I(a, b))\} \quad (5)$$

Where A' is the crisp intent and B' is the fuzzy extent.

On the other hand, Eq. (6) and Eq. (7) present the second viewpoint that relies primarily on threshold interval $[t_1, t_2]$ [19-21].

$$A' = \{b \in M \mid \forall a \in A: t_1 \leq \mu_I(a, b) \leq t_2\} \quad (6)$$

$$B' = \{a \in G \mid t_1 \leq min_{b \in B}(\mu_I(a, b)) \leq t_2\} \quad (7)$$

Where $A' = B$ is the crisp intent and $B' = A$ is the fuzzy extent.

In this paper, the proposed algorithm relies on Eqs. (6) and (7). Such equations set usually generates less number of fuzzy concepts than Eqs. (4) and (5).

Definition 7: Let $(\varphi(A_1), B_1)$ and $(\varphi(A_2), B_2)$ be two fuzzy concepts of the fuzzy context $K(G, M, I)$. The fuzzy concept $(\varphi(A_1), B_1)$ is said to be a sub-concept of $(\varphi(A_2), B_2)$, denoted by $(\varphi(A_1), B_1) \leq (\varphi(A_2), B_2)$, if $\varphi(A_1) \subseteq \varphi(A_2) \Leftrightarrow B_2 \subseteq B_1$.

A fuzzy concept lattice of the fuzzy formal context $K(G, M, I)$ is defined as the set of all fuzzy formal concepts ordered by the child/parent concept relation (partial order $\leq$).

4. FFCA related works

In general, fuzzy concepts generation algorithms can be classified according to the type of generated fuzzy concepts. Consequently, such algorithms can be categorized into three main categories (a) crisply-generated fuzzy concepts. (b) full-sided fuzzy concepts and (c) one-sided fuzzy concepts.

Algorithms that generate fuzzy concepts crisply aim mainly to convert the fuzzy context into an isomorphic crisp one using some transformation techniques. Afterward, any classical FCA algorithm can be used to generate the whole set of fuzzy concepts. Moreover, a post-processing step is usually required to eliminate redundant concepts and generate a reliable complete set of fuzzy concepts. Such approaches can be found in [3, 22].

Usually, the number of crisply generated fuzzy concepts is relatively less than the number of all full-sided-fuzzy concepts. Although transforming fuzzy context into binary one enables the use of the existing classical algorithms, it causes an additional complexity due to the following two causes:

1) Additional processing is needed to convert the fuzzy context to crisp one and the consequent concepts transformation to fuzzy ones.
2) Expanding the context results in increasing the number of objects and hence causes a serious increase in complexity [16].

On the other hand, full-sided based algorithms generate fuzzy concepts with fuzziness involved in both concept extent and intent. Such algorithms are found in [11-14] and rely on the fact that the fuzzy subsets relation is strongly linked to the implication notion. Consequently, it formulates FFCA in terms of fuzzy algebras. Abstractly, this approach defines the complete lattice L as $(L, \vee, \wedge, \otimes, \rightarrow, 0, 1)$ such that $(L, \vee, \wedge, 0, 1)$ represents the complete lattice. Moreover, $(L, \otimes, 1)$ represents the abelian monoid and $\rightarrow$, $\otimes$ represent the adjoint operations $(i.e., a \otimes b \leq c \Leftrightarrow a \leq b \wedge c)$. The set of whole fuzzy sets in the universe of discourse G is denoted by L_G where A: $G \rightarrow L$ is a mapping that gives each element $g \in G$ a truth value $A(g) \in L$. Using such definition, Eqs. (8) and (9) define the derivation operators of extent and intent for context (G, M, I).

$$A\uparrow(m) := \wedge_{g \in G}(A(g) \rightarrow I(g, m)) \quad (8)$$

$$B\downarrow(g) = \wedge_{m \in M}(B(m) \rightarrow I(g, m)) \quad (9)$$

In Eqs. (8) and (9), both intents and extents are fuzzy sets. It is worth mentioning that the count of extracted fuzzy concepts depends entirely on the count of distinct membership values generated by the used implication function (Lukasiewicz/ Gödel implications). Professionally, full-sided based algorithms extract all possible fuzzy concepts. But unfortunately, they usually result in a huge number

of fuzzy concepts and hence consume unordinary larger processing time. Consequently, they are of limited practical use, especially for huge datasets. As a result, some algorithms have been inspired to decrease the count of extracted fuzzy concepts and perform concepts reduction as in [23, 24].

On the other hand, one-sided-based algorithms aim mainly to generate fuzzy-concepts with fuzzy extent / crisp intent or crisp extent / fuzzy intent. Such algorithms usually reduce the number of generated fuzzy concepts with an acceptable loss of information by ignoring fuzziness of one concept side. In consequence, they are suitable for large datasets. Furthermore, they gained wide use in practical applications such as association rule mining and fuzzy Ontology construction. Such algorithms can be found in [5, 10, 15-18].

However, almost recent algorithms suffer from the use of linguistic values with a fixed and different definition for each specified universe of discourse.

5. The proposed approach

The proposed approach incorporates two main phases. The first one is concerned with fuzzifying the given dataset into data-sensitive fuzzy one in a reliable manner. The second phase represents an enhanced algorithm for fuzzy concepts extraction with an impacting reduction in complexity and improvement in the output quality.

