Automatic Fuzzy Semantic Web Ontology Learning from Fuzzy Object-Oriented Database Model

Fu Zhang, Z.M. Ma, Gaofeng Fan, and Xing Wang

College of Information Science & Engineering, Northeastern University, Shenyang, 110819, China
mazongmin@ise.neu.edu.cn

Abstract. How to construct Web ontologies that meet applications' needs has become a key technology to enable the Semantic Web. Manual development of ontologies remains a cumbersome and time-consuming task. In real-world applications, however, information is often vague or ambiguous. Thus, developing approaches and tools for constructing fuzzy ontologies by extracting domain knowledge from huge amounts of existing fuzzy databases can facilitate fuzzy ontology development. In this paper, we propose a formal approach and an automated tool for constructing fuzzy ontologies from fuzzy Object-Oriented database (FOOD) models. Firstly, we introduce the fuzzy ontology, which consists of the fuzzy ontology structure and instances. Then, the FOOD models are investigated, and we propose a kind of formal definition of FOOD models. On this basis, we develop a formal approach that can translate the FOOD model and its corresponding database instances into the fuzzy ontology structure and the fuzzy ontology instances, respectively. Furthermore, following the proposed approach, we implement an automated learning tool, which can automatically construct fuzzy ontologies from FOOD models. Case studies show that the approach is feasible and the automated learning tool is efficient.

Keywords: Fuzzy database; Fuzzy Object-Oriented database (FOOD) model; Fuzzy ontology; Ontology learning; Automated learning tool.

1 Introduction

How to construct Web ontologies that meet applications' needs has become a key technology to enable the Semantic Web [2]. Ontology can be generated from various data resources such as textual data, dictionary, knowledge-based, semi-structured schemata, and database models [15]. Compared to the other types of data resources, ontology construction from database models has increasingly attracted the most attention because most formatted data on the Web are still stored in databases and are not published as an open Web of inter-referring resources [2], [15].

On this basis, constructing Web ontologies by extracting domain knowledge from database models has been extensively investigated. The mappings from relational models to ontologies were established in [12], [23], [29]. The approach for translating ER models into ontologies was developed in [27]. The ontology development tools

P. García Bringas et al. (Eds.): DEXA 2010, Part I, LNCS 6261, pp. 16–30, 2010.

[15] such as Protégé-2000 and OntoEdit can be used to build or reuse ontologies. For a comprehensive review of constructing Web ontologies, ones can refer to [15].

However, information is often vague or ambiguous. Thus, the problems that emerge are how to represent these uncertain data within the ontology definitions and database models. To fill this gap, the fuzziness has been extensively introduced into ontologies [4], [5], [14], [20], [21], [25], [26]. Also, many researches have already been made to represent and manipulate fuzzy data in various database models such as relational database models [11], [14], Object-Oriented database models [7], [8], [9], [18]. For the comprehensive reviews about fuzzy ontologies and fuzzy database models, please refer to [20], [13], respectively.

In recent years, how to construct fuzzy ontologies from fuzzy database models has increasingly received the attention. In [21], Quan proposed a fuzzy ontology framework (FOGA) that can generate a fuzzy ontology from uncertainty data based on Formal Concept Analysis (FCA) theory. Zhang [30] developed an approach for constructing fuzzy ontologies from fuzzy ER models. Ma [14] presented a fuzzy ontology generation framework from fuzzy relational databases. Zhang [29] investigated how to construct fuzzy ontologies from fuzzy UML models. And the automated learning tools were missed in [14], [29], [30]. Blanco [4] realized the translation from fuzzy relational databases to fuzzy ontologies.

In particular, it has been found that the classical database models (such as ER model, relational model, and UML) and their extensions of fuzziness do not satisfy the need of handling complex objects with imprecision and uncertainty [7], [11]. Fuzzy Object-Oriented database (FOOD) models are hereby developed. The FOOD model, which can model uncertain data and complex-valued attributes as well as complex relationships among objects, has been extensively applied in the data and knowledge intensive applications such as CAD/CAM, office automation systems, and so on [7], [8], [9], [11], [13], [18]. Currently, many data sources are modeled in fuzzy Object-Oriented databases, and abundant domain knowledge is contained in FOOD models [9]. Hence, it is possible and meaningful to use the FOOD models as the base knowledge for domain ontology constructing, which will facilitate the development of fuzzy ontology.

Moreover, as mentioned in [23], to support several Semantic Web applications (e.g., Semantic Web sites), the existing Web data and documents must be "upgraded" to Semantic Web content that is semantically annotated with Web ontologies. It is well known that the Web-accessible databases are the main content sources on the current Web [27]. However, a precondition here is that ones have a Web ontology at hand that can capture the domain knowledge of the database. As lots of fuzzy databases today are modeled in FOOD models (see [7], [9], [18], etc.), it is necessary to develop approaches and tools for constructing fuzzy ontologies from FOOD models, which can also act as a gap-bridge between lots of existing database applications and the Semantic Web.

To our best knowledge, there are no reports on fuzzy ontology construction from FOOD models. In this paper, we develop a formal approach and an automated tool for constructing fuzzy ontologies from FOOD models. The paper includes the details:

- How to represent the constructed fuzzy ontology? Since the paper aims at constructing fuzzy ontologies from FOOD models, it is necessary for us to give the formal definition of target fuzzy ontologies. In **Section 3**, we report the fuzzy ontology definition presented in [30].

- How to formalize the FOOD model? In **Section 4**, FOOD models are studied, and we propose a kind of formal definition of FOOD models.
- How to construct fuzzy ontologies from FOOD models? The answer is (see **Section 5**): (**i**) translating the FOOD model into the fuzzy ontology structure at conceptual level; (**ii**) translating the database instances w.r.t. the FOOD model into the fuzzy ontology instances at instance level; (**iii**) proving that the translations are "semantics-preserving" and giving the translation examples.
- How to develop the automated learning tool? Following the proposed approach, in **Section 6**, we implement an automated learning tool called *FOOD2FOWL*, which can automatically construct fuzzy ontologies from FOOD models. Case studies show that the proposed approach is feasible and the tool is efficient.

