IJICTR
International Journal of Information & Communication Technology Research

Volume 11- Number 1 – Winter 2019 (45 -56)

An RDF Based Fuzzy Ontology Using Neural Tensor Networks

Farhad Abedini
Faculty of Computer and Information Technology Engineering,
Qazvin Branch, Islamic Azad University,
Qazvin, Iran
abedini@ariau.ac.ir

Mohammad Reza Keyvanpour*
Department of Computer Engineering,
Alzahra University,
Vanak, Tehran, Iran
keyvanpour@alzahra.ac.ir

Mohammad Bagher Menhaj
Electrical Engineering Department,
Amirkabir University of Technology,
Hafez Ave., Tehran, Iran
menhaj@aut.ac.ir

Received: 5 May 2018 - Accepted: 3 August 2018

Abstract— **As an extension of classical ontology, a fuzzy ontology by employing fuzzy set theory can easily and yet better deal with uncertainties especially for the cases in which knowledge is vague. Obviously, fuzzification plays an important role in each fuzzy ontology. The main goal of this paper is to present an RDF based ontology, which indeed should contain many facts about the real world, inevitably facing with some uncertainties. In this perspective, an RDF based ontology is converted into a fuzzy most probably an incomplete one due to the fact that there will be some missing relations in the converted fuzzy ontology. To remedy this, the paper introduces a new method in the general framework of conversion and completion of an RDF based ontology into a fuzzy ontology mainly using the facts aspect. Therefore, first a new definition of the fuzzy ontology is proposed. To do so, a neural tensor network, which is indeed state-of-the-art of RDF based ontology completion, is proposed. Furthermore, a new application is suggested for this network that can create a fuzzy ontology. To furnish this goal, two new algorithms are then introduced for the conversion and completion of the proposed fuzzy ontology. In the proposed method, ontology facts are first embedded in a vector space, and then a score value is given to each fact by a learning method. Using these scores and threshold values of each relation, ontology facts can be fuzzified. Finally, some simulation studies are conducted to evaluate better the merit of the proposed method.**

Keywords- Ontology; Fuzzy Ontology; Facts; Neural Tensor Network; RDF.

I. INTRODUCTION

One important application of an RDF based ontology is that it can be used as the background knowledge in the semantic web [1]. In fact, an ontology is a knowledge base in which there are many facts about the real world [2]. In the RDF data model, each of these facts is defined as a triple <e1, R, e2> in which two entities *e1* and *e2* have *R* relation together. The first entity is a subject, the second entity is an object, and the relation is called predicate. For example, the fact of "Farhad was born in Iran" is shown with triple of <*Farhad, born in, Iran*> in which *e1* is Farhad, *e2* is Iran and *R* is *born in*. Ontologies such as YAGO [2], DBpedia [3], Freebase [4] and WordNet [5] are extremely useful resources for query expansion, coreference resolution, question answering, information retrieval, and NLP [6] as the background knowledge.

RDF triples can be extracted from structured and unstructured resources. But in many cases, the accuracy of these triples cannot be certainly determined. These uncertain triples can be represented by fuzzy ontology, and fuzzy logic can be used for the reasoning of these ontologies [7]. So far, many methods have been introduced to create the fuzzy ontologies that each of them represent a special method

* Corresponding Author

www.SID.ir

for this purpose and focuses on some aspects of the ontology, but fewer studies have focused on the converting a standard ontology into the fuzzy one.

There are many definitions for the fuzzy ontologies according to their different aspects. One of the best and general was presented by Bobillo stated as: "a fuzzy ontology is simply an ontology which uses fuzzy logic to provide a natural representation of imprecise and vague knowledge and eases reasoning over it" [11]. By this definition, a fuzzy ontology can have various aspects, and each method is focused on some of them. Here, an investigation of the fact aspect is proposed.

In this paper, a new method is proposed to convert a standard ontology into a fuzzy one with extended facts, such that can solve its incompleteness. A global view of this method is shown in Fig. 1. In this method, it is suggested that new fuzzy facts can be obtained by the neural tensor network (NTN) [43] method in the conversion step. In this step, a score between [0, 1] interval is gained for each fact. With these fuzzy facts, the fuzzy ontology will be constructed. Adding new fuzzy facts into fuzzy ontology increase the power of ontology reasoning. Furthermore, using fuzzy logic, more reasoning on the new fuzzy ontology can be done. By the NTN, this fuzzy ontology can be also completed in the completion step. Details of these steps are explained in the proposed method section.

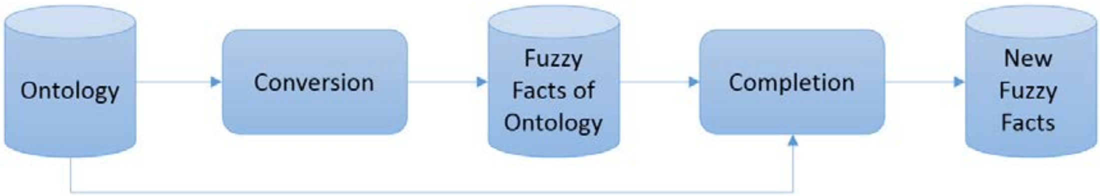

Fig. 1: Global view of the proposed method

After fuzzy ontology creation, it must be represented by a suitable method. For this end, there are many methods, but so far, W3C has been not proposed any standard in this field [8]. For this reason, it is suggested that OWL2 can be used as recently W3C recommendation presented for ontology representation [9, 10]. It is used for fuzzy ontology representation the same as the method of [8]. The contribution of this paper is summarized as follows:

- Introducing a new method for converting a standard ontology into a fuzzy ontology
- Create a fuzzy ontology from the standard one
- Fuzzy ontology completion using a standard ontology
- Creating a fuzzy ontology using the neural tensor network
- Finding a new application for the neural tensor network

The rest of the paper is organized as follows. In section 2, related studies are investigated. Then in section 3, the neural tensor network is briefly introduced. Section 4 presents the suggested method. In this section first a new representation of neural tensor network will be introduced and then a new definition of fuzzy ontology is given. Then, the proposed method is suggested in two parts. In section 5, experimental results are presented, and finally, the results of the paper will be fully discussed.

II. RELATED STUDIES

Research on the fuzzy ontology was started in the early 2000s with the focus on information retrieval aims by Widyantoro [12]. He created a fuzzy ontology of terms. This fuzzy ontology is used to refine a query of the user and was usable in the field of search engine [12]. Some works like [13] represented the aspect of ontology relations as fuzzy. This fuzzy ontology was defined as a pair (C, R_f) where C is a set of domain concepts and R_f is a set of fuzzy binary relations as <r, (c,d), v_f, q_f> where r is the name of the relation, (c,d) is in (C×C), v_f is a fuzzy value for the relation, and q_r is a fuzzy qualifier.