5.1 Data-sensitive fuzzification phase

This phase is concerned with the preparation of the given dataset into a fuzzy one. It is based mainly on the idea of data-sensitive fuzzy sets in which a linguistic label or linguistic value has dynamic definition respecting the minimum and maximum values in its universe of discourse [25]. Fig. 1 represents an example of three linguistic-values namely 'low', 'moderate' and 'high' defined over a linguistic variable respecting its minimum and maximum values. Accordingly, the use of the same linguistic-label with different attributes in the dataset becomes valid and reliable since it has just one unified definition.
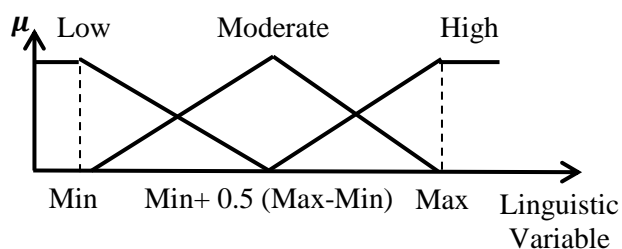

Figure. 1 Data sensitive linguistic labels

5.2 Fuzzy concept extraction phase

In this phase, the proposed enhanced algorithm is used to extract the set of one-sided fuzzy concepts satisfying a predefined threshold. This algorithm extracts fuzzy concepts by applying a set of modifications to the Yang et al. algorithm [21]. Such enhancement reduces the time consumed to obtain the same result by pruning unnecessary processing. Furthermore, the proposed algorithm generates one-sided fuzzy concepts using Eqs. (6) and (7). Consequently, the set operations used for evaluating fuzzy concepts' extents are Zadeh's fuzzy set operations [7] whereas the set operations used for intents are crisp set operations.

The proposed algorithm takes a raw input fuzzy context and a threshold interval as inputs. Afterward, it filters the input fuzzy context respecting the predefined threshold interval on the membership values while extracting the fuzzy concept extent. Consequently, no extra time wasted for converting the entire context to the filtered one.

The processing scenario of the proposed algorithm can be described as follows: Initially, in line 1, the fuzzy concept list C is initialized by the fuzzy concept *(M′, M)* where M denotes the whole attributes set and, *M′* is the result of applying derivation operator over M. In consequence, *M′* refers to objects that share all attributes in M. Here *M′* is defined using Eq. (7).

Proposed algorithm for fuzzy concepts extraction
Inputs: 1) A fuzzy formal context K=(G, M, I=φ (G×M))
2) A confidence threshold $T = [t_1, t_2]$
Output: The set *C* of all fuzzy concepts of *K*
Begin
1   Initialize $C \leftarrow (M′, M)$
2   *for each* object $g \in G$
3   Set NewPerRow $\leftarrow 0$;
4   *if* $g′ \notin C.\text{intents}$ *then*
5           $C \leftarrow C \cup ((g″, \mu_I), g′)$;
6   NewPerRow $++$;
7   *for* $j = 0$ to $(C.\text{size} - \text{NewPerRow})$ *do*
8   *if* $|c_j.\text{Intent}| > 1$ and $|c_j.\text{Intent}| < |M|$
9                   $\text{Inters} := c_j.\text{Intent} \cap g′$;
10   *if* $\text{Inters} \notin C.\text{intents}$ then
11                       $C = C \cup ((\text{Inters }″, \mu_I), \text{Inters }′)$;
12   NewPerRow $++$;
13   *end if*
14   *end if*
15   *end for*
16   *end if*
17   *end for*
18   *if* $G \notin C.\text{extents}$ *then*
19       $C \leftarrow C \cup (G, G′)$
20   *end if*

In line 2, a loop is used to iterate over each object g in the object set G. In line 3 a variable *NewPerRow* is initialized to 0 for each object. Such variable is used mainly to eliminate redundant iterations to extract fuzzy concepts. Consequently, line 4 checks if the current object intent (g') already exists in the fuzzy concept list intents *(C.intents)*. If so, then a new concept *((g'', μ_I), g')* is added to the concept list *C* and the *NewPerRow* variable is incremented by one (line 6). Such new concept *((g'', μ_I), g')* is calculated such that:
- g'' is the closure of the object g that is resulted by applying Eq. (6) then applying Eq. (**7**).
- μ_I represents the membership value of each object in the set g'' to the intent set g'. Such membership value μ_I is calculated by Zadeh's T-norm operator (Minimum).

Subsequently, the loop in line 7 iterates over all fuzzy concepts in the list C and excludes the new added fuzzy concepts by the object g. Therefore, this loop starts from the beginning index 0 to the size of the concept list minus *NewPerRow* that represents the number of fuzzy concepts generated by the object g. Such treatment removes the unjustified redundant iterations. Moreover, this loop aims mainly to extract the new fuzzy concepts resulted by intersecting the newly discovered concept (line 5) and every old concept c_j in the fuzzy concept list *C*.

In line 8, another test checks if the concept intent length $|cj.Intent|$ is greater than 1 and less than the total count of attributes $|M|$. If so, the intersection between the concept intent (*cj.Intent*) and the currently-discovered concept intent g' is calculated. Moreover, if the intersection results in a new concept intent (line 10), then a new concept is discovered with the intersection result as intent (line 11) and the *NewPerRow* variable is incremented.

Finally, after completing all loops, an additional test is carried out to check if the concept list *C* doesn't contain the extent *G*. If yes, then the new concept (G, G') is added to the list C such that:
- *G* is the whole objects set all with 1 membership value.
- *G'* is the derivation of *G* that is evaluated using Eq. (6).