The remainder of this paper is organized as follows. Section II introduces related work. Section III introduces the fuzzy ontology. Section IV proposes the formal definition of FOOD models. Section V proposes the learning approach. Section VI implements the automated learning tool. Section VII shows conclusions.

2 Related Work

The importance of ontologies to the Semantic Web has prompted the development of kinds of methods and tools to help people construct ontologies. Besides the approaches introduced in Section 1, the following researches are related to our work.

There are several approaches for establishing the relationships between Description Logics and Object-Oriented database models. The approaches presented in [3], [6], [22] investigated how to reason on the Object-Oriented database model by translating it into a Description Logic knowledge base, whereas our work aims at constructing fuzzy ontologies.

Moreover, several works investigated the relationships between Object-Oriented languages and ontologies, but they did not focus on constructing ontologies from Object-Oriented database models. Meditskos [16] used the Object-Oriented language COOL to model and handle ontology concepts and RDF resources. Mota [17] developed an Object-Oriented framework for representing ontologies. Oren [19] presented an Object-Oriented API for managing RDF data. Koide [10] demonstrated the possibility of the integration of OWL and Object-Oriented Programming.

As mentioned in Section 1, there have been several approaches for constructing fuzzy ontologies from fuzzy database models. However, considering that the classical database models (e.g., ER model, relational model, and UML) and their extensions of fuzziness do not satisfy the need of modeling complex objects with imprecision and uncertainty, and many data sources are modeled in FOOD models in real-world applications [7], [8], [9], [11], [13], [18], thus the paper aims at investigating how to construct fuzzy ontologies from FOOD models.

In addition, the fuzzy ontology instances presented in [14], [29] were represented by the fuzzy RDF models, in this paper, as with the representation form of the fuzzy ontology structure, the fuzzy ontology instances were also represented by the form of axioms, which are facilitate access to and evaluation of the fuzzy ontologies [5], [26].

3 Fuzzy OWL DL Ontology

A fuzzy ontology formulated in fuzzy OWL DL language is called fuzzy OWL DL ontology [30]. In this section, we first introduce the fuzzy OWL DL language based on the works [24], [25], [26], [30], and then report the fuzzy OWL DL ontology definition presented in [30].

3.1 Fuzzy OWL DL Language

OWL has three increasingly expressive sublanguages OWL Lite, OWL DL, and OWL Full [26]. OWL DL is the language chosen by the major ontology editors because it supports those users who want the maximum expressiveness without losing computational completeness and decidability of reasoning systems [26]. In addition, OWL DL has two types of syntactic form [27]: the exchange syntax, i.e., the RDF/XML syntax, and the frame-like style abstract syntax. The abstract syntax facilitates access to and evaluation of the ontologies [5], [26], [27]. However, OWL DL cannot handle the information represented with a not precise definition.

The fuzzy OWL DL [5], [26] is the fuzzy extension of OWL DL, which can be approximately viewed as the expressive fuzzy Description Logic f-SHOIN(D) [24]. Based on [26], Table 1 gives the *fuzzy OWL DL abstract syntax*, *Description Logic syntax*, and *semantics*. Here, parts of OWL constructors are omitted in Table 1.

The semantics for fuzzy OWL DL is based on the interpretation of f-SHOIN(D) (details refer to [24]). The interpretation is given by a pair $<\Delta^{FI}, \bullet^{FI}>$ where Δ^{FI} is the individual domain, $\bullet^{FI}$ is a fuzzy interpretation function. In Table 1, Δ_D^{FI} is the data-value domain, $\forall d^{FI}, c^{FI}, o^{FI} \in \Delta^{FI}$, $v^{FI} \in \Delta_D^{FI}$, C denotes class description, D denotes fuzzy data range, fuzzy ObjectProperty and DatatypeProperty identifiers are denoted by R and U, respectively, $\#S$ denotes the cardinality of a set S, $\bowtie \in \{\geq, >, \leq, <\}$.

3.2 Fuzzy OWL DL Ontology

A fuzzy OWL DL ontology is a couple $FO = (FO_S, FO_I)$ where FO_S is the fuzzy ontology structure and FO_I is the fuzzy ontology instances associated with the fuzzy ontology structure. The following Definition 1 gives the formal definition of fuzzy OWL DL ontology [30], and we further add fuzzy individual axioms to the definition.

Definition 1 (Fuzzy OWL DL Ontology). A fuzzy OWL DL ontology is a couple $FO = (FO_S, FO_I) = (FID_0, FAxiom_0)$, where:

(1) $FID_0 = FCID_0 \cup FIID_0 \cup FDRID_0 \cup FOPID_0 \cup FDPID_0$ is a fuzzy OWL DL identifier set (see Table 1) partitioned into:

- a subset $FCID_0$ of fuzzy class identifiers, include user-defined identifiers plus two predefined fuzzy classes owl: Thing and owl: Nothing.
- a subset $FIID_0$ of individual identifiers.
- a subset $FDRID_0$ of fuzzy data range identifiers; each fuzzy data range identifier is a predefined XML Schema fuzzy datatype [13], [25].

- a subset $FOPID_0$ of fuzzy object property identifiers.
- a subset $FDPID_0$ of fuzzy datatype property identifiers.

(2) $FAxiom_0$ is a fuzzy OWL DL axiom set (see Table 1) partitioned into:

- a subset of fuzzy class/property axioms.
- a subset of fuzzy individual axioms.

From a semantics point of view, a fuzzy OWL DL ontology *FO* is a set of fuzzy OWL DL axioms in Table 1. An interpretation *FI* is a model of *FO* iff it satisfies all axioms in *FO*. Furthermore, an *FO* is satisfiable iff there is a model of it.