Sanchez et al. made a distinction between the fuzzy ontology and the fuzzy knowledge base which consists of both the fuzzy ontology structure and the set of instances associated with the fuzzy ontology [14]. Their fuzzy ontology definition is given as a tuple <C, R, T, A, X> that C is a set of fuzzy concepts. However, this definition does not clearly specify what makes a concept fuzzy. It does not specify a function from an instance to concept as in [15] to specify the instance's degree of membership in the concept. It does state that the definition of a concept could be inherently vague but does not provide a formalism for handling a vague concept. R is a set of fuzzy relations in C × C. Fuzzy relations as defined here can only be between fuzzy concepts. T is a relation in C × C referred to as a concept hierarchy but also includes mereological (part-of) relationships. A is a set of non-taxonomic associative relationships between concepts relating the concepts across the hierarchical structure. X is a set of ontology axioms expressed in an appropriate logical language to assert class subsumption, equivalence, or more generally to (fuzzily) constrain the possible values of concepts or instances.

Calegari et al. defined a fuzzy ontology as the tuple <I, C, R, F, A> [16]. In this definition, an instance of a concept can have a fuzzy membership degree of belonging to that concept and fuzzy axioms can be specified. I is the set of individuals, also called instances of the concepts. C is the set of concepts. Each concept C ∈ C specifies a fuzzy set on the domain of

www.SID.ir

instances, that is, C: I → [0, 1]. The set of entities of the fuzzy ontology will be indicated by E where E = C ∪ I. R is the set of relations. Each R ∈ R is a n-ary fuzzy relation on the domain of entities so that R: E^n → [0, 1]. This part of the fuzzy definition appears to apply to relations between concepts, and relations between concepts and instances. A special role is held by the taxonomic or subsumption relation T: E^2 → [0, 1] among the entities. Here this definition is somewhat confusing. A subsumption relationship should only exist between concepts T: C^2 → [0, 1] and not between two instances or between a concept and an instance. The C: I → [0, 1] function seems to define the is-a between an instance and a concept C. F is the set of the fuzzy relations on the set of entities E and a specific domain contained in D = {integer, string, ...}. These fuzzy relations are n-ary functions such that each element F ∈ F is a relation F: $E^{(n-1)} \times P \rightarrow [0, 1]$ where P ∈ D. A is the set of axioms defined using a suitable logical language. Predicates are needed to constrain the meaning of concepts, individuals, relationships, and functions.

Another definition of Calegari is slightly modified by separating the intentional component of the ontology [17], i.e., the instances from the extensional component, the ontology definition. A fuzzy knowledge base is defined to include both components, that is, as a tuple (OF, I) where OF = <C, R, F, A> is a fuzzy ontology with each element in the tuple identical to that of in their original definition. As before, I is the set of instances. The use of fuzzy values and how hedges can be used as modifiers to fuzzy values are also described.

Zhai et al. presented a layered ontology structure for defining and using a fuzzy ontology as a 4-tuple <C, P_F, R_F, A_F > [18]. C is a set of concepts where each concept has at least one property whose value must be a fuzzy concept or fuzzy set. P_F is a set of properties. The use of the term “fuzzy concept” is unclear in this definition because the example given is that the value of the property price could be the fuzzy concept cheap. But cheap is not a fuzzy concept in the sense of a fuzzy ontology. It is simply a fuzzy linguistic value. A property $p_F \in P_F$ is a 5-tuple of the form <c, v_F, q_F, f ,U> where c∈C is an ontology concept, v_F represents property values, q_F models linguistic qualifiers which can control or alter the strength of a property value v_F , f is the restriction facets on v_F , and U is the universe of discourse. Both v_F and q_F are the fuzzy concepts at U, but q_F changes the fuzzy degree of v_F. R_F is a set of inter-concept relations between concepts. Like fuzzy concept properties, $r_F \in R_F$ is a 5-tuple of the form <c_1, c_2, t, s_F, U> where c_1 c_2 ∈C are ontology concepts, t represents relation type, U is the universe of discourse , and s_F models relation strengths and is fuzzy concept at U , which can represent the strength of association between concept-pairs <c_1, c_2>. Here again, the use of ‘fuzzy concept’ for modeling relation strengths is confusing; instead what is meant is fuzzy linguistic value. Finally, A_F is a set of fuzzy rules.

In fact, the v_F and q_F were used in [18] before they were used in [13] which also uses v_F and q_F in its r_F tuple to quantify the strength of the relation and to be able to linguistically modify that strength. Because properties can take on fuzzy linguistic values such as cheap, another layer is needed to define a fuzzy linguistic variable ontology. The fuzzy linguistic variable ontology consists of a tuple for a linguistic variable that includes its name, the set of fuzzy linguistic values it may take on, the binary relations between the fuzzy linguistic values including an ordering, and the mapping of the fuzzy linguistic values to their actual fuzzy membership values. This fuzzy linguistic variable ontology is extended with qualifiers such as very and the mapping of qualifiers to their respective fuzzy operators.

In [19], the definition of a fuzzy ontology separates out the crisp and the fuzzy components, that is, crisp concepts, properties and relations from the fuzzy concepts, properties, and relations. A fuzzy ontology is defined as a 7-tuple OF = <C, P, C_F, P_F, R, R_F, As_F, A_F, A>. C is a set of crisp concepts defined for the domain. P is a set of crisp concept properties. C_F is a set of fuzzy concepts where a fuzzy concept c_f possesses at least one fuzzy property p_f from the set of fuzzy properties P_F. R is a set of crisp binary semantic relations defined between concepts in C or fuzzy concepts in C_F. R_F is a set of fuzzy binary semantic relations. As is a set of crisp binary associations defined between concepts in C or fuzzy concepts in C_F. As_F is a set of fuzzy binary associations defined between crisp concepts in C or fuzzy concepts in C_F. A is a set of axioms which are real facts or reasoning rules.

The fuzzy concept of c_f is defined as a triple <N_{cf}, P, Pf> that N_{cf} is a name of the fuzzy concept, for example, Young Person, P is a set of crisp properties P_F is a not empty set of fuzzy properties. A fuzzy property p_f is a triple < N_{pf}, T, M >. N_{pf} is a name of the fuzzy property, e.g., “age”. T is a set of terms which are the values of the fuzzy property, e.g., T= {young, middle aged, old}. M is a mapping rule which maps every term of T to a fuzzy set.

