IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 1, 2018, accepted April 5, 2018, date of publication April 9, 2018, date of current version May 9, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2824833

A Formal Approach of Construction Fuzzy XML Data Model Based on OWL 2 Ontologies

WEIJUN LI [1], LI YAN [2], FU ZHANG [1], AND XU CHEN [3]

[1]School of Computer Science and Engineering, Northeastern University, Shenyang 110819, China
[2]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China
[3]Internet Information Technology Center, North Minzu University, Yinchuan 750021, China

Corresponding author: Li Yan (yanli@nuaa.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61370075 and Grant 61672139.

ABSTRACT With the rapid development of the Web, a large number of electronic resources have been generated. Currently, XML has been an important tool for data representation and exchange over the Web. The incompleteness of information in the real-world is inherent. To deal with imprecise and uncertain data, fuzzy XML and fuzzy ontology modeling recently receive more attention. In order to represent the fuzzy information, we concentrate on fuzzy information modeling in a fuzzy XML model and fuzzy OWL 2 ontology in this paper. Furthermore, we propose an approach of transforming fuzzy OWL 2 ontologies (including structures and instances) into fuzzy XML models. Then we prove that the sementics of this transformation approach are preserved and propose a transforming example to explain the transforming process. This paper provides a new approach for the fuzzy XML modeling and fuzzy XML mapping based on the fuzzy OWL 2 ontologies.

INDEX TERMS Fuzzy XML model, fuzzy DTD, fuzzy OWL 2 ontology, constructing, transforming, mapping.

I. INTRODUCTION

With the rapid development and the comprehensive utilization of the Internet, XML (eXtensible Markup Language) has been an important tool for data representation and exchange over the Web mainly because it is a self-descriptive format that supports a flexible representation of data, and it is an open and free pattern [24]. The reasoning and decision system based on XML has been widely used in artificial intelligence and knowledge engineering. In order to share, exchange, reuse, and integrate information between different systems and users, it is necessary to transform the XML model to other data models [3]. The mapping from data models into XML can benefit database interoperability over the Web. Various data models, including relational databases [10], nested relational databases [11], object-oriented databases [17], [18], [24], object-relational databases [7], EER models [30] and UML models [6], [15], have been mapped to XML document.

In the real-world information is inherently imprecise and uncertain since it values is subjective. To represent and handle imperfect information with XML, Abiteboul *et al.* [1] provided a system using XML and DTD processing incomplete information. They utilize probability theory to deal with ambiguous data in XML has received widespread attention, such as [19], [25], and [27]. Gaurav and Alhajj [8] proposed an approach to incorporate fuzzy and inaccurate data into an XML document. This approach utilizes the possibility theory and the similarity relationship to present fuzzy data and maps the fuzzy data from the fuzzy relational database to a fuzzy XML document with the corresponding XML schema. Oliboni and Pozzani [20] proposed a definition of general XML Schema for representing fuzzy information. Ma and Yan [15] represented a fuzzy XML data model based on possibility distribution theory, and proposed a conceptual structure and database storage methods of this model. Then they presented two mappings from fuzzy UML model to the fuzzy XML model and from the fuzzy XML model to the fuzzy relational database, respectively. Yan *et al.* [29] presented a definition of multiple granularity of data fuzziness based on elements and attribute values of the elements in the XML. They developed this fuzzy XML data model for dealing with all fuzziness based on the XML model. A new fuzzy XML model based on XML Schema and algebraic operations in this model was proposed in [13]. Ma and Yan [16] provide an up-to-date overview of fuzzy XML data modeling in fuzzy data management and the main approaches of modeling fuzzy XML data.

In addition, in order to express and reasoning fuzzy knowledge, fuzzy ontology definitions [12], [28] have been proposed by incorporating fuzzy description logic and fuzzy set theory [31], [32]. In the context of the Semantic Web [12], the Web ontology language (OWL) 2 [21]–[23] becomes the latest standard ontology description language recommended by W3C Web Ontology Working Group. Bobillo and Straccia [4] presented a concrere approach to represent fuzzy ontologies based on OWL 2 annotation properties and a prototypical tool to implement. Our work mainly focus on the fuzzy OWL 2, which is an extension of the OWL 2 based on the Zadehąŕs fuzzy set theory [31], [32]. The logical foundation of fuzzy OWL 2 is the fuzzy DL called *f-SROIQ(D)* [5].

To deal with XML with ontologies, some research has been made to map XML into ontologies. This work in [33] and [36] pay attention to represent and reason about fuzzy XML models with fuzzy ontologies. Hacherouf *et al.* [9] summed up a survey on the different approaches of conversion XML documents to OWL ontologies and presented two main processes of ontology enrichment (Abox) and ontology population (Tbox). In addition, Zhang *et al.* [35] proposed a formal definition of fuzzy XML model and gave an approach and automated tool for constructing fuzzy ontologies from fuzzy XML model. Actually, it is needed to reengine ontologies into other data models. Benslimane *et al.* [2], for example, propose a reverse engineering approach of extracting domain ontology schema to construct conceptual data model so that ontologies can be reused at a conceptual level. Similarly, to reuse and exchange ontologies on the Web, it is useful to map ontologies into XML. This just likes the mapping from databases into XML [7], [10], [17], [18], [24]. Unfortunately, few work investigate the mapping of ontologies into XML. It is especially true to map fuzzy ontologies into fuzzy XML.

Based on Zadeh's fuzzy set theory, we extend a fuzzy XML data model to deal with all types of fuzzy. Then we propose a formal approach of transforming fuzzy OWL 2 ontologies (including structure and instance levels) into fuzzy XML models. The correctness of this approach is proved, and a transformation example is provided to illustrate the proposed approach.

The remainder of this paper is organized as follows. The fuzzy XML data models and fuzzy OWL 2 ontologies are introduced in Section II. In Section III, the approaches to transform fuzzy OWL 2 ontologies (including structure and instance levels) into fuzzy XML models are proposed. Section IV concludes the paper.

II. FUZZY XML MODEL AND FUZZY OWL 2 ONTOLOGY

A. THE REPRESENTATION OF FUZZY XML MODEL

To deal with fuzzy information, we extend XML documents based on fuzzy sets and probability distribution theory. We utilize membership degrees to indicate the fuzziness in elements and possibility distribution to indicate the fuzziness in attribute values of elements. In [13], we propose some concepts about fuzzy XML model.

Definition 1: Let V be a finite set of vertices, $E \in V \times V$ be a set of edges and $\ell : E \to \Gamma$ be a mapping from edges to a set Γ of strings called labels. The triple $G = (V, E, \ell)$ is an edge labeled directed graph.

Definition 2: A fuzzy XML tree τ can be a 6-tuple $\tau = (V, \sigma, \lambda, \eta, \rho, \gamma)$ where

- $V = V_1, \ldots, V_n$ is a set of vertices.
- $\sigma \subset \{(V_i, V_j)|V_i, V_j \in V\}$, (V, σ) is a directed tree.
- $\lambda : V \to (L \cup \{NULL\})$, where L a set of strings called labels. For $v \in V$ and $l \in L$, $\lambda(v, l)$ specifies the set of objects that may be children of v with label l.
- $\eta \to T$, where T is a set of fuzzy XML types [20].
- ρ is a possibility function. It defines the possibility of a set of children nodes given belonging to the parent node.
- γ is a mapping relationship. It defines the number of child nodes that pass through a label l as parent node v, where $v \in V$, $l \in L$. $\gamma(v, l) = [min, max]$, where $min \geq 0$, $max \geq min$, γ is used to represent the lower and upper bounds.

Definition 3 (Fuzzy DTDs): A fuzzy DTD D is a pair (**P**, r), where **P** is a set of *element type definitions*, and $r \in E$ is the *root element type*, which uniquely identifies a fuzzy DTD. Each element type definition has the form $E \to (\alpha, A)$, constructed according to the following syntax:

$$\alpha ::= S|empty|(\alpha_1|\alpha_2)|(\alpha_1, \alpha_2)|\alpha^?|\alpha^*|\alpha^+|any$$
$$A ::= empty|(AN; AT; VT)$$

Here:

- $\mathbf{S} = \mathbf{T} \cup \mathbf{E}$; **T** denotes the atomic types of elements and attributes; **E** denotes a set of elements including the basic elements and special elements **Val** and **Dist**; $'empty'$ denotes the empty string; $'|'$ denotes *union*, and $','$ denotes *concatenation*; α can be extended with cardinality operators $'?'$, $'*'$, and $'+'$, where $'?'$ denotes 0 or 1 time, $'*'$ denotes 0 or n times, and $'+'$ denotes 1 or n times; the construct *any* stands for any sequence of element types defined in the fuzzy DTD.
- $AN \in \mathbf{A}$ denotes the attribute names of the element **E**; *AT* denotes the attribute types; and *VT* is the value types of attributes which can be #REQUIRED, #IMPLIED, #FIXED *value*, *value*, and *disjunctive/conjunctive* possibility distribution.