Table 1. Fuzzy OWL DL Abstract Syntax, Description Logic (DL) Syntax, and Semantics

Fuzzy OWL DL Syntax	DL Syntax	Model-Theoretic Semantics
Fuzzy class description (C)		
A is a *URIref* of a fuzzy class owl:Thing owl:Nothing unionOf ($C_1 \ldots C_n$)	A T $\perp$ $C_1 \cup \ldots \cup C_n$	$A^{FI}: \Delta^{FI} \rightarrow [0, 1]$ $T^{FI}(d) = 1$ $\perp^{FI}(d) = 0$ $(C_1 \cup \ldots \cup C_n)^{FI}(d) = \max\{C_1^{FI}(d), \ldots, C_n^{FI}(d)\}$
restriction (R allValuesFrom(C))	$\forall$R.C	$(\forall R.C)^{FI}(d) = inf_{d' \in \Delta^{FI}} \{\max\{1 - R^{FI}(d,d'), C^{FI}(d')\}\}$
restriction (R minCardinality(n))	$\geq$ n R	$(\geq nR)^{FI}(d) = sup_{c_1,\ldots,c_n \in \Delta^{FI}} \wedge_{i=1}^{n} R^{FI}(d,c_i)$
restriction (R maxCardinality(n))	$\leq$ n R	$(\leq nR)^{FI}(d) = inf_{c_1,\ldots,c_{n+1} \in \Delta^{FI}} \vee_{i=1}^{n+1} (1 - R^{FI}(d,c_i))$
restriction (R cardinality(n))	= n R	$(= nR)^{FI}(d) = (\geq nR \cap \leq nR)^{FI}(d)$
restriction (U allValuesFrom(D))	$\forall$U.D	$(\forall U.D)^{FI}(d) = inf_{v \in \Delta_D^{FI}} \{\max\{1 - U^{FI}(d,v), D^{FI}(v)\}\}$
Fuzzy class axioms		
Class (A partial $C_1 \ldots C_n$) SubClassOf (C_1 C_2) EquivalentClasses ($C_1 \ldots C_n$) DisjointClasses ($C_1 \ldots C_n$)	$A \subseteq C_1 \cap .. \cap C_n$ $C_1 \subseteq C_2$ $C_1 \equiv \ldots \equiv C_n$ $C_i \neq C_j$	$A^{FI}(d) \leq \min\{C_1^{FI}(d), \ldots, C_n^{FI}(d)\}$ $C_1^{FI}(d) \leq C_2^{FI}(d)$ $C_1^{FI}(d) = \ldots = C_n^{FI}(d)$ $\min\{C_i^{FI}(d), C_j^{FI}(d)\} = 0 \quad 1 \leq i < j \leq n$
Fuzzy property axioms		
DatatypeProperty (U domain(C_1)…domain(C_m) range(D_1)…range(D_k) [Functional]) ObjectProperty (R domain(C_1)…domain(C_m) range(C_1)… range(C_k) [inverseOf (R_0)])	 $\geq 1U \subseteq C_i$ $T \subseteq \forall U.D_i$ $T \subseteq \leq 1U$ $\geq 1R \subseteq C_i$ $T \subseteq \forall R.C_i$ $R = (R_0)^-$	 $U^{FI}(d, v) \leq C_i^{FI}(d) \quad i = 1,\ldots, m$ $U^{FI}(d, v) \leq D_i^{FI}(v) \quad i = 1,\ldots, k$ $\forall d \in \Delta^{FI} \, \#\{v \in \Delta_D^{FI} : U^{FI}(d, v) \geq 0\} \leq 1$ $R^{FI}(d_1, d_2) \leq C_i^{FI}(d_1) \quad i = 1,\ldots, m$ $R^{FI}(d_1, d_2) \leq C_i^{FI}(d_2) \quad i = 1,\ldots, k$ $R^{FI}(d_1, d_2) = R_0^{FI}(d_2, d_1)$
Fuzzy individual axioms		
Individual (*o* type(C_1) [$\bowtie m_1$]… value(R_1, o_1) [$\bowtie k_1$]… value(U_1, v_1) [$\bowtie l_1$]…) SameIndividual ($o_1 \ldots o_n$) DifferentIndividuals ($o_1 \ldots o_n$)	$o : C_i \bowtie m_i$ $(o,o_i):R_i \bowtie k_i$ $(o,v_i):U_i \bowtie l_i$ $o_1 = \ldots = o_n$ $o_i \neq o_j$	$C_i^{FI}(o) \bowtie m_i, \quad m_i \in [0, 1], \quad 1 \leq i \leq n$ $R_i^{FI}(o, o_i) \bowtie k_i, \quad k_i \in [0, 1], \quad 1 \leq i \leq n$ $U_i^{FI}(o, v_i) \bowtie l_i, \quad l_i \in [0, 1], \quad 1 \leq i \leq n$ $o_1^{FI} = \ldots = o_n^{FI}$ $o_i^{FI} \neq o_j^{FI} \quad 1 \leq i < j \leq n$

4 The Fuzzy Object-Oriented Database (FOOD) Model

The fuzzy Object-Oriented database (FOOD) model is the fuzzy extension of traditional Object-Oriented database model (OODM) [3], [6], i.e., some major notions in OODMs such as objects, classes, object-class relationships, and inheritance relationship are extended under fuzzy information environment. In addition, similarly for the references [1], [3], [6], [8], [9], [13], we restrict our attention to the structural components of FOOD models.

The FOOD models are based on the notions of fuzzy objects, fuzzy classes, fuzzy attributes, and fuzzy inheritances [9]. In the following, we propose a kind of formal definition of FOOD models, which is the fuzzy extension of OODM definition in [6].

Definition 2 (FOOD Model). A FOOD model is a finite set of class declarations, which is a tuple $FS = (FC_{FS}, FA_{FS}, FD_{FS})$, where:

- FC_{FS} is a finite set of fuzzy class names, denoted by the letter FC;
- FA_{FS} is a finite set of fuzzy attribute names, denoted by the letter FA;
- FD_{FS} is a finite set of fuzzy class declarations. For each fuzzy class FC $\in$ FC_{FS}, FD_{FS} contains exactly one such declaration:

  Class FC *is-a* $FC_1, \ldots, FC_n$ *type-is* FT,

  where *is-a* denotes the inheritance relationship between classes, *type-is* specifies the structure of class FC through a *type expressive* FT, which is built according to the following syntax:

  FT → FC |
  Union $FT_1, \ldots, FT_k$ *End* |
  Set-of FT |
  Record $FA_1{:}FT_1, \ldots, FA_k{:}FT_k$ *End*.