In fact, each of these works considers different aspects of a fuzzy ontology. But in this paper, the aspect of *facts* in RDF based ontologies is considered. There are some works such as [35-42] about this aspect that is not fuzzy, and there are some other works about fuzzy facts aspect as the title of fuzzy RDF [20-25]. In these works, with different methods, RDF triples have been shown as fuzzy. Some other works in the field of fuzzy ontologies can also be fined in [26-34, 45-47], but the focus of this paper is on RDF based ontologies. For this reason, the fact aspect of the ontology is considered to be fuzzified. In the following of the paper, the neural tensor network will be presented to use in proposed algorithms.

www.SID.ir

III. Neural Tensor Network

The NTN was introduced by Socher for the knowledge base completion using existing facts in the knowledge base [6]. This method is state-of-the-art in the field of knowledge base completion [44]. In this section, this neural network is presented that converts each RDF triple into a score. Each relation R of the ontology is described by a neural network in which the entities e1 and e2 are the inputs. If these entities are in that relationship R then the model returns a high score, otherwise a low one. Due to this, any fact can be scored with some certainty mentioned implicitly or explicitly in the knowledge base [43].

The aim of this network is to learn models which have the ability to realize the additional facts that hold purely due to the existing relations in the same knowledge base. The goal of the model is to state whether two entities *e1 and e2* are in a certain relationship R. For instance, whether the relationship *<e1, R, e2> = <Farhad, born in, Iran>* is true and with what score of certainty. It is supposed *that* $e1, e2 \in R_d$ be the vector representations (or features) of the two entities. For now, it can be assumed that each value of this vector is randomly initialized to a small uniformly random number.

The neural tensor network replaces a standard linear neural network layer with a bilinear layer that directly relates the two entity vectors across multiple dimensions. By this model, a score of how probable it is that two entities are in a certain relationship can be computed. Let $e1, e2 \in R_d$ be the vector representation of two entities then, the neural tensor network-based function that predicts the relationship of two entities can be described as equation (1) and Fig. 2 [6].

$$g\langle e1, R, e2\rangle = U^T f(e1^T W_R^{[1:k]} e2 + V_R [e1\ e2] + b_R) \quad (1)$$

Where $f = tanh$ is a standard nonlinearity applied element-wise. $W_R^{[1:k]} \in R^{d \times d \times k}$ is a tensor and the bilinear tensor product $e1^T W_R^{[1:k]} e2$ results in a vector $h \in R^k$, where each entry is computed by one slice $i = 1 \ldots k$ of the tensor: $h_i = e1^T W_R^{[1:k]} e2$. The other parameters for relation R are the standard form of a neural network: $V_R \in R^{k \times 2d}$, $U_R \in R^d$ and $b_R \in R^d$.

Fig. 2 shows a visualization of this model. The main advantage is that it can relate the two inputs multiplicatively instead of only implicitly through the nonlinearity as with standard neural networks where the entity vectors are simply concatenated [6].

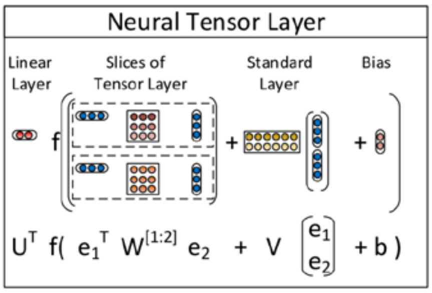

Fig. 2: Neural Tensor Network Model [6]

All models are trained with contrastive max-margin objective functions. The main idea is that each triplet in the training set $T^{(i)} = \langle e1^{(i)}, R^{(i)}, e2^{(i)}\rangle$ should receive a higher score than a triplet in which one of the entities is replaced with a random entity. There are N_R many relations, indexed by $R^{(i)}$ for each triplet. Each relation has its associated neural tensor net parameters. We call the triplet with a random entity corrupted and denote the corrupted triplet as $T_c^{(i)} = \langle e1^{(i)}, R^{(i)}, e_c\rangle$ where we sampled entity e_c randomly from the set of all entities that can appear at that position in that relation. Let the set of all relationships' NTN parameters be $\Omega = u, W, V, b, E$. We minimize the (2) objective function [6].

$$J(\Omega) = \sum_{i=1}^{N}\sum_{c=1}^{C} max(0, 1 - g(T^{(i)}) + g(T_c^{(i)})) + \lambda ||\Omega||_2^2 \quad (2)$$

Where N is the number of training triplets and we score the correct relation triplet higher than its corrupted one up to a margin of 1. For each correct triple, we sample C random corrupted triplets. Standard L_2 regularization of all the parameters, weighted by the hyperparameter λ has been used. Taking derivatives from this objective function is not possible. For this reason, an optimization method [29] has been used for its minimization. Then the model is trained by taking derivatives with respect to the five groups of parameters.

The first step of this method is the learning step in which it is done using available facts in the knowledge base. In this step, standard neural network and tensor parameters are regularized to use in the next step. After regularization of parameters and using the neural tensor network, one score g of equation (1) is assigned to each new fact. Helping this score, the accuracy of the fact is computed [44]. In this method, a fact is accurate or not. But in this paper it is suggested that using the score of fact as its membership function value, it can be converted into fuzzy.

The result of this network showed that this is the best method for knowledge base completion using existing facts [6]. For this reason, in this paper, it is used for the purpose. For using this method must have a representation of it. There is a representation model for the neural tensor network in [43, 44]. Thus, in the next section first, a representation model will be introduced briefly, and then the proposed method is explained.

www.SID.ir

IV. Proposed Method

In Fig. 3, the details of the proposed method are shown. In this figure two steps of conversion and completion are clear. It is obvious that fuzzy ontology is gained from fuzzy facts of ontology and new fuzzy facts. In the proposed method, a new method was suggested to create a fuzzy ontology from standard one and another new method was suggested for completion of the fuzzy ontology using the standard one. In the following, first, representation of the NTN network for the proposed method is introduced. Then, the new definition of fuzzy ontology is presented and is used for converting RDF Based Ontology into Fuzzy Ontology. Then conversion and completion algorithms are introduced.