Definition 4 (Fuzzy XML Documents): A fuzzy XML document d over a fuzzy DTD D is a tuple $d = (N, <, \lambda, \eta, \gamma)$, where:

- N: is a set of nodes in a fuzzy XML document tree.
- $<$: denotes the parent-child relationship between nodes, i.e., for two nodes $v_i, v_j \in N$, if $v_i < v_j$, then v_i is the parent node of v_j.
- λ: $N \to \mathbf{E} \cup \mathbf{A}$ is a labeling function for distinguishing elements and attributes.
- η: $N \times N \to \mathbf{dom}$ is a function for mapping attributes to values such that for each pair nodes $v_i, v_j \in N$ with $v_i < v_j$, if $\lambda(v_j) = @a_j \in \mathbf{A}$, then $\eta(v_i, v_j) = d_j \in \mathbf{dom}$. In particular,

if $\lambda(v_j) = e \in N$ is a leaf element node **E** (such as the element *sname* in *Figure 1*), then $g(v_i, v_j) = d_j \in$ **dom**.

• γ is the root node of a fuzzy XML document tree.

A fuzzy XML document is intuitionally deemed a syntax tree, and conforms to a fuzzy DTD that consists of elements and their associated attributes. A fuzzy XML document [15] has several fuzzy constructs for fuzzy data modeling. A possibility attribute "**Poss**" with a value of [0, 1] together a fuzzy constructor called "**Val**" specifies the possibility of a given element in the XML document. Pair **<Val Poss>** and **</Val>**indicates possibility distribution of an element. The fuzzy construct "**Dist**" has multiple elements "**Val**" as children, each of element has an associated possibility. A construct "**Dist**" indicates two types of possibility distribution *disjunctive* and *conjunctive*.

```
<customer>
 <name> Lucky Vitamin</name>
 <Val Poss=0.78>
  <corporate-customer>
   <contactName>Lucy</ contactName >
    < creditRating >
     <Dist type="disjunctive">
      <Val Poss=0.94>
       <creditRating_value >Level II </creditRating_value >
      </Val>
      </Dist >
     </ creditRating >
   </ corporate-customer >
</customer>
```

FIGURE 1. A fragment of the fuzzy XML document.

Figure 1 gives a fragment of an XML document with fuzzy information [35]. In the example, assuming that it is the possibility that "LuckyVitamin" is included in the *customer*. In addition, the *corporate-customer* has fuzzy values in the attributes `age`, which are represented by a *disjunctive* possibility distribution. *Figure 2* gives a tree representation of the fuzzy XML document in *Figure 1*.

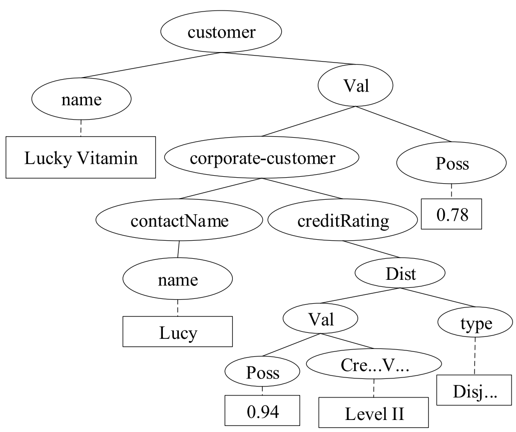


FIGURE 2. The tree representation of Figure.1.

B. FUZZY OWL ONTOLOGY

To define fuzzy OWL 2 ontology, it is necessary to introduce fuzzy OWL language [34], which is based on the Zadeh's fuzzy set theory [31]. The semantics for fuzzy OWL 2 are equivalent to the expressive description logics *f-SHID(D)* and *f-SHONF(D)* [26]. After summarizing the fuzzy OWL in [34] and [35], we present *Table 1* to show the fuzzy OWL 2 abstract syntax, the corresponding description logics syntax, and the semantics.

In *Table 1*, *FC* indicates a fuzzy class; *FCE* indicates a fuzzy class expression; *FDT* indicates a fuzzy datatype; *FDR* indicates a fuzzy data range; *FDP* indicates a fuzzy data property; *FDPE* indicates a fuzzy data property expression; *FOP* indicates a fuzzy ObjectProperty; *FOPE* indicates a fuzzy ObjectProperty expression; α indicates an individual (named or anonymous); *lt* indicates a literal; *FA* indicates a constraining facet; *#S* indicates the cardinality set *S*, and $\bowtie \in \{\geq, >, \leq, <\}$. The semantics of the fuzzy OWL 2 language is proposed in *Definition 5*.

Definition 5 (Semantics of Fuzzy OWL 2 Language): FI is provided by a fuzzy interpretation of the semantics. A datatype map *FD* and a vocabulary *FV* over *FD*, $FI = (\triangle^{FI}, \triangle^{FD}, \bullet^{FC}, \bullet^{FOP}, \bullet^{FDP}, \bullet^{FI}, \bullet^{FDT}, \bullet^{LT}, \bullet^{FA}, NAMED)$ for *FD* and *FV* is a 10-tuple with the following structure [21]:

1) $\triangle^{FI}$ is a nonempty fuzzy set called the *fuzzy object domain*.

2) $\triangle^{FD}$ is a nonempty set disjoint with $\triangle^{FI}$ called the *data domain* such that $(DT)^{FDT} \subseteq \triangle^{FD}$ for each datatype $FDT \in FV^{FDT}$.

3) $\bullet^{FC}$ is the *fuzzy class interpretation function* that assigns to each class $FC \in FV^{FC}$ a subset $(FC)^{FC} \subseteq \triangle^{FI}$ such that $(owl : Thing)^{FC} = \triangle^{FI}$ and $(owl : Nothing)^{FC} = \varnothing$.

4) $\bullet^{FOP}$ is the *fuzzy object property interpretation function* that assigns to each object property $FOP \in FV^{FOP}$ a subset $(FOP)^{FOP} \subseteq \triangle^{FI} \times \triangle^{FI}$ such that $(owl : topObjectProperty)^{FOP} = \triangle^{FI} \times \triangle^{FI}$ and $(owl : bottomObjectProperty)^{FOP} = \varnothing$.

5) $\bullet^{FDP}$ is the *fuzzy data property interpretation function* that assigns to each data property $FDP \in FV^{FDP}$ a subset $(FDP)^{FDP} \subseteq \triangle^{FI} \times \triangle^{FD}$ such that $(owl : topDataProperty)^{FDP} = \triangle^{FI} \times \triangle^{FD}$ and $(owl : bottomDataProperty)^{FDP} = \varnothing$.

6) $\bullet^{FI}$ is the *fuzzy individual interpretation function* that assigns to each individual $\alpha \in FV^{FI}$ an element $(\alpha)^{FI} \in \triangle^{FI}$.

7) $\bullet^{FDT}$ is the *datatype interpretation function* that assigns to each datatype $FDT \in FV^{FDT}$ a subset $(FDT)^{FDT} \in \triangle^{FD}$ such that $\bullet^{FDT}$ is the same as in *FD* for each datatype $FDT \in FV^{FDT}$, and $(rdfs : Literal)^{FDT} = \triangle^{FD}$.

8) $\bullet^{LT}$ is the *literal interpretation function* that is defined as $(lt)^{LT} = (LV, FDT)^{LS}$ for each $lt \in FV^{LT}$, where *LV* is the lexical form of *lt* and *FDT* is the datatype of *lt*.

9) $\bullet^{FA}$ is the *facet interpretation function* that is defined as $(F, lt)^{FA} = (F, (lt)^{LT})^{FS}$ for each $(F, lt) \in FV^{FA}$.

10) *NAMED* is a subset of $\triangle^{FI}$ such that $\alpha^{FI} \in NAMED$ for each named individual $\alpha \in FV^{FI}$.

TABLE 1. Fuzzy OWL abstract syntax, description logic (DL) syntax and interpretation.