Fig. 1 shows a simple *FOOD model* FS_1 modeling part of the reality at a company.

```
Class Computer type-is
    Union HP-Computer, Dell-Computer
    End
Class Old-Computer is-a Computer type-is
    Record
        ComID: String
        Brand: String
        UseYear: Integer
    End
Class Young-Employee is-a Employee type-is
    Record
        EmpID: String
        Name: String
        FUZZY Age: Integer
        FUZZY Use: Set-of   Old-Computer
    End
```

Fig. 1. A FOOD model FS1 modeling part of the reality at a company

The detailed instructions about Fig. 1 are as follows:

(1) *Computer* is a generalization of *HP-Computer* and *Dell-Computer*.

(2) Inverse of attributes: in order to distinguish the *subject* and the *object* of attribute *Use*, the roles *Useby* and *Useof* are introduced (see Fig. 2). Therefore, the inverse of attribute *Use* is introduced, which will be discussed in Section 5.1 in detail.

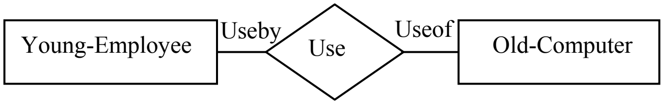

Fig. 2. The graphical representation of the inverse of attribute Use

(3) The attributes FA_i (such as *Name*, *Age* in Fig. 1), called *datatype attributes*, which associate with the basic domains such as Integer, Real, etc.

(4) The attributes *FA* (such as *Use* in Fig. 1), called *object attributes*, which denote the relationship between two participant classes.

The semantics of a FOOD model can be given by the fuzzy database states *FJ*, i.e., the fuzzy database instances. Fig. 3 shows the *database instances* w.r.t. the FOOD model FS_1 (only part of data):

(1) u = 0.9 denotes that the possibility which an object O_1 belongs to the class *Young-Employee* is 0.9.

(2) The value of attribute *Use* is a possibility distribution {0.8/O_2, 0.7/O_3}, where 0.8/O_2 denotes that the possibility which object O_1 uses object O_2 is 0.8.

(3) The values of the other fuzzy attributes (e.g., *Age*) are represented by the possibility distributions [31], [32].

Young-Employee

EmpID	*Name*	*Fuzzy Age*	*Fuzzy Use*	*u*
O_1	Chris	{ 0.8/27, 0.9/30 }	{ 0.8/O_2, 0.7/O_3 }	0.9

Old-Computer

ComID	*Brand*	*UseYear*	*u*
O_2	Dell	4	0.85

Old-Computer

ComID	*Brand*	*UseYear*	*u*
O_3	Hp	3	0.8

Fig. 3. The database instances w.r.t. the FOOD model FS1 in Fig. 1

5 Fuzzy OWL DL Ontology Learning from FOOD Model

This section proposes a formal approach for constructing fuzzy OWL DL ontologies from FOOD models, including: **(i)** translating the FOOD model into the fuzzy ontology structure (see Definition 3). **(ii)** translating the database instances w.r.t. the FOOD model into the fuzzy ontology instances (see Definition 4). **(iii)** proving that the translations are "semantics-preserving" and giving the translation examples.

5.1 Translating FOOD Model into Fuzzy OWL DL Ontology Structure

The Definition 3 gives a formal approach for translating the FOOD model into the fuzzy OWL DL ontology structure. Starting with the construction of the *atomic identifiers* FID_0, the approach induces a set of *fuzzy class/property axioms* $FAxiom_0$ from the FOOD model.

Definition 3 (Structure Translation). Given a FOOD model $FS = (FC_{FS}, FA_{FS}, FD_{FS})$ in Definition 2. The fuzzy OWL DL ontology structure $FO_S = \varphi(FS) = (FID_0, FAxiom_0)$ can be derived by function φ as shown in Table 2.

Table 2. Translating rules from a FOOD model to a fuzzy OWL DL ontology structure

FOOD model $FS = (FC_{FS}, FA_{FS}, FD_{FS})$	**Fuzzy ontology structure $FO_S = \varphi(FS) = (FID_0, FAxiom_0)$**
fuzzy class FC_{FS}, fuzzy attribute FA_{FS}	**identifier set FID_0**
Each fuzzy class symbol FC	A fuzzy class identifier $\varphi(FC) \in FCID_0$
Each type expression *Record* FA : *Set-of* FT *End*	
Each *object attribute* symbol FA	A fuzzy class identifier $\varphi(FA) \in FCID_0$
Each *object attribute* symbol FA	Add two additional roles FU_1 and FU_2 ; FU_1 and FU_2 are mapped into two fuzzy object property identifiers $\varphi(FU_1)$ and $\varphi(FU_2)$: $\varphi(FU_1) \in FOPID_0$ $\varphi(FU_2) \in FOPID_0$; In addition, we use symbols FV_1 and FV_2 denote the inverse properties of $\varphi(FU_1)$ and $\varphi(FU_2)$, respectively: FV_1 = invof_$\varphi(FU_1) \in FOPID_0$ FV_2 = invof_$\varphi(FU_2) \in FOPID_0$
Each type expression *Record* $FA_1:FD_1, \ldots, FA_k:FD_k$ *End*	
Each *datatype attribute* symbol FA_i	A fuzzy datatype property identifier $\varphi(FA_i) \in FDPID_0$
Each domain symbol FD_i	A fuzzy XML Schema datatype identifier $\varphi(FD_i) \in FDRID_0$
fuzzy class declarations FD_{FS}	**fuzzy class/property axiom set $FAxiom_0$**
Each fuzzy class declaration: Class FC *is-a* $FC_1, \ldots, FC_n$	Create a fuzzy class axiom: Class ($\varphi(FC)$ partial $\varphi(FC_1), \ldots, \varphi(FC_n)$).