Generally, the proposed algorithm relies mainly on the number of distinct intents per objects. In consequence, it performs more efficiently in case of the counts of distinct intents per objects are relatively small.

Unlike Yang et al. algorithm, the proposed algorithm checks if the object intent already exists in the concept list before any computation (line 4). If it exists, then no extra-iterations are performed. And

hence processing time is shortened. Moreover, the use of *NewPerRow* variable prevents rechecking intents previously generated by the same object. Consequently, the proposed algorithm outperforms the corresponding Yang et al. algorithm by removal of unnecessary iterations. Such observation is shown in details in the experiments section.

6. Experiments and evaluation

First of all, an example is introduced to illustrate the entire process of the proposed approach for fuzzy concepts extraction.

Example 1: Consider a multi-valued context given in Table 1 where there exist two attributes: salary and age. Such multi-valued context is fuzzified such that each linguistic variable has three linguistic values. For instance, salary linguistic variable has three linguistic values: low, moderate and high. On the other hand, the age linguistic variable has three linguistic values: young, youth and old. As illustrated in Fig. 2, all linguistic values are defined as non-stationary fuzzy sets to get a unified definition for all linguistic variables. In consequence, Fig. 2 introduces an example of a unified definition that is set by the designer for all linguistic variables. Afterward, such data-sensitive definition of linguistic values is applied with all used linguistic variables.

Table 2 illustrates the range intervals of both salary and age attributes. Moreover, Table 3 shows the fuzzy context produced by fuzzifying the multi-valued context in Table 1 through applying linguistic values definitions described in Fig. 2.

Table 1. Multi-valued context example

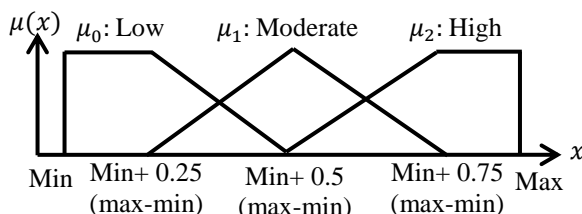| Obj.	Salary	Age
O0	1000	20
O1	5000	27
O2	7000	30
O3	10000	45
O4	8000	50

Figure. 2 Data sensitive linguistic labels definition designed for the multi-valued context shown in Table 1.

Table 2. Ranges of salary and age attributes

Attribute	Min	Max
Salary	1000	10,000
Age	16	60

In consequence, the resulted fuzzy context becomes ready for applying the proposed fuzzy concept generation algorithm that uses Eqs. (6) and (7) like in [21]. Accordingly, the whole set of generated fuzzy concepts with a threshold interval of [0, 1] is illustrated in Table 4.

Alternatively, Table 5 shows the result of adopting equations Eqs. (4) and (5) in algorithms like in attribute-based algorithm [26] and Fuzzy CbO algorithm [27].

It is noticeable that the count of the fuzzy concepts generated using Eqs. (6) and (7) is mostly less than the corresponding ones using Eqs. (4) and (5). Consequently, recent FFCA-related works, like [28], that need a great reduction in the fuzzy concepts count use Eqs. (6) and (7) for generating fuzzy concepts.

Table 3. The fuzzy context generated by applying the data-sensitive linguistic values defined in Fig. 2

Obj	Sal_{Low}	Sal_{mod}	Sal_{high}	Age_{young}	Age_{youth}	Age_{Old}
O0	1	0	0	1	0	0
O1	0.22	0.78	0	1	0	0
O2	0	0.33	0.67	0.73	0.27	0
O3	0	0	1	0	0.36	0.64
O4	0	0	1	0	0	1

Table 4. Fuzzy concepts extracted using Eqs. (6) and (7) from fuzzy context in Table 3 with threshold interval [0, 1]

#	Fuzzy formal concepts
1	$\left\{\left(\frac{O0}{1.0}\right)\left(\frac{O1}{1.0}\right)\left(\frac{O2}{1.0}\right)\left(\frac{O3}{1.0}\right)\left(\frac{O4}{1.0}\right)\right\},\{\ \}$
2	$\left\{\left(\frac{O0}{1.0}\right)\left(\frac{O1}{1.0}\right)\left(\frac{O2}{0.73}\right)\right\},\{Age_{young}\}$
3	$\left\{\left(\frac{O2}{0.67}\right)\left(\frac{O3}{1.0}\right)\left(\frac{O4}{1.0}\right)\right\},\{Sal_{high}\}$
4	$\left\{\left(\frac{O0}{1.0}\right)\left(\frac{O1}{0.22}\right)\right\},\{Sal_{Low},Age_{young}\}$
5	$\left\{\left(\frac{O1}{0.78}\right)\left(\frac{O2}{0.33}\right)\right\},\{Sal_{mod},Age_{young}\}$
6	$\left\{\left(\frac{O2}{0.27}\right)\left(\frac{O3}{0.36}\right)\right\},\{Sal_{high},Age_{youth}\}$
7	$\left\{\left(\frac{O3}{0.64}\right)\left(\frac{O4}{1.0}\right)\right\},\{Sal_{high},Age_{Old}\}$
10	$\left\{\left(\frac{O3}{0.36}\right)\right\},\{Sal_{high},Age_{youth},Age_{Old}\}$
8	$\left\{\left(\frac{O1}{0.22}\right)\right\},\{Sal_{Low},Sal_{mod},Age_{young}\}$
9	$\left\{\left(\frac{O2}{0.27}\right)\right\},\{Sal_{mod},Sal_{high},Age_{young},Age_{youth}\}$
11	$\{\ \},\{Salary_{Low},Salary_{Mod},Salary_{High},Age_{young},Age_{Youth},Age_{Old}\}$