Fuzzy OWL abstract syntax	Fuzzy DL syntax	Interpretation
Fuzzy Class description		
$Class(FC)$	FC	$FC^{FI} \subseteq \triangle^{FI}$
$owl:Thing$	$\top$	$(owl:Thing)^{FC} = \triangle^{FI}$
$owl:Nothing$	$\bot$	$(owl:Nothing)^{FC} = \varnothing$
$ObjectIntersectionOf(FCE_1 ... FCE_n)$	$FCE_1 \sqcap ... \sqcap FCE_n$	$(FCE_1)^{FC} \cap ... \cap (FCE_n 0^{FC}$
$ObjectUnionOf(FCE_1 ... FCE_n)$	$FCE_1 \sqcup ... \sqcup FCE_n$	$(FCE_1)^{FC} \cup ... \cup (FCE_n)^{FC}$
$ObjectComplementOf(FCE)$	$\rightharpoondown FCE$	$\triangle^{FI} \setminus (FCE)^{FC}$
$ObjectOneOf(a_1 ... a_n)$	$\{a_1\} \sqcup ... \sqcup \{a_n\}$	$\{(a_1)^{FI}, ..., (a_n)^{FI}\}$
$ObjectSomeValuesFrom(FOPE\ FCE)$	$\exists FOPE \cdot FCE$	$\{x \mid \exists y : (x, y) \in (FOPE)^{FOP}$ and $y \in (FCE)^{FC}\}$
$ObjectAllValuesFrom(FOPE\ FCE)$	$\forall \neq FOPE \cdot FCE$	$\{x \mid \forall y : (x, y) \in (FOPE)^{FOP}$ implies $y \in (FCE)^{FC}\}$
$ObjectHasValue(FOPE\ a)$	$\exists FOPE \cdot \{a\}$	$\{x \mid (x, (a)^{I}) \in (FOPE)^{FOP}\}$
$ObjectHasSelf(FOPE)$	$FOPE \equiv (FOPE)^{-}$	$\{x \mid (x, x) \in (FOPE)^{FOP}\}$
$ObjectMinCardinality(n\ FOPE)$	$\geq nFOPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FOPE)^{FOP}\} \geq n\}$
$ObjectMaxCardinality(n\ FOPE)$	$\leq nFOPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FOPE)^{FOP}\} \leq n\}$
$ObjectExactCardinality(n\ FOPE)$	$\equiv nFOPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FOPE)^{FOP}$ and $y \in (FOPE)^{FOP}\} = n\}$
$DataSomeValuesFrom(FDPE_1 ... FDPE_n\ FDR)$	$\exists FDR \cdot \{FDPE_1 ... FDPE_n\}$	$\{x \mid \exists y_1, ..., y_n : (x, y_k) \in (FDPE_k)^{FDP}$ for each $1 \leq k \leq n$ and $(y_1, ..., y_n) \in (FDR)^{FDT}\}$
$DataAllValuesFrom(FDPE_1 ... FDPE_n\ FDR)$	$\forall FDR \cdot \{FDPE_1 ... FDPE_n\}$	$\{x \mid \forall y_1, ..., y_n : (x, y_k) \in (FDPE_k)^{FDP}$ for each $1 \leq k \leq n$ imply $(y_1, ..., y_n) \in (FDR)^{FDT}\}$
$DataHasValue(FDPE\ lt)$	$\exists FDPE \cdot \{lt\}$	$\{x \mid (x, (lt)^{LT}) \in (FDPE)^{FDP}\}$
$DataMinCardinality(n\ FDPE)$	$\geq nFDPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FDPE)^{FDP}\} \geq n\}$
$DataMaxCardinality(n\ FDPE)$	$\leq nFDPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FDPE)^{FDP}\} \leq n\}$
$DataExactCardinality(n\ FDPE)$	$\equiv nFDPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FDPE)^{FDP}\} = n\}$
$DataMinCardinality(n\ FDPE\ FDR)$	$\geq nFDR \cdot FDPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FDPE)^{FDP}$ and $y \in (FDR)^{FDT}\} \geq n\}$
$DataMaxCardinality(n\ FDPE\ FDR)$	$\leq nFDR \cdot FDPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FDPE)^{FDP}$ and $y \in (FDR)^{FDT}\} \leq n\}$
$DataExactCardinality(n\ FDPE\ FDR)$	$\equiv nFDR \cdot FDPE$	$\{x \mid \sharp\{y \mid (x, y) \in (FDPE)^{FDP}$ and $y \in (FDR)^{FDT}\} = n\}$
Fuzzy Data Ranges		
$DataIntersectionOf(FDR_1 ... FDR_n)$	$FDR_1 \sqcap ... \sqcap FDR_n$	$(FDR_1)^{FDT} \cap ... \cap FDR_n)^{FDT}$
$DataUnionOf(FDR_1 ... FDR_n)$	$FDR_1 \sqcup ... \sqcup FDR_n$	$(FDR_1)^{FDT} \cup ... \cup FDR_n)^{FDT}$
$DataComplementOf(FDR)$	$\rightharpoondown FDR$	$(\triangle_n) \setminus (FDR)^{FDT}$ where n is the arity of *FDR*
$DataOneOf(lt_1 ... lt_n)$	$\{lt_1\} \sqcup ... \sqcup \{lt_n\}$	$\{(lt_1)^{LT}, ..., (lt_n)^{LT}\}$
$DatatypeRestriction(FDT\ F_1\ lt_1 ... F_n\ lt_n)$		$(FDR)^{FDT} \cap (F_1, lt_1)^{FA} \cap ... \cap (F_n, lt_n)^{FA}$
Fuzzy Class axioms		
$Class(FC\ partial\ FCE_1 ... FCE_n)$	$FC \sqsubseteq FCE_1 \sqcap ... \sqcap FCE_n$	$(FC)^{FC} \subseteq (FCE_1)^{FC} \cap ... \cap (FCE_n)^{FC}$
$SubClassOf(FCE_1\ FCE_2)$	$FCE_1 \sqsubseteq FCE_2$	$(FCE_1)^{FC} \subseteq (FCE_2)^{FC}$
$EquivalentClasses(FCE_1 ... FCE_n)$	$FCE_1 \equiv ... \equiv FCE_n$	$(FCE_j)^{FC} = (FCE_k)^{FC}$ for each $1 \leq j < k \leq n$
$DisjointClasses(FCE_1 ... FCE_n)$	$FCE_j \neq FCE_k 1 \leq j < k \leq n$	$(FCE_j)^{FC} \cap (FCE_k)^{FC} = \varnothing$ for each $1 \leq j < k \leq n$
$DisjointUnion(FC\ FCE_1 ... FCE_n)$	$FC \equiv (FCE_1 \sqcup ... \sqcup FCE_n), FCE_j \neq FCE_k, 1 \leq j < k \leq n$	$(FC)^{FC} = (FCE_1)^{FC} \cup ... \cup (FCE_n)^{FC}$ and $(FCE_j)^{FC} \cap (FCE_k)^{FC} = \varnothing$ for each $1 \leq j < k \leq n$
Fuzzy Object property axioms		
$SubObjectPropertyOf(FOPE_1\ FOPE_2)$	$FOPE_1 \sqsubseteq FOPE_2$	$(FOPE_1)^{FOP} \subseteq (FOPE_2)^{FOP}$
$EquivalentObjectProperties(FOPE_1 ... FOPE_n)$	$FOPE_1 \equiv ... \equiv FOPE_n$	$(FOPE_j)^{FOP} = (FOPE_k)^{FOP}$ for each $1 \leq j < k \leq n$
$DisjointObjectProperties(FOPE_1 ... FOPE_n)$	$FOPE_j \neq FOPE_k, 1 \leq j < k \leq n$	$(FOPE_j)^{FOP} \cap (FOPE_k)^{FOP} = \varnothing$ for each $1 \leq j < k \leq n$
$ObjectPropertyDomain(FOPE\ FCE)$	$\exists FOPE \cdot FCE$	$\forall x, y : (x, y) \in (FOPE)^{FOP}$ implies $x \in (FCE)^{FC}$
$ObjectPropertyRange(FOPE\ FCE)$	$\top \sqsubseteq \forall FOPE \cdot FCE$	$\forall x, y : (x, y) \in (FOPE)^{FOP}$ implies $x \in (FCE)^{FC}$
$InverseObjectProperties(FOPE_1\ FOPE_2)$	$FOPE_1 \equiv (FOPE_2)^{-}$	$(FOPE_1)^{FOP} = \{(x, y) \mid (y, x) \in (FOPE_2)^{FOP}\}$
$FunctionalObjectProperty(FOPE)$	$\top \sqsubseteq \leq 1FOPE$	$\forall x, y_1, y_2 : (x, y_1) \in (FOPE)^{FOP}$ and $(x, y_2) \in (FOPE)^{FOP}$ imply $y_1 = y_2$
$InverseFunctionalObjectProperty(FOPE)$	$\top \sqsubseteq \leq 1(FOPE)^{-}$	$\forall x_1, x_2, y : (x_1, y) \in (FOPE)^{FOP}$ and $(x_2, y) \in (FOPE)^{FOP}$ imply $x_1 = x_2$
$ReflexiveObjectProperty(FOPE)$	$FOPE \equiv (FOPE)^{-}$	$\forall x : x \in \triangle^{FI}$ implies $(x, x) \in (FOPE)^{FOP}$
$IrreflexiveObjectProperty(FOPE)$	$FOPE \neq (FOPE)^{-}$	$\forall x : x \in \triangle^{FI}$ implies $(x, x) \notin (FOPE)^{FOP}$
$SymmetricObjectProperty(FOPE)$	$FOPE \equiv (FOPE)^{FT}$	$\forall x, y : (x, y) \in (FOPE)^{FOP}$ implies $(y, x) \in (FOPE)^{FOP}$
$AsymmetricObjectProperty(FOPE)$	$FOPE \neq (FOPE)^{FT}$	$\forall x, y : (x, y) \in (FOPE)^{FOP}$ implies $(y, x) \notin (FOPE)^{FOP}$
$TransitiveObjectProperty(FOPE)$	$(FOPE)^{2} \sqsubseteq FOPE$	$\forall x, y, z : (x, y) \in (FOPE)^{FOP}$ and $(y, z) \in (FOPE)^{FOP}$ implies $(x, z) \in (FOPE)^{FOP}$
Fuzzy Data property Axioms		
$SubDataPropertyOf(FDPE_1\ FDPE_2)$	$FDPE_1 \sqsubseteq FDPE_2$	$(FDPE_1)^{FDP} \subseteq (FDPE_2)^{FDP}$
$EquivalentDataProperties(FDPE_1 ... FDPE_n)$	$FDPE_1 \equiv ... \equiv FDPE_n$	$(FDPE_i)^{FDP} = (FDPE_j)^{FDP}$ for each $1 \leq j < k \leq n$
$DisjointDataProperties(FDPE_1 ... FDPE_n)$	$FDPE_j \neq FDPE_k, 1 \leq j < k \leq n$	$(FDPE_i)^{FDP} \cap (FDPE_j)^{FDP} = \varnothing$, for each, $1 \leq j < k \leq n$
$DataPropertyDomain(FDPE\ FCE)$	$\exists FDPE.\top \sqsubseteq FCE$	$\forall x, y : (x, y) \in (FDPE)^{FDP}$ implies $x \in (FCE)^{FDP}$
$DataPropertyRange(FDPE\ FDR)$	$\top \sqsubseteq \forall FDPE.FCE$	$\forall x, y : (x, y) \in (FDPE)^{FDP}$ implies $y \in (FDR)^{FDP}$
$FunctionalDataProperty(FDPE)$	$\top \sqsubseteq \leq 1FDPE$	$\forall x, y_1, y_2 : (x, y_1) \in (FDPE)^{FDP}$ and $(x, y_2) \in (FDPE)^{FDP}$ implies $y_1 = y_2$
Fuzzy Assertions Axioms		
$SameIndividual(a_1 ... a_n)$	$\{a_j\} \equiv ... \equiv \{a_k\}$	$(a_j)^{FI} = (a_k)^{FI}\ 1 \leq j < k \leq n$
$DifferentIndividuals(a_1 ... a_n)$	$\{a_j\} \neq \{a_k\} 1 \leq j < k \leq n$	$(a_j)^{FI} \neq (a_k)^{FI}\ 1 \leq j < k \leq n$
$ClassAssertion(FCE\ a)$	$\exists FCE \cdot \{a\}$	$(a)^{FI} \in (FCE)^{FC}$
$ObjectPropertyAssertion(FOPE\ a_1\ a_2)$	$\exists FOPE \cdot \{a_1, a_2\}$	$((a_1)^{FI}, (a_2)^{FI}) \in (FOPE)^{FOP}$
$NegativeObjectPropertyAssertion(FOPE\ a_1\ a_2)$	$\nexists FOPE \cdot \{a_1, a_2\}$	$((a_1)^{FI}, (a_2)^{FI}) \notin (FOPE)^{FOP}$
$DataPropertyAssertion(FOPE\ a\ lt)$	$\exists FOPE \cdot \{a, lt\}$	$(a^{FI}, (lt)^{FI}) \in (FOPE)^{FOP}$
$NegativeDataPropertyAssertion(FOPE\ a\ lt)$	$\nexists FOPE \cdot \{a, lt\}$	$(a^{FI}, (lt)^{FI}) \notin (FOPE)^{FOP}$