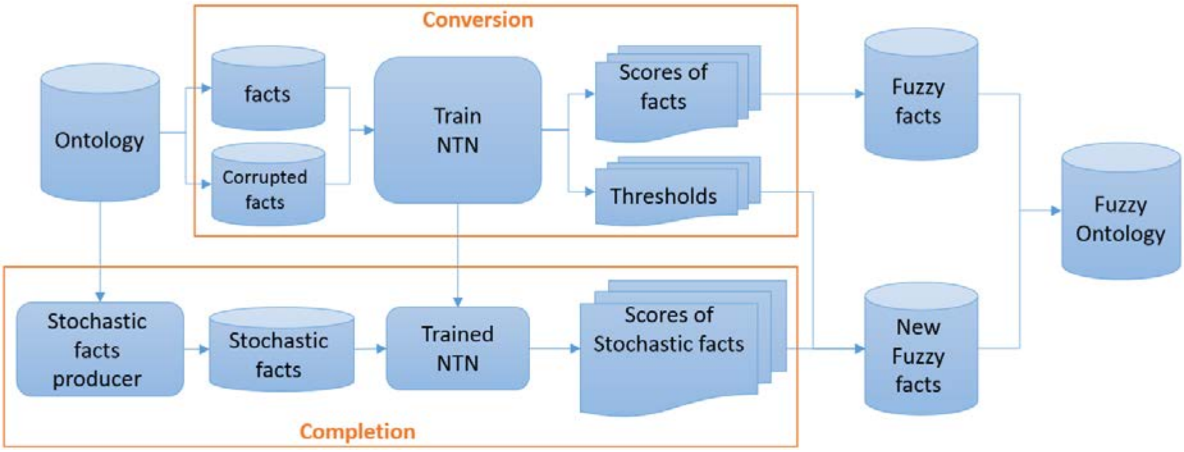

Fig. 3: Details of the proposed method

A. Representation of NTN for Proposed Method

The proposed method of this paper uses the NTN and its representation is necessary. For this reason, in this section, a representation for this network is presented that has been introduced in [43]. In the NTN method, a separate network was considered for each relation. So here, one network will be considered for one relation and then will be generalized to other relations. Therefore, a tensor layer of the NTN for relation *R* with 2 slice and 3-dimensional entity vectors is shown as (2) [43].

$$g = [e1_1\ \ e1_2\ e1_3]\begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}\begin{bmatrix} e2_1 \\ e2_2 \\ e2_3 \end{bmatrix} + [e1_1\ \ e1_2\ e1_3]\begin{bmatrix} w'_{11} & w'_{12} & w'_{13} \\ w'_{21} & w'_{22} & w'_{23} \\ w'_{31} & w'_{32} & w'_{33} \end{bmatrix}\begin{bmatrix} e2_1 \\ e2_2 \\ e2_3 \end{bmatrix} \quad (2)$$

Value of *g* is equal with the score of the network after parameters regularization. This score is allocated to each triple *<e1, R, e2>*. With this definition, a suggested representation of *g* has been presented as (3).

$$W^T = \begin{bmatrix} w_{11},\ w_{21}\ ,w_{31}\ ,w_{12}\ ,w_{22}\ ,w_{32}\ ,w_{13}\ ,w_{23}\ ,w_{33} \\ w'_{11},w'_{21},w'_{31},w'_{12},w'_{22},w'_{32},w'_{13},w'_{23},w'_{33} \end{bmatrix}$$

$$P = [e1_1e2_1, e1_2e2_2, e1_3e2_3, e1_1e2_1, e1_2e2_2, e1_1e2_1, e1_3e2_3, e1_2e2_2, e1_3e2_3]^T$$

$$U^T = [1, 1]$$

$$g = U^T\,(W^TP) \quad (3)$$

This representation has been shown in Fig. 4. In this figure, one score with a value of *g* is considered for each triple *<e1, R, e2>*. The new representation of the tensor layer shows that this layer has been changed with a single layer neural network. By adding other cases of the neural tensor network, Fig. 5 shows the way of obtaining the value of g in the NTN with two slices. This figure is suggested instead of an NTN that was introduced in [6].

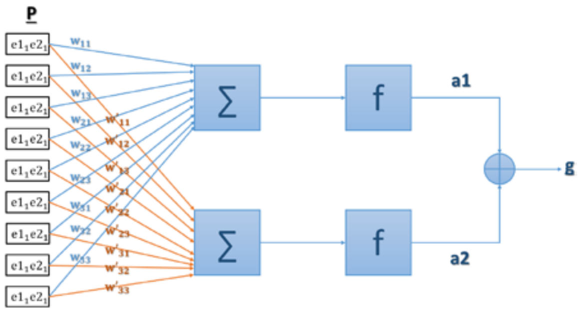

Fig. 4: Suggested representation of the tensor layer

www.SID.ir

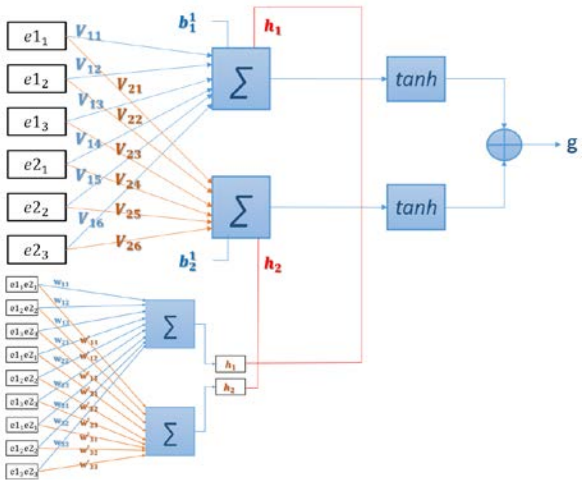

Fig. 5: Suggested representation of the NTN with two slice

In other work, we showed that the neural tensor network can be represented by a standard neural network [43]. But here, the representation of this network is important to use in the next sections in which the way of obtaining *g* is shown. In this paper, it is suggested that the score of *g* for each triple *<e1, R, e2>* is considered as a membership function of the triple for the fuzzy ontology. For this reason in the next section, first, a new definition has been presented for the simple fuzzy ontology and then the way of this fuzzy ontology construction using the NTN is explained.

B. New Definition of Fuzzy Ontology

In this section, a new definition is introduced for fuzzy ontology. This definition is simple for the usage of the special case of fuzzy ontology completion. Each of the past definitions has different fuzzy aspects. A fuzzy aspect of this paper definition is in fuzzy facts and is the same as stated in [13]. Bellow, details are explained.

In the RDF data model, each fact is defined as a triple *<e1, R, e2>* in which two entities *e1* and *e2* have *R* relation together. For example, the fact of "Farhad was born in Iran" is shown with triple of *<Farhad, born in, Iran>* in which *e1* is *Farhad*, *e2* is *Iran* and *R* is *born in*. Here, it is suggested that each two entities *e1* and *e2* can be had a fuzzy relation. With this suggestion, each triple or fact can be a fuzzy triple or fact with a membership function v_f. Equation (4), define this new fuzzy ontology *FO* in which *E* is a set of ontology entities and R_F is a set of fuzzy relations.

$$FO = (E, R_F),\ Where \begin{cases} \mathrm{E} \ = \{e_1, e_2, \dots, e_n\} \\ R_F \ = \{r_1, r_2, \dots, r_m\} \end{cases} \qquad (4)$$

Where each e_i is an ontology entity, each r_i is a fuzzy relation, *n* is the number of entities, and *m* is the number of relations.