Table 5. Fuzzy concepts extracted using Eqs. (4) and (5) from fuzzy context in Table 3

#	Fuzzy Formal Concepts
1	$\left\{\left(\frac{O0}{1.0}\right)\left(\frac{O1}{1.0}\right)\left(\frac{O2}{1.0}\right)\left(\frac{O3}{1.0}\right)\left(\frac{O4}{1.0}\right)\right\},\{\ \}$
2	$\left\{\left(\frac{O0}{1.0}\right)\left(\frac{O1}{1.0}\right)\left(\frac{O2}{0.73}\right)\right\},\{Age_{Young}\}$
3	$\left\{\left(\frac{O2}{0.67}\right)\left(\frac{O3}{1.0}\right)\left(\frac{O4}{1.0}\right)\right\},\{Salary_{High}\}$
4	$\left\{\left(\frac{O0}{1.0}\right)\left(\frac{O1}{0.22}\right)\right\},\{Salary_{Low},Age_{young}\}$
5	$\left\{\left(\frac{O1}{0.78}\right)\left(\frac{O2}{0.33}\right)\right\},\{Salary_{Mod},Age_{young}\}$
6	$\left\{\left(\frac{O2}{0.67}\right)\right\},\{Salary_{High},Age_{young}\}$
7	$\left\{\left(\frac{O2}{0.27}\right)\left(\frac{O3}{0.36}\right)\right\},\{Salary_{High},Age_{Youth}\}$
8	$\left\{\left(\frac{O3}{0.64}\right)\left(\frac{O4}{1.0}\right)\right\},\{Salary_{High},Age_{old}\}$
9	$\left\{\left(\frac{O3}{0.36}\right)\right\},\{Salary_{High},Age_{Youth},Age_{old}\}$
10	$\left\{\left(\frac{O1}{0.22}\right)\right\},\{Salary_{Low},Salary_{mod},Age_{Young}\}$
11	$\left\{\left(\frac{O2}{0.33}\right)\right\},\{Salary_{Mod},Salary_{High},Age_{young}\}$
12	$\left\{\left(\frac{O2}{0.27}\right)\right\},\{Salary_{Mod},Salary_{High},Age_{young},Age_{Youth}\}$
13	$\{\ \},\{Salary_{Low},Salary_{Mod},Salary_{High},Age_{young},Age_{youth},Age_{old}\}$

For instance, in Table 5, fuzzy concepts no. 11 and no. 12 are two different concepts over the same object {O2} but with different membership degrees. Moreover, the intent of concept no.11 is a subset of the intent of concept no. 12. However, Table 4 preserves only the concept no. 12 (the concept no. 9 in Table 4).

Consequently, Eqs. (6) and (7) generate only fuzzy concepts with the largest intent with respect to the set inclusion. Therefore, the proposed algorithm always generates such type of fuzzy concepts and hence, generates less count of fuzzy concepts. In the worst case, it generates the same number of fuzzy concepts. Consequently, Eqs. (6) and (7) are more suitable for hardly processed datasets.

It is worth mentioning that all recent fuzzy concepts generation algorithms like [21, 26, 27] and the proposed algorithm generate the same count of concepts in case of crisp datasets. This backs to the fact that the fuzzy formal concept analysis is a generalized form of the traditional crisp one.

Besides, this section presents some experiments conducted over two benchmark datasets. The first dataset "Iris"[1] is available in the machine learning repository (UCI)[2]. Additionally, the second dataset "Countries investment dataset" is available in [29].

The Iris dataset comprises 150 objects, representing a set of iris plant instances, and five attributes. Four of the five attributes are quantitative attributes that represent the iris plant characteristics. The remaining attribute is a categorical one that represents the iris class (three classes). Ordinarily, each iris instance belongs to a specific iris class. Illustratively, Table 6 presents a snapshot of the iris dataset that only includes ten objects and five attributes. These five attributes include sepal length (SL), sepal width (SW), petal length (PL), petal width (PW) and the iris class.

Table 7 reveals that each attribute (linguistic variable) has its own range of minimum and maximum values such that no value can lie outside these ranges. Consequently, the design of the data-sensitive linguistic values copes effectively with the differences between attributes min and max values.

As a preparation phase, the Iris dataset is fuzzified using a set of predefined data-sensitive linguistic values namely "low", "middle" and "high" shown in Fig. 3. Such definition is just produced by setting the min and max values in the design depicted in Fig. 1. Likewise, all other attributes are fuzzified. Therefore, the total count of all attributes after fuzzification using three linguistic labels becomes 15 attributes. In consequence, each attribute value feds into each of its linguistic value membership functions to obtain how much each object matches each linguistic value. Moreover, the categorical attribute is discretized to be in a binary representation. As a sample, Table 8 shows the result of the fuzzification process of only two attributes: sepal length and sepal width. Moreover, it shows the discretization of the class attribute.