The abstract domain $\triangle^{FI}$ is a set of objects, the datatype domain $\triangle^{FD}$ is the domain of interpretation of all datatypes (*disjoint* from $\triangle^{FI}$) consisting of data values, and $\bullet^{FI}$ and $\bullet^{FD}$ are two fuzzy interpretation functions. These two functions can map:

- An abstract individual o to an element $o^{FI} \in \triangle^{FI}$,
- For individuals o_1 and o_2, if $o_1 \neq o_2$, $o_1^{FI} \neq o_2^{FI}$,
- A concrete individual v to an element $v^{FD} \in \triangle^{FD}$,
- A concept name FA to a membership degree function $FA^{FI} : \triangle^{FI} \rightarrow [0, 1]$,

- An abstract role name R to a membership degree function $R^{FI}: \Delta^{FI} \times \Delta^{FI} \rightarrow [0, 1]$,

- A concrete datatype FD to a membership degree function $FD^{FD}: \Delta^{FD} \rightarrow [0, 1]$,

- A concrete role name FT to a membership degree function $FT^{FI}: \Delta^{FI} \times \Delta^{FD} \rightarrow [0, 1]$.

A fuzzy ontology formulated in fuzzy OWL 2 language is called fuzzy OWL 2 ontology. Several definitions of fuzzy ontologies are proposed based on the language fuzzy OWL (e.g. [34], [35]). In order to represent both the structure and instance information of fuzzy OWL 2 ontologies, we present a formal definition of fuzzy OWL 2 ontologies in the following, which considers both the structure and instance information of fuzzy ontologies.

Definition 6 (Fuzzy OWL 2 Ontology): A fuzzy OWL ontology is formally represented as 8-tuple $FO = (FOP_O, FDP_O, FC_O, FDT_O, FDR_O, FI_O, Flt_O, FO_{Axiom})$, consisting of the following elements [21]:

1) FOP_O is a set of object properties identifiers linking individuals to individuals, and each property may have its characters and its restrictions;

2) FDP_O is a set of data properties linking individuals to data values;

3) FC_O is a set of fuzzy class defined in the OWL 2. Each class can be an *AbstractClass* or a *ConcreteClass*;

4) FDT_O is a set datatype, containing the datatype *rdfs:Literal* and possibly other datatypes;

5) FDR_O is a set containing all data range;

6) FI_O is a collection of fuzzy individuals (*named* and *anonymous*);

7) Flt_O is a literal containing each datatype FDT_O and each lexical form of Flt_O;

8) FO_{Axiom} is a set of finite fuzzy OWL 2 axioms.

In summary, a fuzzy OWL 2 ontology FO includes two parts: the structure and the instance. Now we illustrate a fuzzy OWL 2 ontology of *E-commerce* in an abstract syntax in *Figure 3*. There are several kinds of fuzziness in the *E-commerce* fuzzy ontology.

The element *Corporate-Customer* may be fuzzy since we cannot precisely describe the element. In this case, we provide an attribute $\mu \in [0, 1]$ in the axiom of the element *Corporate-Customer*.

A fuzzy keyword *FUZZY* indicates an attribute to be fuzzy values. For example, the attribute *FUZZY-creditRating* of the element *Corporate-Customer* may be fuzzy. Moreover, there may be other fuzzy elements and attributes in the fuzzy ontology *E-commerce*.

A fuzzy OWL 2 ontology structure of customers of E-commerce:

FO_{Axiom} = {
Class (Corporate-Customer partial Customer);
Class (Personal-Customer partial Customer);
EquivalentClasses(Customer, unionOf (Corporate-Customer, Personal-Customer));
DisjointClasses (Corporate-Customer, Personal-Customer);
Class(Customer complete intersectionOf partial(Name Address μ));
ObjectProperty (*Customer*hasop*Name* domain (Customer) range (Name) [Functional]);
ObjectProperty (*Customer*hasop*Address* domain (Customer) range (Address));
Class (Customer partial restriction (*Customer*hasop*Name* allValuesFrom Name) Cardinality (1)) restriction (*Customer*hasop*Address* allValuesFrom (Address) minCardinality (1)));
Class (Name partial restriction (*Name*hasdp*PCDATA*) allValuesFrom (xsd:String) cardinality (1)));
DatatypeProperty (*Name*hasdp*PCDATA*) domain (Name) range (xsd:String) [Functional]);
Class (Address partial restriction (*Address*hasdp*PCDATA*) allValuesFrom (xsd:String) cardinality (1)));
DatatypeProperty (*Address*hasdp*PCDATA*) domain (Address) range (xsd:String) [Functional]);

Class (Personal-Customer partial restriction (*Personal-Customer*hasop*CardNo*) allValuesFrom (CardNo) minCardinality (1)));
ObjectProperty (*Personal-Customer*hasop*CardNo* domain (Personal-Customer) range (CardNo));
Class (CardNo partial restriction (*CardNo*hasdp*PCDATA*) allValuesFrom (xsd:String) cardinality (1)));
DatatypeProperty (*CardNo*dp*PCDATA*) domain (CardNor) range (xsd:String) [Functional]);

Class (Corporate-Customer complete intersectionOf (FUZZY-CreditRating FUZZY-Discount μ));
Class (Corporate-Customer partial restriction (*Corporate-Customer*hasopFUZZY-*CreditRating*) allValuesFrom (FUZZY-CreditRating) maxCardinality (1)) restriction (*Corporate-Customer*hasop FUZZY-*Discount*) allValuesFrom (FUZZY-Discount) maxCardinality (1)));
ObjectProperty (*Corporate-Customer*hasopFUZZY-*CreditRating* domain (Corporate-Customer) range (FUZZY-CreditRating));
ObjectProperty (*Corporate-Customer*hasopFUZZY-*Discount* domain (Corporate-Customer) range (FUZZY-Discount));
Class (FUZZY-CreditRating partial restriction (*FUZZY-CreditRating*hasdp*PCDATA*) allValuesFrom (xsd:single) cardinality (1)));
DatatypeProperty (*FUZZY-CreditRating*dp*PCDATA*) domain (FUZZY-CreditRating) range (xsd:single) [Functional]);
Class (FUZZY-Discount partial restriction (*FUZZY-Discount*hasdp*PCDATA*) allValuesFrom (xsd:String) cardinality (1)));
DatatypeProperty (*FUZZY-Discount*hasdp*PCDATA*) domain (FUZZY-Discount) range (xsd:String) [Functional]);
…
SubClassOf(Corporate-customer, Customer);
SubClassOf(Personal-customer, Customer);
}

FIGURE 3. A fuzzy OWL ontology in the abstract syntax.