Each *fuzzy fact* is expressed as:

$$F_f = < r, (e1, e2), v_f, q_f >,$$

$$Where \begin{cases} (\mathrm{e1}, \mathrm{e2}) \ \in \ \mathrm{E} \times \mathrm{E} \\ v_f \ \in \ [0, 1] \\ \mathrm{r} \ \in \ R_F \\ q_f \in \{low, middle, high\} \end{cases} \qquad (5)$$

In equation (5), $(\mathrm{e1}, \mathrm{e2})$ represents an ordered pair of ontology entities, v_f denotes membership function value of each fuzzy fact and q_f models fuzzy qualifiers that determine accuracy grade of the fact. This definition will be compared with other definitions in the experimental results section. Each of them can be used for a special application of them. In this paper, an aspect of *facts* must be fuzzified to create the special fuzzy ontology for the application of conversion and completion. This is explained below.

C. Converting and Completing RDF Based Ontology into Fuzzy Ontology

Details of the proposed method were shown in Fig. 3. This method consists of two parts. First a new algorithm is proposed to convert a standard ontology into fuzzy one, and in the second part the completion is done by extending this fuzzy ontology.

Conversion

In this section, the algorithm of conversion for the proposed method is introduced. This algorithm is given in *Algorithm 1*. Inputs of this step are the facts of ontology with the structure of *<e1, r, e2>* and its outputs will be fuzzified facts of this ontology. First, some corrupted facts must be produced for each fact. Each corrupted fact is obtained by replacement of an entity of fact with a random entity of ontology. By having facts of ontology and the corrupted facts, NTN can be trained. After training, network parameters are regularized to determine the scores of each fact.

For classification of corrupted facts and ontology facts, a threshold value is also obtained for each relation. Final, each fact of ontology is converted into the structure of fuzzy one with its score as the membership function value and high qualifier. The qualifier of all ontology facts is considered high because they were accurate facts in ontology previously.

Completion

In this section, the algorithm of conversion for the proposed method is introduced. This algorithm is given in *Algorithm 2*. Inputs of this step are the facts of ontology with the structure of *<e1, r, e2>*, threshold values of relations, and NTN parameters after training. The output will be new fuzzy facts that complete the fuzzy ontology.

In this algorithm, using entities of ontology and with different types of their combinations with the

www.SID.ir

relations, new random triples are obtained to complete the fuzzy ontology. Because the accuracy of these triples has not been determined yet, these triples are considered fuzzy. This is obtained using trained NTN with properly tuned parameters.

Algorithm 1. Conversion Algorithm to convert an ontology into fuzzy ontology

Input: F: *Facts of ontology with the structure of <e1, r, e2>*
Output: F_f: *Fuzzy facts of ontology with the structure of <r, (e1, e2),* v_f, q_f *>*
1: produce *corrupted facts* F_c *for each F*
2: train *NTN with* F_c *and F*
3: *obtain score* $\boldsymbol{g}$ *for each* F_c *and F*
4: *obtain thresholds for each relation*
5: end train
6: for *i=1 : number_of_F*
7: return *<*r_i, *(*$e1_i$, $e2_i$*)*, g_i, *high>*
8: end for

Algorithm 2. Completion Algorithm to complete a fuzzy ontology using ontology facts

Inputs: F: *Facts of ontology with the structure of <e1, r, e2>*
T: *thresholds that obtained from algorithm 1 for each r*
NTN parameters after training in algorithm 1
Output: F_f: *New fuzzy facts with structure of <r, (e1, e2),* v_f, q_f *>*
1: *produce some stochastic facts* F_s *with replacement of facts entities with one of E*
2: for *i=1 : number_of_*F_s
3: *compute* $\boldsymbol{g_i}$ *using trained NTN*
4: if $g_i >= T_r$ then
5: return *<*r_i, *(*$e1_i$, $e2_i$*)*, $\boldsymbol{g_i}$ *high>*
6: else
7: if $g_i > T_r/2$ then
8: return *<*r_i, *(*$e1_i$, $e2_i$*)*, $\boldsymbol{g_i}$, *middle>*
9: else
10: return *<*r_i, *(*$e1_i$, $e2_i$*)*, $\boldsymbol{g_i}$, *low>*
11: end if
12: end if
13: end for

Each new triple is given to the network and then based on equation (3) and Fig. 4, the value of g is obtained as a membership function value v_f of the triple. So, new triples can be converted into *< r, (e1, e2),* v_f, q_f *>* structure as fuzzy facts. The accuracy of each fuzzy fact is explained by the fuzzy aspect of $v_f = g$, too. In this algorithm, the qualifier is dependent on the g value that is clear in lines of 4 to 12. Therefore, the fuzzy ontology is completed by these fuzzy triples as fuzzy facts.

V. Experimental Results

To implement the proposed method, a standard WordNet dataset, which was introduced in [6], has been used as an RDF based ontology sample. Properties of this dataset are given in Table 1. This dataset is used because the NTN method was evaluated on this dataset.

Table 1: WordNet dataset

#R	#Entities	#Train	#Test
11	38696	112581	10544

In this dataset, there are 112581 RDF triples of WordNet that were used to train the neural tensor network. These triples are used as RDF based ontology sample that must be converted into the fuzzy ontology. In Table 2, the proposed method is compared with similar studies that were discussed in the related studies section. While the fact aspect is fuzzified in this paper, this aspect is not considered in similar studies. This aspect is general and in some cases requires more deep considerations, because many triples of an ontology are not indeed true facts.

As it is clear in this table, each method defines its fuzzy ontology, based on some aspects for a special domain. This issue was investigated in [47]. In this paper, each fact (ontology triple) is fuzzified, separately. It can be used for the applications in which the triples are uncertain. In these applications, the ontology can be converted and completed by the proposed method. Therefore in this paper, a new perspective of fuzzy ontologies is considered. Since there is no other fuzzy ontology on the aspect of the

www.SID.ir

fact, it is not an easy task to do comparison study. Related studies such as [13-19] are also evaluated only their proposed methods. Thus, here, only the implementation results of the method proposed in the paper will be evaluated. Nevertheless, aspects, advantages, disadvantages, and domain of related works are compared with those of the proposed method as shown in Table 2. Despite the fact that the problem of fuzzy ontology learning has not received much attention, a learning method is used in the proposed method.