Table 6. A snapshot of the iris dataset

ID	SL	SW	PL	PW	Class
O1	5.1	3.8	1.9	0.4	Iris-setosa
O2	4.8	3	1.4	0.3	Iris-setosa
O3	5.1	3.8	1.6	0.2	Iris-setosa
O4	4.6	3.2	1.4	0.2	Iris-setosa
O5	5.3	3.7	1.5	0.2	Iris-setosa
O6	5	3.3	1.4	0.2	Iris-setosa
O7	7	3.2	4.7	1.4	Iris-versicolor
O8	6.4	3.2	4.5	1.5	Iris-versicolor
O9	6.9	3.1	4.9	1.5	Iris-versicolor
O10	5.5	2.3	4	1.3	Iris-versicolor

Table 7. Ranges of iris dataset linguistic variables

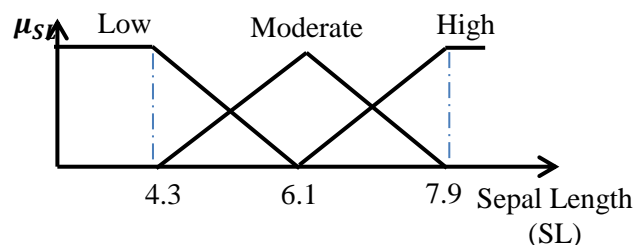| Linguistic variable	Min	Max
Sepal length	4.3	7.9
Sepal width	2.0	4.4
Petal length	1.0	6.9
Petal width:	0.1	2.5

Figure. 3 Linguistic values design for the sepal length linguistic variable in the iris dataset

Table 8. Fuzzy context of the iris dataset snapshot shown in Table 6 for only SL, SW and the class attributes

ID	SL_{low}	SL_{mid}	SL_{high}	SW_{low}	SW_{mid}	SW_{high}	setosa	versi color
O1	0.56	0.44	0	0	0.5	0.5	1	0
O2	0.72	0.28	0	0.17	0.83	0	1	0
O3	0.56	0.44	0	0	0.5	0.5	1	0
O4	0.83	0.17	0	0	1	0	1	0
O5	0.44	0.56	0	0	0.58	0.42	1	0
O6	0.61	0.39	0	0	0.92	0.08	1	0
O7	0	0.5	0.5	0	1	0	0	1
O8	0	0.83	0.17	0	1	0	0	1
O9	0	0.56	0.44	0.08	0.92	0	0	1
O10	0.33	0.67	0	0.75	0.25	0	0	1

The proposed algorithm is evaluated versus the Yang et al. algorithm presented in [21], the attribute-based algorithm presented in [26], and the fuzzy CbO algorithm presented in [27] over the complete iris dataset. In consequence, Table 9 shows a comparison in terms of time in milliseconds for extracting the whole set of fuzzy concepts over the iris dataset with different threshold intervals. On the other hand, Table 10 shows a comparison with respect to the number of extracted fuzzy concepts. As expected, the proposed algorithm generates the same count of fuzzy concepts as the Yang et al. algorithm with a great reduction in the processing time and enhanced semantics due to involving data-sensitive fuzzy terms.

It is worth mentioning that, both proposed algorithm and Yang et al. algorithm use the same equations namely Eqs. (6) and (7) to extract fuzzy concepts.

Table 9. The proposed algorithm versus Yang et al. algorithm, attribute-based algorithm and fuzzy CbO algorithm in terms of time in milliseconds over the iris dataset.

Threshold Interval	(1) proposed algorithm	(2) Yang's algorithm	(3) Attribute-based algorithm	(4) Fuzzy CbO algorithm
[0.0, 1.0]	6.644	58.902	20.86	60.006
[0.2, 1.0]	5.436	43.627	11.432	18.055
[0.4, 1.0]	4.604	17.934	7.911	11.952
[0.6, 1.0]	2.938	8.195	4.689	8.77
[0.8, 1.0]	2.708	9.798	2.813	9.376

Table 10. The proposed algorithm versus Yang et al., attribute-based, and fuzzy CbO algorithms in terms of no. of extracted concepts over the iris dataset.

Threshold Interval	proposed algorithm	Yang's algorithm	Attribute-based algorithm	Fuzzy CbO algorithm
[0.0, 1.0]	293	293	702	702
[0.2, 1.0]	263	263	476	476
[0.4, 1.0]	139	139	227	227
[0.6, 1.0]	78	78	120	120
[0.8, 1.0]	62	62	71	71

On the other hand, both the attribute-based algorithm and fuzzy CbO algorithm generate more fuzzy concepts as a result of using Eqs. (4) and (5).

All the experimented algorithms have been programmed using java programming language and all experiments are conducted using the same machine with the following capabilities: core i5 3.2 GHz processor and 4 GB RAM memory. Moreover, the installed operating system was 64-bit windows 8.1.

Both of Figs. 4 and 5 show the corresponding graphs that summarize the experimental results. As shown in Table 9, the proposed algorithm achieves a great reduction in the execution time compared with Yang et al. algorithm, the attribute-based algorithm and, CbO algorithm.