III. TRANSFORMING FUZZY OWL 2 ONTOLOGIES TO FUZZY XML MODEL

A. TRANSFORMING FUZZY OWL 2 ONTOLOGY INTO FUZZY XML DTD AT STRUCTURE LEVEL

In the following, *Definition 7* firstly propose the formal approach for converting a fuzzy OWL 2 ontology into a fuzzy XML DTD. Then, *Theorem 1* proves the correctness of the approach. Finally, we provide a transformation example. All of these will help to understand how to transform fuzzy OWL 2 ontologies to fuzzy XML DTD.

Giving a fuzzy OWL 2 Ontology model $FO = (FOP_O, FDP_O, FC_O, FDT_O, FDR_O, FI_O, Flt_O, FO_{Axiom})$, *Definition 7* transforms the fuzzy OWL 2 ontology FO to fuzzy XML DTD elements and attributes.

Definition 7 (Structure Transformation): Given a fuzzy OWL 2 ontology $FO = (FOP_O, FDP_O, FC_O, FDT_O, FDR_O, FI_O, Flt_O, FO_{Axiom})$. The fuzzy XML DTD D= (**P**, r)

TABLE 2. Transforming rules from a fuzzy OWL 2 ontology to fuzzy XML DTD structure.

Fuzzy OWL 2 ontology FO	**Fuzzy XML DTD structure**
Each fuzzy individual identifier FI_O	A fuzzy DTD identifier $\varphi(FI_O) \in D$ =(**P**,r)
Each fuzzy datatype property identifier FDP_O	An attribute $\varphi(FDP_O) \in$ **A**,
Each fuzzy class identifier FC_O	An element symbol identifier $\varphi(FC_O) \in$ **E**
Each datatype identifier FDT_O	FDT_O are analogous the set of XML Schema Datatypes
Each date range identifier FDR_O	Types of attribute or element $\varphi(FDR_O) \in$ **T**
Each literal Flt_O	Representation data value $\varphi(Flt_O) \in$ **S**
Annotation properties	Annotation information such as documentation
Fuzzy OWL 2 axiom set FO_{Axiom}	**Fuzzy XML DTD contents**
Class(FC partial...restriction(FDR_i allValuesFrom(FDT_i) Cardinality(1))...); DatatypeProperty (FDR_i domain (FC) range (FDT_i) [Functional]).	Creating an element type definition $\varphi(FC) \to \varphi(FDR)$, $\varphi(FC) \in E$, $\varphi(FDR) \in$ **T**, where $FDR \in FDR_1 \cup ... \cup FDR_n$, $FDT \in FDT_1 \cup ... \cup FDT_n$.
Class(FC partial restriction (FDP_1 allValuesFrom(FDR_1) minCardinality(m_1) maxCardinality(n_1))...restriction FDP_k allValuesFrom(FDR_k minCardinality(m_k) maxCardinality(n_k)); DatatypeProperty (FDP_1 domain (FC) range(FDR_1))... DatatypeProperty (FDP_k domain (FC) range(FDR_k)).	Creating an attribute type definition $\varphi(FC) \to \varphi(FDP)$, $\varphi(FC) \in E$, $\varphi(FDP) \in A$, $\varphi(FDP_i) \in AN$, $\varphi(FDR_i) \in AT$, $\varphi(m_i), \varphi(n_i) \in VT$,where $FDP \in FDP_1 \cup ... \cup FDP_k$,$FDR \in FDR_1 \cup ... \cup FDR_k$, if *VT* = ' #IMPLIED' , then m_i = 0 or 1; if *VT* =' #REQUIRED', then m_i = 1; if *VT* = ' #Fixed value' or 'value', then FDR_i is the 'value'.
Class (FC partial restriction (FOP_i allValuesFrom (FDR_i) Cardinality (1))); ObjectProperty (FOP_i domain (FC) range (FDR_i) [Functional]).	Creating an attribute type definition $\varphi(FC) \to \varphi(FOP_i)$, $\varphi(FC) \in$ **E**, $\varphi(FDR_i) \in$ **T**, $\varphi(FOP_i) \in \alpha$.
Class(FC partial restriction (FOP_i allValuesFrom (FDP_i) maxCardinality (1))); ObjectProperty (FOP_i domain(FC) range (FDP_i)).	Creating an attribute type definition $\varphi(FC) \to \varphi(FOP_i)^?$, $\varphi(FC) \in$ **E**, $\varphi(FDP_i) \in$ **A**, $\varphi(FOP_i) \in \alpha$, where '?' denotes 0 or 1 time.
Class (FC partial restriction (FOP_i allValuesFrom (FDP_i))); ObjectProperty (FOP_i domain (FC) range (FDP_i)).	Creating an attribute type definition $\varphi(FC) \to \varphi(FOP_i)^*$, $\varphi(FC) \in$ **E**, $\varphi(FDP_i) \in$ **A**, $\varphi(FOP_i) \in \alpha$, where '*' denotes 0 or n times.
Class (FC partial restriction (FOP_i allValuesFrom (FDP_i) minCardinality (1))); ObjectProperty (FOP_i domain (FC) range (FDP_i)).	Creating an attribute type definition $\varphi(FC) \to \varphi(FOP_i)^+$, $\varphi(FC) \in$ **E**, $\varphi(FDP_i) \in$ **A**, $\varphi(FOP_i) \in \alpha$, where '+' denotes 1 or n times.
Class (FC partial owl: Nothing).	Creating an attribute type definition $\varphi(FC) \to$ empty.
Class(FC complete unionOf (intersectionOf (FDP_1 complementOf (FDP_2)) intersectionOf (complementOf (FDP_1 FDP_2); Class (FC partial restriction (FOP_1 allValuesFrom (FDP_1)) restriction (FOP_2 allValuesFrom (FDP_2))); ObjectProperty (FOP_1 domain (FC) range (FDP_1)); ObjectProperty (FOP_2 domain (FC) range (FDP_2)).	An attribute type definition $\varphi(FC) \to \varphi(FOP_1 \vert \varphi(FOP_2)$, where $\varphi(FC) \in$ **E**, $\varphi(FDP_1), \varphi(FDP_2) \in$ **A**, $\varphi(FOP_1) \in \alpha_1, \varphi(FOP_2) \in \alpha_2$, $\alpha_1 \cap \alpha_2 = \varnothing$.
Class(FC complete intersectionOf (FDP_1, FDP_2));Class (FC partial restriction (FOP_1 allValuesFrom (FDP_1)) restriction (FOP_2 allValuesFrom (FDP_2))); ObjectProperty (FOP_1 domain (FC) range (FDP_1)); ObjectProperty (FOP_2 domain (FC) range (FDP_2)).	An attribute type definition $\varphi(FC) \to \varphi(FOP_1, \varphi(FOP_2)$, where $\varphi(FC) \in$ **E**, $\varphi(FDP_1), \varphi(FDP_2) \in$ **A**, $\varphi(FOP_1) \in \alpha_1, \varphi(FOP_i) \in \alpha_2$, $\alpha_1 \cap \alpha_2 = \varnothing$.

can be derived by transformation function $\varphi()$ as shown in *Table 2*.

Applying the rules in *Table 2*, we can finally obtain the fuzzy DTD correspond to the fuzzy OWL 2 ontology structure in *Figure 3*. The corresponding fuzzy XML DTD model shown in *Figure 4*.

B. TRANSFORMATION FUZZY OWL 2 ONTOLOGY TO FUZZY XML DOCUMENT AT INSTANCE LEVEL

In this section, we propose some rules in *Table 3* to transform a fuzzy OWL 2 ontology instance into fuzzy XML document based on the constructed DTD in *Section A*. Given a fuzzy OWL 2 ontology instance o, the corresponding fuzzy XML document $\varphi(o) = (N, <, \lambda, \eta, \gamma)$ can be derived from the following rules in *Table 3*.

C. THE CORRECTNESS OF THE TRANSFORMATION APPROACH

The *Sections A* and *B* specify some mapping rules that can transform fuzzy OWL 2 ontology structure and instance to fuzzy XML DTD and document. In this section, we discuss the correctness of the approach. Then we establish mapping instance of fuzzy OWL 2 ontology and fuzzy XML document and DTD.