Table 2: Comparison with similar works

Definitions	aspects	Advantages	Disadvantages	Domain	Source
<*r*, (*c*, *d*), v_f, q_f>	*Relations*	Handling inconsistent triples of ontology, Ability to store imprecise concept definitions, Using reasoning and fuzzy set theoretic	There is no distinction between the fuzzy ontology and the fuzzy knowledge base, And only cover Relation aspect, and it does not use the learning method	knowledge description	[13]
<*C, R, T, A, X*>	*Relations, Concepts, Axioms*	The distinction between the fuzzy ontology and the fuzzy knowledge base	This method does not clearly specify what makes a concept fuzzy, and does not specify a function from an instance to concept, and does not use learning method	Semantic Web	[14]
<*C, R, T, A, X*>	*Relations, Concepts, Axioms*	specify the instance's degree of membership in the concept	This method does not provide a formalism for handling a vague concept, and dose does not use a learning method	Integrating Fuzzy Logic in Ontologies	[15]
<*I, C, R, F, A*>	*Relations, Concepts, Individuals*	an instance of a concept can have a fuzzy membership degree of belonging to that concept and fuzzy axioms can be specified	It cannot separate the intentional component of the ontology, and does not use the learning method	Fuzzy Description Logics	[16]
(OF, I) *where OF* = <*C, R, F, A*>	*Relations, Concepts, Individuals*	separating the intentional component of the ontology	It cannot cover the property aspect of ontology, and does not able to linguistically modify that strength, and does not use the learning method	Ontology Editor	[17]
<C, P_F, R_F, A_F >	*Relations, Concepts, Properties*	Ability to linguistically modify that strength, Using a layered ontology structure is presented for defining and using a fuzzy ontology	The use of the term "fuzzy concept" is unclear in this definition, and dose does not use learning method	Knowledge Management	[18]
<$C, P, C_F, P_F, R, R_F, As_F, A_F, A$>	*Relations, Concepts, Properties, Terms, Axioms*	It separates out the crisp and the fuzzy components, and cover more aspects	The fuzziness is not in the serves relation, and does not use the learning method	Fuzzy ontologies building	[19]
<*r*, (*e1*, *e2*), v_f, q_f >	*Facts*	It uses the learning approaches and embedding Method, It is suitable for the applications in which the triples are uncertain, it covers the fact aspect	Fact aspect is general, it is suitable only for RDF based ontologies, this method needs enriched datasets	RDF based ontology	Suggested Method

In the implementation step, the ontology must be converted into the fuzzy one through the proposed method. For this, the following steps have been taken. A part of RDF ontology is shown in Table 3, before the conversion step and a part of the implementation result of Algorithm 1 on WordNet dataset is shown in Table 4. There are 112581 triples as facts in the dataset. After the training of these facts, the values of network parameters are obtained. Due to the accuracy of ontology facts, qualifier values of these facts will be high in the fuzzy ontology. Therefore, the main part of fuzzy ontology is equal with <*r*, (*e1*, *e2*), v_f, q_f > in which $v_f = \boldsymbol{g_i}$ as membership function value for fuzzy triple and <*r*, (*e1*, *e2*)> is same as <*e1*, *r*, *e2*> in the standard ontology. Therefore the original ontology is converted into a fuzzy one.

In Table 4, column 1 is the number of first entities e1, column 2 is the number of relation r, column 3 is the number of second entities e2, column 4 is shown corrupted facts by -1 and ontology facts by 1, column 5 is the aim value of each triple in the test dataset, column 6 is the membership function value v_f, and column 8 is the qualifier. In the WordNet dataset, there are also 11 relations that their threshold values after implementation are shown in Table 5.

www.SID.ir

Table 3: A part of RDF ontology before conversion step

e1	R	e2	#e1	#R	#e2
__medical_aid_1	type_of	__irrigation_2	2389	2	11297
__entandrophragma_1	member_holonym	__african_scented_mahogany_1	15839	4	17142
__life_6	has_instance	__animation_3	1702	1	9491
__arundo_1	member_meronym	__family_graminaceae_1	10589	3	113

Table 4: A part of fuzzy RDF ontology after a conversion step

#e2	#R	#e1	t	a	g	Threshold	q
2389	2	11297	1	1	0.1643	0.336	high
15839	4	17142	1	1	0.1659	0.266	high
1702	1	9491	1	1	0.3713	0.326	high
10589	3	113	1	1	0.3838	0.286	high

In Algorithm 2, different compositions of 38696 entities with 11 relations are used to obtain new random triples. These triples can help to complete the fuzzy ontology though the accuracy of these triples has not been determined yet. For this reason, these triples must be fuzzified. These fuzzifications can be obtained using a trained NTN. The NTN implementation conditions are the same as those in [43,48-50]. In Table 6, a part of fuzzy RDF ontology is shown, after the completion step. In this table, the score of each new triple (that is added in the completion step) is compared with the threshold, and then, its *q* qualifier is detected. Thus, we have a fuzzy ontology in which there are fuzzified ontology facts and additional triples that can be considered as the facts with a score and *q* qualifiers.

Table 5: Threshold values of 11 WordNet relations

#Relation	1	2	3	4	5	6	7	8	9	10	11
Threshold	0.3260	0.3360	0.2860	0.2660	0.1560	0.2460	-0.1740	0.3360	-0.0540	0.1860	-0.1440

Table 6: A part of fuzzy RDF ontology after completion step

#e1	#R	#e2	t	a	g	Threshold	q
15447	2	3676	-1	-1	0.3812	0.336	low
11297	2	2389	1	1	0.1643	0.336	high
11297	2	30450	-1	-1	0.3451	0.336	middle
17142	4	15839	1	1	0.1659	0.266	high
17142	4	16980	-1	-1	0.3445	0.266	low
9491	1	1702	1	1	0.3713	0.326	high
9491	1	29864	-1	-1	0.3116	0.326	middle
113	3	12793	1	1	0.2746	0.286	middle
113	3	14563	-1	-1	0.3833	0.286	low
113	3	10589	1	1	0.3838	0.286	high
113	3	27131	-1	1	0.2732	0.286	middle

www.SID.ir

Discussion

So far, there is a fuzzy ontology in which there are fuzzy facts with different v_f. While the accuracy of ontology completion of NTN was 86.2% in [6] meaning that 13.8% of new triples is wrong and some accurate triples do not exist, in this method, each triple is either true or false. In this paper, it is suggested that each triple can be accurate with a membership function value v_f and qualifier of q_f. For example, after training the NTN, triple of <mouse, type of, cat> is considered as an accurate fact, while this is not indeed true. In the proposed method, a membership function value is assigned to this triple that is less than an accurate triple (such as <tiger, type of, cat >).

With this suggestion, fuzzy ontology tools can be used for more reasoning in compared to the method proposed in [6]. Using methods of [25-32], this goal can be reached. If is represented as these methods are, the fuzzy ontology becomes more powerful. This new method can be also used to complete other fuzzy ontologies. The new triples can be added into a new fuzzy ontology for the completion.

To evaluate the proposed method, the test set of the dataset has been used. This set has 10544 true or false triples. The membership function values of these triples are available in the fuzzy ontology. Using threshold values and evaluating new facts, the accuracy of this method got as high as the one in [6] (86.2%). Moreover, in the proposed method, new reasoning abilities are also obtained to previous state-of-the-art advantages.