Tabel 2. (*continued*)

Each fuzzy class declaration: Class FC *type-is* *Union* FT_1 ,…, FT_k *End*	Create fuzzy class axioms: Class (φ(FC) partial unionOf(φ(FT_1),…, φ(FT_k))); SubClassOf (φ(FT_i) φ(FC)).
Each fuzzy class declaration: Class FC_1 *type-is* *Record* FA: *Set-of* FC_2 *End*	Create a fuzzy class axiom: Class (φ(FA) partial restriction (φ(FU_1) allValuesFrom (φ(FC_1)) cardinality(1)) restriction (φ(FU_2) allValuesFrom (φ(FC_2)) cardinality(1))); FOR i = 1, 2 DO: The fuzzy property axioms: ObjectProperty (φ(FU_i) domain (φ(FA)) range (φ(FC_i))); ObjectProperty (FV_i domain (φ(FC_i)) range (φ(FA)) inverseOf φ(FU_i)); The fuzzy class axioms: Class (φ(FC_i) partial restriction (FV_i allValuesFrom (φ(FA)))).
Each fuzzy class declaration: Class FC *type-is* *Record* FA_1:FD_1,…, FA_k:FD_k *End*	Create fuzzy class/property axioms: Class (φ(FC) partial restriction (φ(FA_1) allValuesFrom (φ(FD_1)) cardinality(1)) … restriction (φ(FA_k) allValuesFrom (φ(FD_k)) cardinality(1))) ; DatatypeProperty (φ(FA_i) domain (φ(FC)) range (φ(FD_i)) [Functional]) where i = 1, 2,…,k .

Below we discuss the effectiveness of the translation, which can be sanctioned by the following Theorem 1.

Theorem 1. For every FOOD model *FS* = (FC_{FS}, FA_{FS}, FD_{FS}), *FJ* is a fuzzy database state w.r.t. *FS*, and φ(*FS*) is the fuzzy OWL DL ontology structure derived from *FS* by Definition 3. There exist mappings:

- α_{FS} from fuzzy database state *FJ* to model of φ(*FS*) such that: For each legal fuzzy database state *FJ* for *FS*, there is α_{FS}(*FJ*) which is a model of φ(*FS*);
- β_{FS} from model of φ(*FS*) to fuzzy database state such that: For each model *FI* of φ(*FS*), there is β_{FS}(*FI*) which is a legal fuzzy database state for *FS*.

Proof: The following briefly gives the proof of the first part of Theorem 1. Given a fuzzy database state *FJ*, and $\bullet^{FJ}$ is a fuzzy interpretation function that can map each element in Fig. 1 to its corresponding value in Fig. 3, we can define a fuzzy interpretation α_{FS}(*FJ*) of φ(*FS*) as follows:

- The domain elements $\Delta^{\alpha_{FS}(FJ)}$ of interpretation $\alpha_{FS}(FJ)$ of $\varphi(FS)$ are constituted by values of *FJ* (e.g., Fig. 3). Since each object of *FS* is assigned a structured value, in order to explicitly represent in fuzzy ontology the type structure of the object, we denote with Δ_{fid} , Δ_{frec} , Δ_{fset} the domain elements of $\Delta^{\alpha_{FS}(FJ)}$ corresponding to object identifiers, fuzzy record values, and fuzzy set values in FV_{FJ}, respectively.
- The fuzzy OWL DL identifier set FID_0 of $\varphi(FS)$ in Definition 3 are defined:
  $(\varphi(X))^{\alpha_{FS}(FJ)} = X^{FJ}$, where $X \in$ FT;
  $(\varphi(FD_i))^{\alpha_{FS}(FJ)} = FD_i^{FJ}$;
  For each class declaration *Class* FC *type-is Record* $FA_1:FD_1,\ldots, FA_k:FD_k$ *End*, we have $(\varphi(FA_i))^{\alpha_{FS}(FJ)} = \{ <c, d_i> \in \Delta^{\alpha_{FS}(FJ)} \times \Delta^{\alpha_{FS}(FJ)} \mid c \in FC^{FJ} \wedge d_i \in FD_i^{FJ} \}$, where i = 1, 2,..., k;
  For class declaration *Class* FC_1 *type-is Record* FA: *Set-of* FC_2 *End*, we have $(\varphi(FU_j))^{\alpha_{FS}(FJ)} = \{<r, c_j> \in \Delta^{\alpha_{FS}(FJ)} \times \Delta^{\alpha_{FS}(FJ)} \mid r \in FA^{FJ} \wedge c_j \in FC_j^{FJ} \}$, j = 1, 2.

$FO_S = (FID_0, FAxiom_0)$. For reasons of space, FID_0 and parts of axioms are omitted.
$FAxiom_0$ = {
SubClassOf (Old-Computer Computer) ;
SubClassOf (Young-Employee Employee) ;
Class (Computer partial unionOf (Hp-Computer, Dell-Computer)) ;
SubClassOf (Hp-Computer Computer) ;
SubClassOf (Dell-Computer Computer) ;
Class (Old-Computer partial restriction (ComID allValuesFrom (xsd:String)
cardinality (1)) restriction (Brand allValuesFrom (xsd:String)
cardinality (1)) restriction (UseYear allValuesFrom (xsd:Integer)
cardinality (1)));
Class (Young-Employee partial restriction (EmpID allValuesFrom (xsd:String)
cardinality (1)) restriction (Name allValuesFrom (xsd:String) cardinality (1))
restriction (Age allValuesFrom (xsd:fuzzy Integer) cardinality (1)));
Class (Use partial restriction (Useby allValuesFrom (Young-Employee) cardinality(1))
restriction (Useof allValuesFrom (Old-Computer) cardinality(1))) ;
Class (Young-Employee partial restriction (invof_Useby allValuesFrom (Use))) ;
Class (Old-Computer partial restriction (invof_Useof allValuesFrom (Use))) ;
ObjectProperty (invof_Useby domain (Young-Employee) range (Use) inverseOf
Useby) ;
ObjectProperty (invof_Useof domain (Old-Computer) range (Use) inverseOf Useof) ;
ObjectProperty (Useby domain (Use) range (Young-Employee)) ;
ObjectProperty (Useof domain (Use) range (Old-Computer)) ;
DatatypeProperty (EmpID domain (Young-Employee) range (xsd:String) [Functional]) ;
DatatypeProperty (Name domain (Young-Employee) range (xsd:String) [Functional]) ;
DatatypeProperty (Age domain (Young-Employee) range (xsd:Integer) [Functional]) ;
DatatypeProperty (ComID domain (Old-Computer) range (xsd:Integer) [Functional]);
DatatypeProperty (Brand domain (Old-Computer) range (xsd:String) [Functional]) ;
… }