On the other hand, Table 10 shows that the proposed algorithm generates the same concepts as Yang et al. algorithm without any loss. In contrary, it generates less number of fuzzy concepts when compared with the attribute-based algorithm and fuzzy Cbo algorithm as a result of using different equations set. The great reduction in the number of generated fuzzy concepts makes the proposed algorithm more suitable for complex datasets where there exists a huge number of fuzzy concepts.
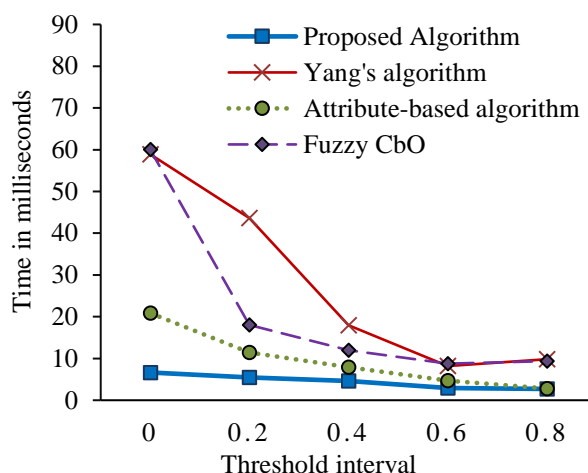

Figure. 4 The proposed algorithm versus Yang et al. [21], attribute-based [26], and fuzzy CbO [27] algorithms in terms of time in milliseconds over the iris dataset with different threshold intervals
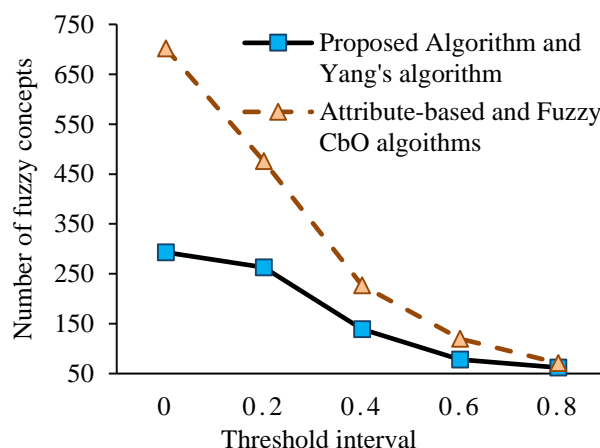

Figure. 5 Proposed algorithm and Yang et al. algorithm versus attribute based algorithm and Fuzzy CbO algorithm [21, 26] in terms of number of fuzzy concepts over iris dataset with different threshold intervals

Table 11. A snapshot of the countries investment dataset

Country	Political stability (A)	Attitude toward investors (B)
Canada	3.7	3.7
USA	3.7	3.7
Mexico	2.9	2.9
Argentina	1.1	1.1
Brazil	2.8	2.8
Chile	0.3	0.3
Colombia	2	2
Peru	1.6	1.6
Venezuela	2.5	2.5
Australia	3.4	3.4

Table 12. Fuzzy context of the countries investment dataset snapshot shown in Table 11

Country	A$_{low}$	A$_{mod}$	A$_{high}$	B$_{low}$	B$_{mod}$	B$_{high}$
Canada	0	0	1	0	0.3	0.7
USA	0	0	1	0	0	1
Mexico	0	0.1	0.9	0	0.8	0.2
Argentina	0.9	0.1	0	0.3	0.7	0
Brazil	0	0.2	0.8	0	0.2	0.8
Chile	1	0	0	1	0	0
Colombia	0	1	0	0	1	0
Peru	0.4	0.6	0	0.8	0.2	0
Venezuela	0	0.5	0.5	0	0.9	0.1
Australia	0	0	1	0	0.3	0.7

Table 13. The proposed algorithm versus Yang et al., attribute-based and fuzzy CbO algorithms in terms of time in milliseconds over countries investment dataset.

Threshold Interval	(1) proposed algorithm	(2)Yang et al. algorithm	(3) Attribute-based algorithm	(4) Fuzzy CbO
[0.4, 1]	530.1	959.5	7330.0	7850.7
[0.5, 1]	150.3	405.7	2197.5	2633.9
[0.6, 1]	50.7	140.8	469.7	487.3
[0.7, 1]	45.9	56.8	76.5	165.9
[0.8, 1]	15.7	33.2	64.5	89.4

The second dataset "Countries investment dataset" has 43 objects, representing a set of countries, and 15 attributes representing investment confidence criteria. All attributes have the same range of [0, 4] such that 0 refers to max risk to invest and 4 refers to min risk to invest. For more clarification, Table 11 shows a snapshot of the countries investment dataset. It shows only 10 countries and 2 attributes namely political-stability and attitude-toward-investors. In short, these attributes are referred to as A and B respectively.

At the first stage of the proposed approach, the data-sensitive linguistic values depicted in Fig. 2 are applied to fuzzify the countries investment dataset. Subsequently, Table 12 shows the fuzzy context resulted from fuzzifying the snapshot in Table 11 with min and max values 0 and 4 respectively.

Intuitively, the total count of attributes in the entire countries investment dataset after fuzzification becomes 45 attributes.

At the second stage of the proposed approach, the resulted fuzzy context of the first stage represents the input of the applied fuzzy concept generation algorithm. Consequently, all of the proposed, attribute-based, Yang et al., and fuzzy CbO algorithms are applied to the fuzzy context. Finally, Table 13 and Fig. 6 show a comparison between these algorithms in terms of execution time in milliseconds for different threshold intervals.

Noticeably, the proposed algorithm outperforms the corresponding algorithms as it only considers distinct intents per objects and only extracts the maximum length fuzzy concept intents for the similar concept extents.