VI. CONCLUSION

The main goal of the paper was on the creation and completion of an RDF based fuzzy ontology. For this objective, a new method was introduced to convert and yet complete an RDF based ontology into a fuzzy ontology. While previous methods did not consider the fact aspect of ontology to be fuzzified, based on which a new definition of fuzzy ontology was firs tproposed. To obtain a fuzzy version of the ontology, a neural tensor network was proposed as the state-of-the-art of RDF based ontology completion methods. In addition, a new application was suggested for this network that can create a fuzzy ontology. For this goal, two new algorithms were proposed for the conversion and completion of fuzzy ontology, in which ontology facts were embedded into a vector space while a score value was given to each fact by a learning method. Using these scores and threshold values of each relation, ontology facts were fuzzified.

Knowing the fact that the proposed method uses a learning approach by an embedding method (NTN), it requires enriched datasets though its accuracy in these datasets is acceptable. The method proposed is obviously suitable for the applications in which the triples are uncertain. While it is true that the fact aspect covered is general, the proposed method is suitable only for RDF based ontologies. Because there is no other fuzzy ontology on the fact aspect, the implementation results of the proposed method were only evaluated and compared with those of the related studies.

In future work, the proposed method can be used to complete an existing fuzzy ontology. Mostly, existing fuzzy ontologies have missing relations between some entities. These relations can be obtained by the proposed completion method fully discussed in the paper, in which each fuzzy ontology must be embedded into a vector space. As a final remark, other aspects of ontology can be added in case for other applications.

REFERENCES

[1] Berners-Lee, Tim, James Hendler, and Ora Lassila. "The semantic web." *Scientific american* 284.5 (2001): 28-37.
[2] Suchanek, Fabian M., Gjergji Kasneci, and Gerhard Weikum. "Yago: a core of semantic knowledge." *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007.
[3] Bizer, Christian, et al. "DBpedia-A crystallization point for the Web of Data." *Web Semantics: science, services and agents on the world wide web* 7.3 (2009): 154-165.
[4] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD international conference on Management of data, SIGMOD, 2008.
[5] G.A. Miller. WordNet: A Lexical Database for English. Communications of the ACM, 1995.
[6] Socher, Richard, et al. "Reasoning with neural tensor networks for knowledge base completion." Advances in Neural Information Processing Systems. 2013.
[7] Cross, V. V. "Fuzzy ontologies: The state of the art." *Norbert Wiener in the 21st Century (21CW), 2014 IEEE Conference on*. IEEE, 2014.
[8] Bobillo, Fernando, and Umberto Straccia. "Fuzzy ontology representation using OWL 2." *International Journal of Approximate Reasoning* 52.7 (2011): 1073-1094.
[9] F. Bobillo and U. Straccia. An OWL ontology for fuzzy OWL 2. In J. R. et al., editor, Proceedings of the 18th International Symposium on Methodologies for Intelligent Systems (ISMIS 2009), volume 5722 of Lecture Notes in Computer Science, pages 151{160. Springer-Verlag, 9 2009.
[10] B. Cuenca-Grau, I. Horrocks, B. Motik, B. Parsia, P. F. Patel-Schneider, and U. Sattler. OWL 2: The next step for OWL. Journal of Web Semantics, 6(4):309{322, 2008.
[11] F. Bobillo, Managing vagueness in ontologies, PhD Dissertation, University of Granada, Spain, 2008.
[12] D.H. Widyantoro and J. Yen, "Using fuzzy ontology for query refinement in a personalized abstract search enginer," in IFSA World Congress and 20th NAFIPS International Conference, Joint 9th (Volume:1) 2001, pp. 610 – 615.
[13] L. Dey, and M. Abulaish, "Fuzzy ontologies for handling uncertainties and inconsistencies in domain knowledge description," in Proceedings IEEE International Conference on *Fuzzy* Systems (IEEE World Congress on Computational Intelligence), 2008, pp 1366-1373.
[14] Elie Sanchez, Takahiro Yamanoi and D'Amico Fuzzy Ontologies for the Semantic Web. FQAS 2006, pp. 691-699.