Fig. 4. The fuzzy OWL DL ontology structure φ(FS1) derived from the FOOD model FS1

Here, we omit the complete proof of Theorem 1 for reasons of space, and the proof of the second part can be done similarly, they are a mutually inverse process.

Example 1. Given a *FOOD model* FS_1 in Fig. 1, by Definition 3, we can obtain the corresponding *fuzzy OWL DL ontology structure* $FO_S = \varphi(FS_1)$ in Fig. 4.

5.2 Mapping Database Instances to Fuzzy OWL DL Ontology Instances

In the previous section, we realize the translation from the FOOD model to the fuzzy OWL DL ontology structure. In order to build fuzzy ontologies of high completeness, in the following, we propose an approach for mapping *the database instances* w.r.t. a FOOD model (e.g., Fig. 3) to *the fuzzy OWL DL ontology instances* (i.e., *a set of fuzzy individual axioms* in Table 1).

Firstly, let us briefly sketch the assertional formalisms used in the fuzzy Object-Oriented databases and fuzzy OWL DL ontologies.

The assertional formalisms of fuzzy OWL DL ontologies are represented by the fuzzy individual axioms, which contain the axioms (see Table 1): Individual (o type(C_1) [$\bowtie$ m_1] … value(R_1, o_1) [$\bowtie$ k_1] … value(U_1, v_1) [$\bowtie$ l_1]…); DifferentIndividuals ($o_1 \ldots o_n$); SameIndividual ($o_1 \ldots o_n$).

From Section 4, a fuzzy Object-Oriented database, which describes the real world by means of objects, values, and their mutual relationships, can be considered as a finite set of assertions [1], [9], [18]. Based on [1], the assertional formalisms of the database instances with respect to a FOOD model specify that: A fuzzy object *fo* is an instance of fuzzy class *FC* with membership degree of n by means of fuzzy assertion *fo*: *FC*: n, where $n \in [0, 1]$; The fuzzy structured value associated with *fo* by means of fuzzy assertion *fo* : [FA_1:FV_1:n_1,…,FA_k:FV_k:n_k], where FV_i is the value of attribute FA_i, $n_i \in [0, 1]$ denotes the membership degree, FA_i may be a fuzzy datatype attribute or a fuzzy object attribute, and $i \in \{1…k\}$. Here, since the value of an attribute FA_i may be a possibility distribution, for simplicity, $FV_i : n_i$ only denotes one element of the possibility distribution; In addition, since the fuzzy subclass/superclass relationships in FOOD models can be assessed by utilizing the inclusion membership degree of objects to the class [8], [13], the assertional formalism of fuzzy subclass/superclass relationships can be re-expressed by the above assertional formalism of objects to the class.

Based on the discussion above, the mappings from the *database instances* w.r.t. a FOOD model to the *fuzzy OWL DL ontology instances* can be established by Definition 4.

Definition 4 (Instance Translation). Given the database instances with respect to a FOOD model (i.e., a finite set of assertions), the corresponding fuzzy OWL DL ontology instances (i.e., a set of fuzzy individual axioms) can be derived as the mapping rules shown in Table 3.

In Table 3, notice that:

(i) In order to represent that the membership degree of an object to a class is equal to n (i.e., *fo*: *FC*: n), the symbol $\bowtie'$ denotes ≥ and ≤.

For example, the axiom *Individual* (*fo* type(*FC*) [$\bowtie'$ n]) denotes two axioms *Individual* (*fo* type(*FC*) [≥n]) and *Individual* (*fo* type(*FC*) [≤n]).

(ii) When the membership degree n = 1.0, the [$\bowtie'$ n] part is omitted in a fuzzy individual axiom.

Table 3. Mapping rules from database instances to fuzzy OWL DL ontology instances

Database instances	Fuzzy OWL DL ontology instances FO_I	
Each fuzzy object symbol *fo*	A fuzzy individual identifier φ(*fo*) ∈ $FIID_0$	
Each fuzzy class symbol *FC*	A fuzzy class identifier φ(*FC*) ∈ $FCID_0$	See Table 2
Each fuzzy *datatype attribute* FA_i	A fuzzy datatype property identifier φ(FA_i) ∈ $FDPID_0$, denoted by U_i	
Each fuzzy *object attribute FA*	A fuzzy class identifier φ(*FA*) ∈ $FCID_0$; In addition, adding the additional fuzzy object property identifiers, denoted by R_i	
The fuzzy assertion *fo*: *FC*: *n*	The fuzzy individual axiom: Individual (φ(*fo*) type(φ(*FC*)) [⋈' *n*])	See Fig. 5
The fuzzy assertion: *fo*:[FA_1:FV_1:n_1,…,FA_k:FV_k:n_k]	The fuzzy individual axiom: Individual (φ(*fo*) value(R_i, o_i) [⋈' n_i]… value(U_i, v_i) [⋈' n_i]…) where o_i, v_i ∈ *FJ*, i ∈ {1…k}	

Example 2. Given the *database instances* in Fig. 3 (w.r.t. the FOOD model FS_1 in Fig. 1), by Definition 4, we can obtain the corresponding *fuzzy OWL DL ontology instances* FO_I in Fig. 5 (w.r.t. the fuzzy OWL DL ontology structure FO_S in Fig. 4).