Theorem 1: For every fuzzy OWL 2 ontology *FO* and its transformed fuzzy DTD $\varphi(FO)$, there is two mappings δ from fuzzy OWL 2 ontologies structure to models $\varphi(FO)$, and ζ from models $\varphi(FO)$ to fuzzy OWL 2 ontology structure, such that:

- For each fuzzy OWL 2 ontology instance *FI* conforming to *FO*, $\delta(FI)$ is a model of fuzzy $\varphi(FO)$.
- For each model d of $\varphi(FO)$, $\zeta(d)$ is a fuzzy OWL 2 ontology instance.

Proof: Between start tags and end tags, a fuzzy XML document contains several elements which are associated with their attribute values. There is two alphabet **T** and **E**, they are basic types and element types. A fuzzy XML document instances $d_{T,E}$ builts over **T** and **E** as follows: (i) If d is a terminal in **T**, then $d_i \in d_{T,E}$; (ii) If d is sequence of the form $< E > d_1, \ldots, d_k < /E >$, where $E \in$ **E** is an element type and $d_1, \ldots, d_k \in d_{T,E}$, then $d \in d_{T,E}$.

Then the following first proves the first part of *Theorem 1*. Let $FI = (\Delta^{FI}, \bullet^{FI})$ be a fuzzy interpretation of fuzzy OWL 2 ontology *FO*, and $o \in \Delta^{FI}$ be an ontology instance, then we can obtain an fuzzy DTD instance model $\delta(o)$, as follow:

(a) If $o \in T^{FI}$ for some terminal $T \in$ **T**, then $\delta(o) = T$;

(b) If for some $E \in$ **E**, there are some integer $n \geq 0$, and objects o_s, o_i, o_i', and o_e, such as $o_s \in StartE^{FI}$,

TABLE 3. Transforming rules from a Fuzzy OWL 2 ontology instance to fuzzy xml document.

Fuzzy OWL 2 ontology instance	Fuzzy XML document
Each fuzzy individual instance axioms: Individual (o type(FC) [$\bowtie$ m_i]), here: o is a fuzzy ontology instance, FC is a fuzzy class identifier, the part [$\bowtie$ m_i] in the fuzzy individual axiom is omitted in case of m_i=1.0.	Creating a node of fuzzy XML document tree $\varphi(o) \in \mathbf{N}$, $\lambda(\varphi(o)) \in \mathbf{E}$. If there is [$\bowtie$ m_i] in fuzzy OWL 2 ontology individual instance, we create a tag **<Val Poss** = m_i > ...**</Val>**.
Fuzzy individual instance axioms: Individual(o_i type (FC_i)) and Individual (o_j type (FC_j) value(FOP_{ji}, o_i)), here o_i, o_j are fuzzy ontology instance, FC_i, FC_j are a fuzzy class identifier, FOP_{ji} identifies that instance o_j has object property instance o_i. Furthermore if there are following instance axioms: Case 1: Individual (o_i type (FC_i) value (FDP_i, FDT_i)), here FDP_i identifies the datatype property of instance o_i, FDT_i identifies a set of datatype. Case 2: Individual (o_i type (FC_i) value (FDP_i, FDT_i) [$\bowtie$ m_i]), here FDP_i identifies the datatype property of instance o_i, FDT_i identifies a set of datatype. Case 3: Individual (o_h type (FC_h) value (FOP_{hj}, o_i) [$\bowtie$ m_i]), here oh is a fuzzy ontology instance, FC_h is a fuzzy class identifier, FOP_{hj} identifies that instance o_h has property instance o_j. Case 4: Individual (o_i type (FC_i) value (FOP_{ik}, o_k) [$\bowtie$ m_i]), here o_k is a fuzzy ontology instance, FOP_{ik} identifies that instance o_i has property instance o_k.	Creating nodes of fuzzy XML document tree $\varphi(o_i), \varphi(o_j) \in N$, element symbol $\lambda(\varphi(o_i)), \lambda(\varphi(o_j)), \varphi(FC_i), \varphi(FC_j) \in \mathbf{E}$, $\varphi(o_j)$ is the parent node of $\varphi(o_i)$, i.e. $\varphi(o_j) < \varphi(o_i)$. According of the condition Case 1-4, we create the following elements: Case 1: $\lambda(\varphi(o_i))$ is a leaf node. Case 2: If $\lambda(\varphi(o_i))$ is a leaf node, and $\lambda(\varphi(o_i)) = Val_i$ and $\lambda(\varphi(o_j)) = Dist_j$, then $m_i = \eta(\varphi(o_i), @Poss_i)$. Case 3: If there are $\lambda(\varphi(o_i)) = Val_i$ and $\lambda(\varphi(o_j)) \neq Dist_j$, then $\varphi(o_h)$ is children node of $\varphi(o_j)$, i.e. $\varphi(o_h) < \varphi(o_j) < \varphi(o_i)$, $m_i = \eta(\varphi(o_i), @Poss_i)$. Case 4: If there are $\lambda(\varphi(o_i)) = Val_i$ and $\lambda(\varphi(o_j)) = Dist_j$, then $\varphi(o_k)$ is parent node of $\varphi(o_j)$, i.e. $\varphi(o_j) < \varphi(o_i) < \varphi(o_k)$,$m_i = \eta(\varphi(o_i), @Poss_i)$. Here $\varphi(o_h), \varphi(o_k) \in N$, $\varphi(FOP_{ji}) \in \varphi(o_j)$hasop$\varphi(o_i)$, same as $\varphi(FOP_{hj}))$,$\varphi(FOP_{ik})$. $\varphi(FDP_i) \in \varphi(o_i)$hasdp*PCDATA*, $\varphi(FDT_i) \in \eta(\varphi(o_i))$, $\varphi(o_j) = d_j \in$ **dom** is content of the element $\lambda(\varphi(o_j))$.
Fuzzy individual instance axioms: Individual (o_i type(FC_i) value (FOP_{ij}, FDT_{ij})), here FOP_{ij} identifies that instance o_i has FOP_j, FDT_{ij} identifies a set of datatype.	Creating nodes of fuzzy XML document tree $\varphi(o_i), \varphi(o_j) \in N$, element symbol $\varphi(FC_i) \in \mathbf{E}$, and $\varphi(o_j)$ is the parent node of $\varphi(o_i)$, i.e. $\varphi(o_j) < \varphi(o_i)$, $\varphi(FOP_{ij}) \in \varphi(o_i)$hasop$\alpha_j$, $\varphi(FDT_i) \in \eta(\varphi(o_i), \varphi(o_j)) = d_j \in$**dom** is a value of the property.

```
<!ELEMENT Customer (Dist)>
    <!ATTLIST Customer CID ID #REQUIRED>
<!ELEMENT Dist (Val+)>
    <!ATTLIST Dist type (disjunctive)>
     <!ELEMENT Val (Name?, Address?, Vla+)>
     <!ELEMENT Name (#PCDATA)>
     <!ELEMENT Address (#PCDATA)>
      <!ATTLIST Val Poss CDATA "1.0">
     <!ELEMENT Val (Corporate-Customer*, Personal-
Customer*)>
      <!ATTLIST Val Poss CDATA "1.0">
     <!ELEMENT Corporate-Customer (Dist)>
      <!ATTLIST Corporate-Customer FID IDREF #REQUIRED>
<!ELEMENT Dist (Val+)>
    <!ATTLIST Dist type (disjunctive)>
<!ELEMENT Val (Discount?, CreditRating?)>
    <!ATTLIST Val Poss CDATA "1.0">
<!ELEMENT Discount (Dist)>
<!ELEMENT Dist (Val+)>
   <!ATTLIST Dist type (disjunctive)>
<!ELEMENT Val (Discount_value)>
   <!ATTLIST Val Poss CDATA "1.0">
<!ELEMENT Discount_value (#PCDATA)>
<!ELEMENT CreditRating (Dist)>
<!ELEMENT Dist (Val+)>
   <!ATTLIST Dist type (conjunctive)>
<!ELEMENT Val (CreditRating_value)>
   <!ATTLIST Val Poss CDATA "1.0">
<!ELEMENT CreditRating_value (#PCDATA)>
     <!ELEMENT Personal-Customer (Dist)>
    <!ATTLIST Personal-Customer FID IDREF #REQUIRED>
<!ELEMENT Dist (Val+)>
    <!ATTLIST Dist type (disjunctive)>
<!ELEMENT Val (CardNo*)>
<!ELEMENT CardNo (#PCDATA)>
```

FIGURE 4. Fuzzy XML DTD model derived from fuzzy OWL 2 Ontology in Figure 3.