www.SID.ir

[15] S. Calegari, D. Ciucci, "Integrating Fuzzy Logic in Ontologies," in Manolopoulos, Y., Filipe, J., Constantopoulos, P., Cordeiro, J., eds.: ICEIS, INSTICC press, 2006, pp. 66–73.
[16] S. Calegari, D. Ciucci, "Fuzzy Ontology, Fuzzy Description Logics and Fuzzy-OWL," WILF 2007, pp. 118-126.
[17] S. Calegari, D. Ciucci, "Towards a Fuzzy Ontology Definition and a Fuzzy Extension of an Ontology Editor," in ICEIS (Selected Papers) 2006, pp. 147-158.
[18] Jun Zhai Lixin Shen Zhou Zhou Yan Liang, "Fuzzy Ontology Model for Knowledge Management," 2007 International Conference on Intelligent Systems and Knowledge Engineering (ISKE), 2007.
[19] H. Ghorbel, A. Bahri, R. Bouaziz, "Fuzzy ontologies building method: Fuzzy OntoMethodology," in Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS), 2010, pp.1-8
[20] Lv, Yanhui, Zongmin M. Ma, and Li Yan. "Fuzzy RDF: A data model to represent fuzzy metadata." *Fuzzy Systems, 2008. FUZZ-IEEE 2008.(IEEE World Congress on Computational Intelligence). IEEE International Conference on*. IEEE, 2008.
[21] Ma, Zongmin, et al. *Fuzzy Knowledge Management for the Semantic Web*. Springer-Verlag Berlin and Heidelberg GmbH & Company KG, 2014.
[22] Zhang, Fu, et al. "Fuzzy semantic web ontology learning from fuzzy UML model." *Proceedings of the 18th ACM conference on Information and knowledge management*. ACM, 2009.
[23] Vaneková, Veronika, et al. "Fuzzy RDF in the semantic web: Deduction and induction." *Proceedings of Workshop on Data Analysis (WDA 2005)*. 2005.
[24] Straccia, Umberto. "A minimal deductive system for general fuzzy RDF." *Web Reasoning and Rule Systems*. Springer Berlin Heidelberg, 2009. 166-181.
[25] Manolis, Nikos, and Yannis Tzitzikas. "Interactive exploration of fuzzy RDF knowledge bases." *The Semantic Web: Research and Applications*. Springer Berlin Heidelberg, 2011. 1-16.
[26] Bobillo, Fernando, et al. "Fuzzy description logics under Gödel semantics." *International Journal of Approximate Reasoning* 50.3 (2009): 494-514.
[27] Stoilos, Giorgos, Giorgos Stamou, and Jeff Z. Pan. "Fuzzy extensions of OWL: Logical properties and reduction to fuzzy description logics." *International Journal of Approximate Reasoning* 51.6 (2010): 656-679.
[28] Garcı, Àngel, Eva Armengol, and Francesc Esteva. "Fuzzy description logics and t-norm based fuzzy logics." *International Journal of Approximate Reasoning* 51.6 (2010): 632-655.
[29] Cerami, Marco, Àngel García-Cerdaña, and Francesc Esteva. "On finitely-valued fuzzy description logics." *International Journal of Approximate Reasoning* 55.9 (2014): 1890-1916.
[30] Borgwardt, Stefan, and Rafael Peñaloza. "Consistency reasoning in lattice-based fuzzy description logics." *International Journal of Approximate Reasoning* 55.9 (2014): 1917-1938.
[31] Asma, Djellal, and Boufaida Zizette. "Fuzzy Ontology Evolution: Classification of a New Individual." *Journal of Emerging Technologies in Web Intelligence* 6.1 (2014): 9-14.
[32] Bobillo, Fernando, and Umberto Straccia. "Aggregation operators for fuzzy ontologies." *Applied Soft Computing* 13.9 (2013): 3816-3830.
[33] Alexopoulos, Panos, et al. "IKARUS-Onto: a methodology to develop fuzzy ontologies from crisp ones." *Knowledge and information systems* 32.3 (2012): 667-695.
[34] Bobillo, Fernando, Juan Gómez-Romero, and Pilar León Araúz. "Fuzzy Ontologies for Specialized Knowledge Representation in WordNet." *Advances on Computational Intelligence*. Springer Berlin Heidelberg, 2012. 430-439.
[35] *Matthew Horridge, Sean Bechhofer The OWL API: A Java API for OWL Ontologies* Semantic Web Journal 2(1), Special Issue on Semantic Web Tools and Systems, pp. 11-21, 2011.
[36] Farhad Abedini, Fariborz Mahmoudi, Amir Hossein Jadidinejad. *OFE: Ontological Facts Extraction from text for computing Semantic Relatedness*. In proceedings of IEEE 3th International Conference on Machine Learning and Computing (ICMLC 2011), Singapore, 26-28 February 2011, pp 84-88.
[37] Farhad Abedini, Fariborz Mahmoudi, Amir Hossein Jadidinejad. *A New Disambiguation Method for Semantic Entity Extraction Using YAGO Ontology*. In proceedings of IEEE 3th International Conference on Machine Learning and Computing (ICMLC 2011), Singapore, 26-28 February 2011, pp 79-83.
[38] Farhad Abedini, Fariborz Mahmoudi, Amir Hossein Jadidinejad. *SESR: Semantic Entity Extraction for computing Semantic Relatedness.* In proceedings of 4th International Conference on Advanced Computer Theory and Engineering (ICACTE2011), Dubai, UAE, 28-30 December 2011, pp 225-228.
[39] Farhad Abedini, Fariborz Mahmoudi, and Amir Hossein Jadidinejad, "From Text to Knowledge: Semantic Entity Extraction using YAGO Ontology," *International Journal of Machine Learning and Computing* vol. 1, no. 2, pp. 113-119, 2011.
[40] Farhad Abedini, Fariborz Mahmoudi, and Seyedeh Masoumeh Mirhashem, "Using Semantic Entity Extraction Method for a New Application," *International Journal of Machine Learning and Computing* vol. 2, no. 2, pp. 178-182, 2012.
[41] Farhad Abedini and Seyedeh Masoumeh Mirhashem, "From Text to Facts: Recognizing Ontological Facts for a New Application," *International Journal of Machine Learning and Computing* vol. 2, no. 3, pp. 183-187, 2012.
[42] Farhad Abedini and Seyedeh Masoumeh Mirhashem, "Entity Disambiguation in Text by YAGO Ontology," *International Journal of Computer Theory and Engineering* vol. 5, no. 3, pp. 432-435, 2013.
[43] Abedini, F., M.B. Menhaj, and M.R. Keyvanpour, *An MLP-based representation of neural tensor networks for the RDF data models.* Neural Computing and Applications, 2019. 31(2): p. 1135-1144.
[44] Abedini, F., Menhaj, M. B., & Keyvanpour, M. R., *Neuron Mathematical Model Representation of Neural Tensor Network for RDF Knowledge Base Completion.* Journal of Computer & Robotics, 2017. 10(1): p. 1-10.
[45] Cross, V. and S. Chen. *Fuzzy Ontologies: State of the Art Revisited*. 2018. Cham: Springer International Publishing.
[46] Lin, L., J. Liu, and Y. Zheng. *Construction and Application of Fuzzy Ontology*. 2018. Singapore: Springer Singapore.
[47] Samani, Zahra Riahi, and Mehrnoush Shamsfard. *The State of the Art in Developing Fuzzy Ontologies: A Survey*. *arXiv preprint arXiv:1805.02290* (2018).
[48] Abedini, F., M.B. Menhaj, and M.R. Keyvanpour, *EPCI: An Embedding method for Post-Correction of Inconsistency in the RDF Knowledge Bases.* IETE Journal of Research, 2019.
[49] Abedini, F., M.R. Keyvanpour, and M.B. Menhaj, *Neural Tensor Network Training Using Meta-Heuristic Algorithms for RDF Knowledge Bases Completion.* Applied Artificial Intelligence, 2019. 33(7): p. 656-667.
[50] Abedini, F., M.R. Keyvanpour, and M.B. Menhaj, *Correction Tower: A general embedding method of the error recognition for the knowledge graph correction.* International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI), 2019.

www.SID.ir

AUTHOR BIOGRAPHIES

Farhad Abedini is a Ph.D. candidate at Qazvin Islamic Azad University, Qazvin, Iran. He received his B.Sc. in Software Engineering from Tabarestan University, Chalus, Iran. He also received his M.Sc. degree in Software Engineering from Qazvin Islamic Azad University, Qazvin, Iran. His research interests include Information Retrieval, Knowledge Discovery and Data Mining.

Mohammad Reza Keyvanpour (corresponding author) is an Associate Professor at Alzahra University, Tehran, Iran. He received his B.Sc. in Software Engineering from Iran University of Science &Technology, Tehran, Iran. He also received his M.Sc. and Ph.D. degree in Software Engineering from Tarbiat Modarres University, Tehran, Iran. His research interests include E-Health, knowledge Discovery and Data Mining.

Mohammad Bagher Menhaj, is full professor of "Amirkabir University of Technology", Tehran, Iran. He has received his Ph.D. degree from school of Electrical and Computer Engineering at OSU in 1992. After completing one year with OSU as postdoctoral fellow, in 1993, he joined the Amirkabir University of Technology, Tehran, Iran. From December 2000 to Aug. 2003, he was with school of Electrical and Computer Engineering and Department of Computer Science at OSU as visiting faculty member and research scholar. He is author and co-author of more than 500 technical papers, and four books: Fundamentals of Neural Networks, 1998, and Application of Computational Intelligence in Control, 1998, Neural Networks, 2000, and Fuzzy Computations, 2007, all in Persian.

www.SID.ir