Given the database instances in Fig. 3, the fuzzy OWL DL ontology instances FO_I can be derived as follows:
Notice that, since each object attribute *FA* is mapped to a fuzzy class identifier φ(*FA*) ∈ $FCID_0$, we need to add an additional object symbol *o′* such that *o′* belongs to class φ(*FA*). $FIID_0$ = { o_1, o_2, o_3, *o′* } ;
$FAxiom_0$ = {
DifferentIndividuals (o_1, o_2, o_3, *o′*) ;
Individual (o_1 type(Young-Employee) [⋈' 0.9]) ;
Individual (o_2 type(Old-Computer) [⋈' 0.85]) ;
Individual (o_3 type(Old-Computer) [⋈' 0.8]) ;
Individual (*o′* type(Use)) ;
Individual (*o′* value(Useby, o_1) value(Useof, o_2) [⋈' 0.8]
value(Useof, o_3) [⋈' 0.7]) ;
Individual (o_1 value(EmpID, o_1) value(Name, Chris) value(Age, 27) [⋈' 0.8]
value(Age, 30) [⋈' 0.9] value(invof_Useby, *o′*)) ;
Individual (o_2 value(ComID, o_2) value(Brand, Dell) value(UseYear, 4)
value(invof_Useof, *o′*) [⋈' 0.8]) ;
Individual (o_3 value(ComID, o_3) value(Brand, Hp) value(UseYear, 3)
value(invof_Useof, *o′*) [⋈' 0.7]) .
}

Fig. 5. The fuzzy OWL DL ontology instances derived from database instances in Fig. 3

6 Automated Learning Tool and Case Study

Based on the proposed approach in Section 5, we developed an automated learning tool called *FOOD2FOWL*, which can automatically construct fuzzy OWL DL ontologies from FOOD models and the corresponding database instances. For reasons of space, the following briefly introduces the implementation of *FOOD2FOWL*.

6.1 Design and Implementation of *FOOD2FOWL*

The tool *FOOD2FOWL* includes three main modules: *parsing module* that parses the FOOD model file, *translation module* that translates the parsed file into the fuzzy OWL DL ontology, and *output module* that produces the constructed fuzzy OWL DL ontology as a text file.

The implementation of *FOOD2FOWL* is based on Java 2 JDK 1.5 platform. The parsing module uses the regular expression to parse the FOOD model file and stores the parsed results as Java ArrayList classes; translation module uses Java class methods to translate the FOOD model and database instances into the fuzzy OWL DL ontology structure and instances based on the proposed approach in Section 5; output module produces the resulting fuzzy OWL DL ontology which is saved as a text file and displayed on the tool screen (see Fig. 6).

6.2 Case Study

We have carried out lots of case studies using *FOOD2FOWL*. For saving space, here we give an example which can show that the proposed approach is feasible and the implemented tool is efficient. Fig. 6 shows the screen snapshot of *FOOD2FOWL*, which displays the translations from the FOOD model (Fig. 1) and the corresponding database instances (Fig. 3) to the fuzzy OWL DL ontology structure (Fig. 4) and the fuzzy OWL DL ontology instances (Fig. 5).

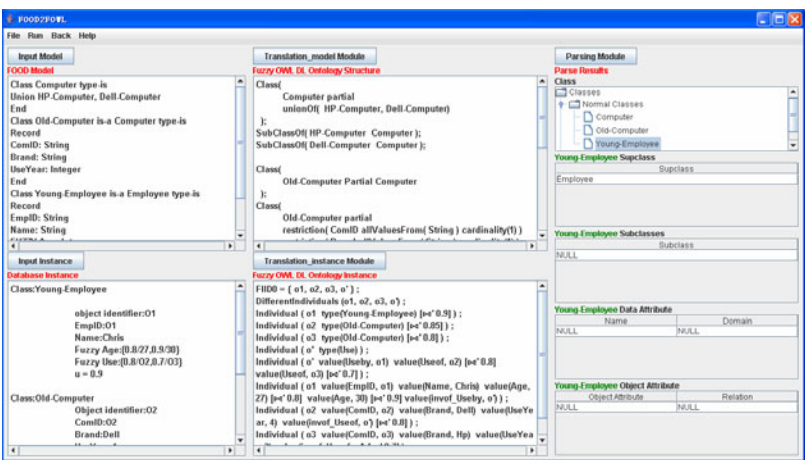

Fig. 6. Screen snapshot of *FOOD2FOWL*

7 Conclusions and Future Work

The real power of the Semantic Web will be realized only when people create much machine-readable content, and ontologies play a key role in this effort. In this paper, we developed a formal approach and an automated tool for constructing fuzzy Semantic Web ontologies from fuzzy Object-Oriented database (FOOD) models. The fuzzy ontology was introduced. A kind of formal definition of FOOD models were proposed. Furthermore, we realized the translations from the FOOD model and its corresponding database instances to the fuzzy ontology structure and fuzzy ontology instances. Following the proposed approach, an automated learning tool called *FOOD2FOWL* was implemented, which can automatically construct fuzzy ontologies from FOOD models. All of these will facilitate the development of Web ontologies and the realization of semantic interoperations between lots of existing database applications and the Semantic Web.

In the future, we aim at developing the other approaches of constructing (fuzzy) ontology. Moreover, it should be noted that many databases have not been designed following the disciplined methodologies, thus we will experiment with other FOOD models to enrich our construction approach.

Acknowledgments. We thank the anonymous reviewers for their valuable suggestions. The work is supported by the *Fundamental Research Funds for the Central Universities* (N090604012 and N090504005) and the *National Natural Science Foundation of China* (60873010), and in part by the *Program for New Century Excellent Talents in University* (NCET-05-0288).