$o_e \in EndE^{FI}$, $(o, o_s), (o_1, o_1'), \ldots, (o_n, o_n') \in \mathbf{f}^{FI}$, and $(o, o_1), (o_1, o_2), \ldots, (o_{n-1}, o_n), (o_n, o_e) \in \mathbf{r}^{FI}$, then $\delta(o) =< E > d_1, \ldots, d_k < /E >$, where (i) two atomic fuzzy class identifiers *StartE* and *EndE* are needed to represent respectively the start tag and end tag of E; (ii) o_s and o_e denote the start and end tags of the root element, o_i denotes the *i-th* component of d, and o_i' is the root of d_i, $i \in \{1, \ldots, n\}$; (iii) in the model $\delta(o)$, for the sake of simplicity, **f** and **r** are used to denote the fuzzy property identifiers constructed, where **f** represents the start tag of an element and **r** represents the other components of the element in the tree structure of the fuzzy XML document d.

And the second part of *Theorem 1* can be proved similarly for the first part above, it is a mutually inverse process. Let $d \in d_{T,E}$ be a fuzzy XML document, then we can obtain a model $\zeta(d) = (\triangle^{\zeta(d)}, \bullet^{\zeta(d)})$ satisfying the fuzzy axioms of *FO*, *as follow*:

(a) If d is a terminal $T \in \mathbf{T}$, then $\triangle^{\zeta(d)} = (\varphi(T))^{\zeta(d)}$;

(b) If d is a sequence of form $< E > d_1, \ldots, d_k < /E >$, where d_i is an instance satisfying to the fuzzy DTD model $E \rightarrow (\alpha, A)$, then a tree-model $\zeta(d)$ can be constructed as follows:

$$\triangle^{\zeta(d)} = \{o, o, o_1, \ldots, o_n, o_e\} \cup \triangle^{\zeta(d_i)}$$

$$StartE^{\zeta(d)} = \{o_s\} \cup \bigcup_{1 \le i \le n} StartE^{\zeta(d_i)}$$

$$EndE^{\zeta(d)} = \{o_e\} \cup \bigcup_{1 \le i \le n} EndE^{\zeta(d_i)}$$

$$Tag^{\zeta(d)} = \{o_s, o_e\} \cup \bigcup_{1 \le i \le n} Tag^{\zeta(d_i)}$$

$$r^{\zeta(d)} = \{(o, o_1), (o_1, o_2), \ldots, (o_{n-1}, o_n), (o_n, o_e)\} \cup \bigcup_{1 \le i \le n} r^{\zeta(d_i)}$$

$$f^{\zeta(d)} = \{(o, o_s), (o_1, o_1'), \ldots, (o_n, o_n')\} \cup \bigcup_{1 \le i \le n} f^{\zeta(d_i)}$$

Given a fuzzy OWL 2 ontology instance *o*
$FAxiom_0$ = {
Individual (o_1 Type (Customer) Value(*Customer*hasop*CID*, o_{21}));
Individual (o_{21} Type (CID) Value(*CID*hasdp*PCDATA*, "2015050908"));
Individual (o_1 Type (Customer) Value(*Customer*hasop*Name*, o_{22}));
Individual (o_{22}Type (Name) Value(*Name*hasdp*PCDATA*, "Smith"));
Individual (o_1 Type (Customer) Value(*Customer*hasop*Address*, o_{23}));
Individual (o_{23} Type (Address)
Value(*Address*hasdp*PCDATA*, "Industries 204 Main St. Chicago"));
Individual (o_1 Type (Customer) Value(*Customer*hasopVal_{24}, o_{24}));
Individual (o_{24} Type (Val_{24})
Value(Val_{24}hasop*Personal-Customer*, o_{31})
Value(Val_{24}hasop*Corporate-Customer*, o_{32})
Value(Val_{24}hasop*Poss*, "0.85"));
Individual (o_{31} Type (Personal-Customer)
Value(*Personal-Customer*hasop*CardNo*, o_{41}));
Individual (o_{41} Type (CardNo)
Value(*CardNo*hasdp*PCDATA*, "1111 2222 3333 4444"));
Individual (o_{32} Type (Corporate-Customer)
Value(*Corporate-Customer*hasop*CreditRating*, o_{42})
Value(*Corporate-Customer*hasop*Discount*, o_{43}));
Individual (o_{42} Type (CreditRating)
Value(*CreditRating*hasop$Dist_{51}$, o_{51}));
Individual (o_{51} Type ($Dist_{51}$)
Value($Dist_{51}$hasopVal_{61}, o_{61})
Value($Dist_{51}$hasop*Type*, "conjunctive"));
Individual (o_{61} Type (Val_{61})
Value(Val_{61}hasop*Poss*, "0.6")
Value(Val_{61}hasop*CreditRating_value*, o_{71}) [$\bowtie$ 0.6]);
Individual (o_{71} Type (CreditRating_value)
Value(*CreditRating_value*hasdp*PCDATA*, *"C"*) [$\bowtie$ 0.6]);
Individual (o_{62} Type (Val_{62})
Value(Val_{62}hasop*Poss*, "0.85")
Value(Val_{62}hasop*CreditRating_value*, o_{72}) [$\bowtie$ 0.85]);
Individual (o_{72} Type (CreditRating_value)
Value(*CreditRating_value*hasdp*PCDATA*, *"D"*) [$\bowtie$ 0.85]);
Individual (o_{43} Type (Discount)
Value(*Discount*hasop$Dist_{52}$, o_{52}));
Individual (o_{52} Type ($Dist_{52}$)
Value($Dist_{52}$hasopVal_{63}, o_{63})
Value($Dist_{52}$hasop*Type*, "disjunctive"));
Individual (o_{63} Type (Val_{63})
Value(Val_{63}hasop*Poss*, "0.95")
Value(Val_{63}hasop*Discount_value*, o_{73}) [$\bowtie$ 0.95]);
Individual (o_{73} Type (Discount_value)
Value(*Discount_value*hasdp*PCDATA*, *"80%"*) [$\bowtie$ 0.95]);
Individual (o_{64} Type (Val_{64})
Value(Val_{64}hasop*Poss*, *"0.8"*)
Value(Val_{64}hasop*Discount_value*, o_{74}) [$\bowtie$ 0.8]);
Individual (o_{74} Type (Discount_value)
Value(*Discount_value*hasdp*PCDATA*, *"90%"*) [$\bowtie$ 0.85]);
…}

FIGURE 5. A fuzzy OWL 2 ontology instance o.

```
<Customer>
  <CID>2015050908</CID>
  <Name>Smith</Neme>
  <Address>BCBA Industries 204 Main St. Chicago</Address >
  <Val Poss = 0.85>
    < Corporate-Customer >
      <Discount>
        <Dist type = "disjunctive">
          <Val Poss = 0.95>
            < Discount_value>80%</ Discount_value>
          </Val>
          <Val Poss = 0.80>
            < Discount_value>90%< Discount_value>
          </Val>
        </Dist>
      </Discount >
      <CreditRating>
        <Dist type = "conjunctive">
          <Val Poss = 0.60>
            < CreditRating_value>"C" </ CreditRating_value>
          </Val>
          <Val Poss = 0.85>
            < CreditRating_value>"B"</ CreditRating_value>
          </Val>
        </Dist>
      </CreditRating >
    </Corporate-Customer >
    <Personal-Customer >
      <CardNo >1111 2222 3333 4444</CardNo >
    </Personal -Customer >
  </Val>
</ Customer >
```

FIGURE 6. The fuzzy XML document derived from the fuzzy OWL 2 ontology instance in Figure 5.

So far, we propose the approach that can map a fuzzy OWL 2 ontology to a fuzzy XML model. As shown in *Section A*, constructing a fuzzy XML model from a fuzzy ontology has two steps: transforming the structure of fuzzy ontology into a fuzzy DTD and transforming the fuzzy ontology instance into the fuzzy XML document conforming the fuzzy DTD. For the first step, *Table 2* provides several rules of transforming all the fuzzy OWL 2 ontology identifiers and axioms into symbols of a fuzzy DTD. For the second step, *Table 3* provides some rules of transforming of instance level from the fuzzy ontology into the fuzzy XML model based on the structure in the first step.

D. A TRANSFORMING EXAMPLE FROM FUZZY OWL 2 ONTOLOGY TO FUZZY XML DOCUMENT

In order to explain the transforming approach well, we provide a fuzzy OWL 2 ontology instance in *Figure 5*, and the fuzzy XML document derived from the instance is shown in *Figure 6*.

IV. CONCLUSIONS

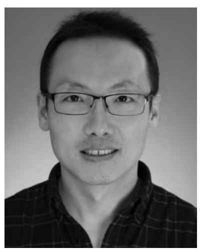
XML has been the standard for data representation and exchange based on the Web. Meanwhile, information is imprecise and uncertain in the real world. Then, fuzzy XML model has been proposed. In this paper, we mainly investigate fuzzy OWL 2 ontology and fuzzy XML model. Their formal definitions are proposed. Furthermore, we propose an approach of transforming fuzzy OWL 2 ontology into fuzzy XML model at structure and instance levels, respectively. The correctness of the approach is proved, and a transformation example is provided to well explain the proposed approach. In the future, we will evaluate the reusing fuzzy OWL 2 ontologies approach with more complex examples based on fuzzy XML model.