Moreover, Table 14 and Fig. 7 illustrate another comparison with respect to the number of generated fuzzy concepts.
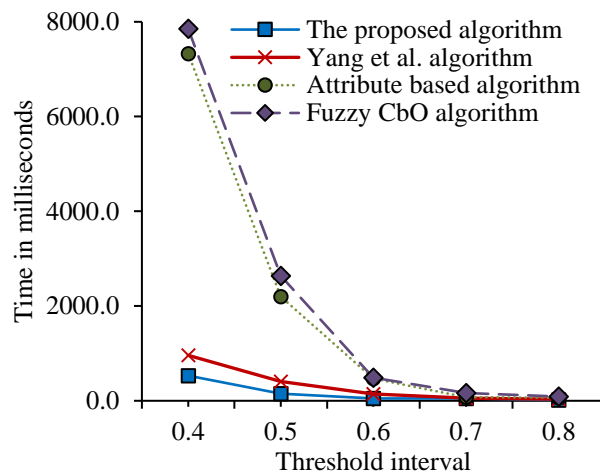

Figure. 6 The proposed algorithm versus Yang et al., attribute-based, and fuzzy CbO algorithms [21, 26, 27] in terms of time in milliseconds over the countries investment dataset with different threshold intervals

Table 14. The proposed algorithm versus Yang et al., attribute-based, and fuzzy CbO algorithms in terms no. of extracted concepts over the countries investment dataset.

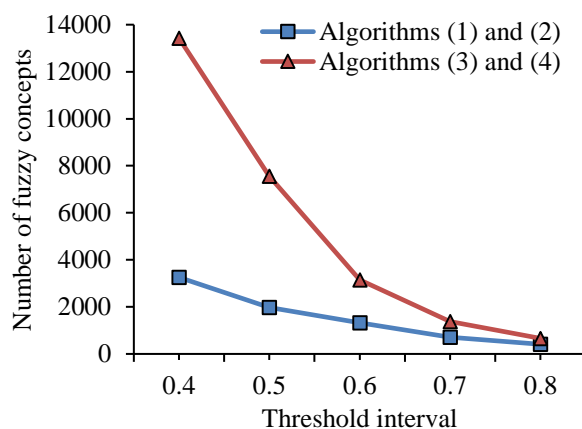| Threshold Interval	Algorithms (1) and (2)	Algorithms (3) and (4)
[0.4, 1]	3256	13427
[0.5, 1]	1979	7552
[0.6, 1]	1323	3141
[0.7, 1]	717	1384
[0.8, 1]	411	661

Figure. 7 Proposed algorithm and Yang et al.'s algorithm versus attribute-based and Fuzzy CbO algorithms [21, 26] in terms of number of fuzzy concepts over the countries investment dataset with different threshold intervals

7. Conclusion

In general, FFCA is considered one of the most essential and challenging fields that are exploited by several applications including association rule mining and ontology construction. Numerous researchers have tried to improve and accelerate the entire FFCA process. However, there still a need for more improved approaches.

This paper attempts to enhance the process of fuzzy formal concept extraction in both of the semantic of generated fuzzy concepts and the processing time it consumes. The semantics of the generated fuzzy concepts are enhanced through the use of data-sensitive fuzzy terms that preserve flexible scalable definitions when applied to different universes of discourses. On the other hand, an improved algorithm has been presented to reduce the execution time consumed in the fuzzy concept generation process.

Compared with Yang et al. algorithm, attribute-based algorithm, and fuzzy CbO algorithm, the proposed enhanced algorithm amazingly reduces fuzzy concepts extraction time by considering only the distinct intent set. In consequence, it skips the identical object intents at earlier step without any additional processing. So, all identical records are processed only once.

On the other hand, the proposed enhanced algorithm generates less fuzzy concepts than the attribute-based and fuzzy CbO algorithms. This backs to the fact that the proposed algorithm generates only fuzzy concepts with largest intents according to the set inclusion. Consequently, It works extremely faster so it can be used for large datasets where extracting fuzzy concepts is an intractable task.

As a future work, we aim to find an enhanced updatable algorithm for incrementally update the already produced fuzzy concepts.

References

[1] G. Bernhard and W. Rudolf, *Formal Concept Analysis: Mathematical Foundations*, Springer, Berlin, Heidelberg, 1999.

[2] R. Wille, "Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts", In: *Ordered Sets, NATO Advanced Study Institutes Series,* Springer, Dordrecht, pp.445-470, 1982.

[3] S. Zheng, Y. Zhou, and T. Martin, "A New Method for Fuzzy Formal Concept Analysis", In: *Proc. of the 2009 IEEE/WIC/ACM International Joint Conf. on Web Intelligence and Intelligent Agent Technology*, pp.405-408, 2009.

[4] R. Wille, "Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies", *Formal Concept Analysis,* Vol.3626, pp.1-33, 2005.

[5] C. A. Kumar, S. C. Mouliswaran, P. Amriteya, and S. Arun, "Fuzzy Formal Concept Analysis Approach for Information Retrieval", In: *Proc. of the Fifth International Conf. on Fuzzy and Neuro Computing*, pp.255-271, 2015.

[6] B. Ganter, G. Stumme, and R. Wille, *Formal Concept Analysis: Foundations and Applications,* Vol.3626, Lecture Notes in Artificial Intelligence LNAI, Springer-Verlag Berlin Heidelberg, 2005.

[7] L. A. Zadeh, "Fuzzy Sets*", Information and Control,* Vol.8, No.3, pp.338-353, 1965.