References

1. Artale, A., Cesarini, F., Soda, G.: Describing database objects in a concept language environment. IEEE Trans. Knowledge and Data Engineering 8(2), pp. 345–351 (1996)
2. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web. Scientific American 284(5), 34–43 (2001)
3. Baader, F., McGuinness, D., Nardi, D. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, Cambridge (2003)
4. Blanco, I.J., Vila, M.A., Martinez-Cruz, C.: The use of ontologies for representing database schemas of fuzzy information. J. Intelligent Syst. 23(4), 419–445 (2008)
5. Calegari, S., Ciucci, D.: Fuzzy ontology, fuzzy description logics and fuzzy-owl. In: Masulli, F., Mitra, S., Pasi, G. (eds.) WILF 2007. LNCS (LNAI), vol. 4578, pp. 118–126. Springer, Heidelberg (2007)
6. Calvanese, D., Lenzerini, M., Nardi, D.: Unifying class-based representation formalisms. J. Artificial Intelligence Research 11(2), 199–240 (1999)
7. Cross, V., Caluwe, R., Vangyseghem, N.: A perspective from the fuzzy object data management group (FODMG). In: Proc. Fuzzy Systems, vol. 2, pp. 721–728 (1997)
8. George, R., Srikanth, R., Petry, F.E., Buckles, B.P.: Uncertainty management issues in the object-oriented data model. IEEE Trans. Fuzzy Systems 4(2), 179–192 (1996)
9. Koyuncu, K., Yazici, A.: IFOOD: an intelligent fuzzy object-oriented database architecture. IEEE Trans. on Knowledge and Data Engineering (TKDE) 15(5), 1137–1154 (2003)
10. Koide, S., Aasman, J., Haflich, S.: OWL vs. object oriented programming. In: Workshop on Semantic Web Enabled Software Engineering (SWESE) at ISWC 2005 (2005)
11. Ling, T.C., Yaacob, M.H., Phang, K.K.: Fuzzy database framework-Relational versus Object-oriented model. In: Proc. of Intelligent Information Systems, pp. 246–250 (1997)

12. Lubyte, L., Tessaris, S.: Automatic extraction of ontologies wrapping relational data sources. In: Bhowmick, S.S., Küng, J., Wagner, R. (eds.) DEXA 2009. LNCS, vol. 5690, pp. 128–142. Springer, Heidelberg (2009)
13. Ma, Z.M., Zhang, W.J., Ma, W.Y.: Extending object-oriented databases for fuzzy information modeling. Information Systems 29(5), 421–435 (2004)
14. Ma, Z.M., Yanhui, L.Y., Yan, L.: A Fuzzy Ontology Generation Framework from Fuzzy Relational Databases. Int. J. Semantic Web Information Systems 4(3), 1–15 (2008)
15. Maedche, A., Staab, S.: Ontology Learning for the Semantic Web. IEEE Intelligent Systems 16(2), 72–79 (2001)
16. Meditskos, G., Bassiliades, N.: A Rule-Based Object-Oriented OWL Reasoner. IEEE Trans. Knowledge & Data Engineering 20(3), 397–410 (2008)
17. Mota, L., Botelho, L.M., Mendes, H., Lopes, A.: O3F: an object oriented ontology framework. In: Proc. of the Second International Joint Conference on AAMAS'03 (2003)
18. Ma, Z.M.: Advances in Fuzzy Object-Oriented Databases: Modeling and Applications. Idea Group Publishing (2004)
19. Oren, E., Heitmann, B., Decker, S.: ActiveRDF: Embedding Semantic Web data into object-oriented languages. Journal of Web Semantics 6(3), 191–202 (2008)
20. Ortega, F.B.: Managing Vagueness in Ontologies. E.T.S. de Ingenierías Informática y de. Telecomunicación. Departamento Ciencias de la Computación e I. A. (2008)
21. Quan, T.T., Hui, S.C., Fong, A.C.M.: Automatic fuzzy ontology generation for Semantic Web. IEEE Trans. on Knowledge & Data Engineering 18(6), 842–856 (2006)
22. Roger, M., Simonet, A., Simonet, M.: Bringing together description logics and database in an object oriented model. In: Hameurlain, A., Cicchetti, R., Traunmüller, R. (eds.) DEXA 2002. LNCS, vol. 2453, pp. 504–513. Springer, Heidelberg (2002)
23. Stojanovic, L., Stojanovic, N., Volz, R.: Migrating data-intensive web sites into the semantic Web. In: Proc. 17th ACM Symp. on Applied Computing, pp. 1100–1107 (2002)
24. Straccia, U.: Towards a fuzzy description logic for the semantic Web. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 167–181. Springer, Heidelberg (2005)
25. Sanchez, E., Yamanoi, T.: Fuzzy ontologies for the semantic web. In: Larsen, H.L., Pasi, G., Ortiz-Arroyo, D., Andreasen, T., Christiansen, H. (eds.) FQAS 2006. LNCS (LNAI), vol. 4027, pp. 691–699. Springer, Heidelberg (2006)
26. Stoilos, G., Stamou, G., Tzouvaras, V., Pan, J.Z., Horrocks, I.: Fuzzy OWL: Uncertainty and the Semantic Web. In: International Workshop of OWL: Experiences and Directions (OWL-ED 2005), Galway, Ireland (2005)
27. Xu, Z., Cao, X., Dong, Y.: Formal approach and automated tool for translating ER schemata into OWL ontologies. In: Dai, H., Srikant, R., Zhang, C. (eds.) PAKDD 2004. LNCS (LNAI), vol. 3056, pp. 464–475. Springer, Heidelberg (2004)
28. Xu, Z., Zhang, S., Dong, Y.: Mapping between Relational Database Schema and OWL Ontology for Deep Annotation. In: Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, December 18-22, pp. 548–552 (2006)
29. Zhang, F., Ma, Z.M., Cheng, J., Meng, X.: Fuzzy Semantic Web Ontology Learning from Fuzzy UML Model. In: Proc. of the 18th ACM Conference on Information and Knowledge Management (CIKM 2009), pp. 1007–1016 (2009)
30. Zhang, F., Ma, Z.M., Lv, Y., et al.: Formal Semantics-Preserving Translation from Fuzzy ER Model to Fuzzy OWL DL Ontology. In: Proc. of 2008 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2008), pp. 503–509 (2008)
31. Zadeh, L.A.: Fuzzy sets. Information and Control 8(3), 338–353 (1965)
32. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets Systems 1(1), 3–28 (1978)