REFERENCES

[1] S. Abiteboul, L. Segoufin, and V. Vianu, "Representing and querying XML with incomplete information," in *Proc. 12th ACM SIGACT-SIGMOD-SIGART Symp. Principles Database Syst.*, 2001, pp. 150–161.
[2] S. M. Benslimane, M. Malki, and D. Bouchiha, "Deriving conceptual schema from domain ontology: A Web application reverse engineering approach," *Int. Arab J. Inf. Technol.*, vol. 7, no. 2, pp. 167–176, 2010.
[3] E. Bertino and B. Catania, "Integrating XML and databases," *IEEE Internet Comput.*, vol. 5, no. 4, pp. 84–88, Jul. 2001.
[4] F. Bobillo and U. Straccia, "Fuzzy ontology representation using OWL2," *Int. J. Approx. Reasoning*, vol. 52, no. 7, pp. 1073–1094, 2011.
[5] F. Bobillo, "Managing vagueness in ontologies," Ph.D. dissertation, Dept. Ciencias Computación e I. A., Univ. Granada, Granada, Spain, 2008.
[6] R. Conrad, D. Scheffner, and J. C. Freytag, "XML conceptual modeling using UML," in *Proc. Int. Conf. Conceptual Modeling*, 2000, pp. 558–571.
[7] M. Carey, J. Kiernan, J. Shanmugasundaram, E. J. Shekita, and S. N. Subramanian, "XPERANTO: Middleware for publishing object-relational data as XML documents," in *Proc. 26th Int. Conf. Very Large Data Bases (VLDB)*, 2000, pp. 646–648.
[8] A. Gaurav and R. Alhajj, "Incorporating fuzziness in XML and mapping fuzzy relational data into fuzzy XML," in *Proc. ACM Symp. Appl. Comput.*, 2006, pp. 456–460.
[9] M. Hacherouf, S. N. Bahloul, and C. Cruz, "Transforming XML documents to OWL ontologies: A survey," *J. Inf. Sci.*, vol. 41, no. 2, pp. 242–259, 2015.
[10] G. Kappel, E. Kapsammer, and W. Retschitzegger, "Integrating XML and relational database systems," *World Wide Web*, vol. 7, no. 2, pp. 343–384, 2004.
[11] W. J. Li, X. Chen, and Z. M. Ma, "Reengineering fuzzy nested relational databases into fuzzy XML model," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Jul. 2014, pp. 1612–1617.
[12] T. Lukasiewicz and U. Straccia, "Managing uncertainty and vagueness in description logics for the semantic Web," *Web Semantics, Sci., Serv. Agents World Wide Web*, vol. 6, no. 4, pp. 291–308, 2008.
[13] Z. M. Ma, J. Liu, and L. Yan, "Fuzzy data modeling and algebraic operations in XML," *Int. J. Int. Syst.*, vol. 25, no. 9, pp. 925–947, 2010.
[14] Z. M. Ma, J. Liu, and L. Yan, "Matching twigs in fuzzy XML," *Inf. Sci.*, vol. 181, no. 1, pp. 184–200, 2011.
[15] Z. M. Ma and L. Yan, "Fuzzy XML data modeling with the UML and relational data models," *Data Knowl. Eng.*, vol. 63, no. 3, pp. 972–996, 2007.
[16] Z. M. Ma and L. Yan, "Modeling fuzzy data with XML: A survey," *Fuzzy Sets Syst.*, vol. 301, pp. 146–159, Oct. 2016.
[17] T. Naser, R. Alhajj, and M. Ridley, "Flexible approach for representing object oriented databases in XML format," in *Proc. 10th Int. Conf. Inf. Integrat. Web-Based Appl. Serv.*, 2008, pp. 430–433.
[18] T. Naser, K. Kianmehr, R. Alhajj, and M. J. Ridley, "Transforming object-oriented databases into XML," in *Proc. IEEE Int. Conf. Inf. Reuse Integr.*, Aug. 2007, pp. 600–605.
[19] A. Nierrman and H. V. Jagadish, "ProTDB: Probabilistic data in XML," in *Proc. 28th Int. Conf. Very Large Data Bases*, 2002, pp. 646–657.
[20] B. Oliboni and G. Pozzani, "Representing fuzzy information by using XML schema," in *Proc. Database Expert Syst. Appl.*, 2008, pp. 683–687.
[21] *OWL 2 Web Ontology Language Direct Semantics (Second Edition)*. [Online]. Available: http://www.w3.org/TR/2012/REC-owl2-direct-semantics-20121211/
[22] *OWL 2 Web Ontology Language Document Overview (Second Edition)*. [Online]. Available: https://www.w3.org/TR/owl2-overview/
[23] *OWL 2 Web Ontology Language New Features and Rationale (Second Edition)*. [Online]. Available: http://www.w3.org/TR/2012/REC-owl2-new-features-20121211/
[24] F. F. F. Peres and R. dos S. Mello, "A rule-based conversion of an object-oriented database schema to a schema in XML schema," in *Proc. IEEE 4th Int. Conf. Digit. Inf. Manag.*, Nov. 2009, pp. 1–7.
[25] P. Senellart and S. Abiteboul, "On the complexity of managing probabilistic XML data," in *Proc. 26th ACM SIGMOD-SIGACT-SIGART Symp. Principles Database Syst.*, 2007, pp. 283–292.
[26] G. Stoilos, G. Stamou, and J. Z. Pan, "Fuzzy extensions of OWL: Logical properties and reduction to fuzzy description logics," *Int. J. Approx. Reasoning*, vol. 51, no. 6, pp. 656–679,2010.
[27] M. Van Keulen, A. de Keijzer, and W. Alink, "A probabilistic XML approach to data integration," in *Proc. Int. Conf. Data Eng.*, 2005, pp. 459–470.
[28] C. A. Yaguinuma, M. T. Santos, H. A. Camargo, M. C. Nicoletti, and T. M. Nogueira, "A meta-ontology for modeling fuzzy ontologies and its use in classification tasks based on fuzzy rules," *Int. J. Comput. Inf. Syst. Ind. Manage. Appl.*, vol. 6, pp. 89–101, Jun. 2014.
[29] L. Yan, Z. M. Ma, and J. Liu, "Fuzzy data modeling based on XML schema," in *Proc. ACM Symp. Appl. Comput.*, 2009, pp. 1563–1567.
[30] L. Yan and Z. M. Ma, "Formal translation from fuzzy EER model to fuzzy XML model," *Expert Syst. Appl.*, vol. 41, no. 8, pp. 3615–3627, 2014.
[31] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965.
[32] L. A. Zadeh, "Fuzzy sets as a basis for a theory of possibility," *Fuzzy Sets Syst.*, vol. 1, no. 1, pp. 3–28, 1978.
[33] F. Zhang, L. Yan, Z. M. Ma, and J. Cheng, "Knowledge representation and reasoning of XML with ontology," in *Proc. ACM Symp. Appl. Comput.*, 2011, pp. 1705–1710.
[34] F. Zhang, Z. M. Ma, G. Fan, and X. Wang, "Automatic fuzzy semantic Web ontology learning from fuzzy object-oriented database model," in *Proc. Int. Conf. Database Expert Syst. Appl.*, 2010, pp. 16–30.
[35] F. Zhang, Z. M. Ma, and L. Yan, "Construction of fuzzy ontologies from fuzzy XML models," *Knowl.-Based Syst.*, vol. 42, pp. 20–39, Apr. 2013.
[36] F. Zhang and Z. M. Ma, "Representing and reasoning about XML with ontologies," *Appl. Intell.*, vol. 40, no. 1, pp. 74–106, 2014.

WEIJUN LI received the M.S. degree from Northwest University, Xi'an, China. He is currently pursuing the Ph.D. degree in computer application technology at Northeastern University, Shenyang, China. His current research interests include knowledge engineering and the Semantic Web.

LI YAN received the Ph.D. degree from Northeastern University, Shenyang, China. She is currently a Full Professor with the Nanjing University of Aeronautics and Astronautics, Nanjing, China. She has authored or co-authored over 60 papers on these topics. She is also the author of two monographs published by Springer. Her research interests include databases, XML, and the Semantic Web, with a special focus on spatiotemporal information and uncertainty.



FU ZHANG received the Ph.D. degree from Northeastern University, Shenyang, China, in 2011. He is currently an Associate Professor with the School of Computer Science and Engineering, Northeastern University. He has authored over 40 refereed international journals and conference papers. His research has been published in high-quality international conferences, e.g., CIKM and DEXA, and in highly cited international journals, e.g., *Fuzzy Sets and Systems*, *Knowledge-Based Systems*, and *Integrated Computer-Aided Engineering*. He has also authored two monographs published by Springer. His current research interests include knowledge graph, the Semantic Web, and knowledge representation and reasoning.

XU CHEN received the Ph.D. degree from Northeastern University, Shenyang, China, in 2017. He is currently a Senior Engineer with North Minzu University, Yinchuan, China. His current research interests include knowledge engineering and spatiotemporal data management.