[8] S. O. Kuznetsov and S. A. Obiedkov, "Comparing Performance of Algorithms for Generating Concept Lattices", *Journal of Experimental & Theoretical Artificial Intelligence,* Vol.14, No.2-3, pp.189-216, 2002.

[9] P. Krajca, J. Outrata, and V. Vychodil, "Parallel Algorithm for Computing Fixpoints of Galois Connections", *Annals of Mathematics and Artificial Intelligence,* Vol.59, No.2, pp.257-272, 2010.

[10] R. Belohlavek, "Algorithms for Fuzzy Concept Lattices", In: *Proc. the Fourth International Conf. on Recent Advances in Soft Computing*, pp.200-205, 2002.

[11] R. Belohlavek, B. De Baets, J. Outrata, and V. Vychodil, "Computing the Lattice of All Fixpoints of a Fuzzy Closure Operator", *IEEE Transactions on Fuzzy Systems,* Vol.18, pp.546-557, 2010.

[12] R. Belohlavek, B. De Baets, J. Outrata, and V. Vychodil, "Lindig's Algorithm for Concept Lattices over Graded Attributes", In: *Proc. of the International Conf. on Modeling Decisions for Artificial Intelligence*, pp.156-167, 2007.

[13] R. Belohlavek, "What Is a Fuzzy Concept Lattice? II", In: *Rough Sets, Fuzzy Sets, Data Mining and Granular Computing, Lecture Notes in Computer Science,* Vol.6743. Springer, Berlin, Heidelberg, pp.19-26, 2011.

[14] R. Belohlavek and V. Vychodil, "Formal Concept Analysis and Linguistic Hedges", *International Journal of General Systems,* Vol.41, No.5, pp.503-532, 2012.

[15] T. Martin, Y. Shen, and A. Majidian, "Soft Concept Hierarchies to Summarise Data Streams and Highlight Anomalous Changes", In: *Proc. of the International Conf. on*

Information Processing and Management of Uncertainty in Knowledge-Based Systems, pp.44-54, 2010.

[16] A. Majidian, T. Martin, and M. E. Cintra, "Fuzzy Formal Concept Analysis and Algorithm", In: *Proc. of the 11th UK Workshop on Computational Intelligence*, pp.61-67, 2011.

[17] T. Martin and A. Majidian, "Beyond the known unknowns-finding fuzzy concepts for creative knowledge discovery", In: *Proc. of World Conference on Soft Computing*, 2011.

[18] T. Martin, Z. Siyao, and A. Majidian, "Fuzzy Taxonomies for Creative Knowledge Discovery", In: *Proc. of the Fifth International Conf. on Uncertainty Reasoning for the Semantic Web*, Vol.527, pp.81-84, 2009.

[19] T. T. Quan, S. C. Hui, and T. H. Cao, "A Fuzzy FCA-based Approach to Conceptual Clustering for Automatic Generation of Concept Hierarchy on Uncertainty Data", In: *Proc. of CLA International Workshop on Concept Lattices and their Applications,* pp.1-12, 2004.

[20] V. Cross and M. Kandasamy, "Creating Fuzzy Concepts: The One-Sided Threshold, Fuzzy Closure and Factor Analysis Methods", In: *Proc. of the International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*, pp.127-134, 2011.

[21] K. M. Yang, E. H. Kim, S. H. Hwang, and S. H. Choi, "Fuzzy Concept Mining Based on Formal Concept Analysis", *International Journal of Computers,* Vol.2, No.3 pp.279-290, 2008.

[22] R. Bělohlávek, V. Sklenář, and J. Zacpal, "Crisply Generated Fuzzy Concepts", In: *Lecture Notes in Computer Science, Third International Conference on Formal Concept Analysis*, Vol.3403, pp.269-284, 2005.

[23] P. K. Singh and C. A. Kumar, "A Method for Reduction of Fuzzy Relation in Fuzzy Formal Context", *Mathematical Modelling and Scientific Computation*, Springer, Berlin, Heidelberg, Vol.238, pp.343-350, 2012.

[24] P. K. Singh, A. K. Cherukuri, and J. Li, "Concepts Reduction in Formal Concept Analysis with Fuzzy Setting Using Shannon Entropy", *International Journal of Machine Learning and Cybernetics,* Vol.8, No.1 pp.179-189, 2017.

[25] A. M. Gadallah and H. A. Hefny, "Fuzzy Cognitive Map with Dynamic Fuzzification and Causality Behaviors", In: *Proc. of the 7th International Conference on Informatics and Systems*, 2010.

[26] E. E. Shemis and A. M. Gadallah, "Enhanced Algorithms for Fuzzy Formal Concepts Analysis", In: *Proc. of International Conf. on Advanced Intelligent Systems and Informatics*, Vol.533 pp.781-792, 2016.

[27] T. Martin and A. Majidian, "Finding Fuzzy Concepts for Creative Knowledge Discovery", *International Journal of Intelligent Systems*, Vol. 28, No.1, pp.93-114, 2013.

[28] K. Ravi, V. Ravi, and S. R. K. Prasad, "Fuzzy Formal Concept Analysis based Opinion Mining for CRM in Financial Services", *Applied Soft Computing*, Vol. 60, pp.786-807, November 2017.

[29] M. Jambu, Exploratory and Multivariate *Data Analysis*, Academic Press Inc., Orlando, 1991.