Research of Knowledge Reasoning in Supply Chain

Yiqing Lu

School of Information Management Beijing Information Science and Technology University, Beijing, 100192, China.

luyiqing@126.com

Abstract

The knowledge reasoning can be used by the inference engine to acquire the implicit knowledge in supply chain,. The reasoning based on the ontology semantic can be directly used the rules and axiom. Four kinds of semantic relations are studied in this paper. Then the Tool of establish reasoning rules is introduced.

Keywords

Supply chain; Ontology; Knowledge reasoning; Knowledge sharing.

1. Introduction

From the perspective of knowledge management, it not only refers to the supply chain of material, but also that of knowledge. Knowledge has gradually become an important factor to promote the collaboration of supply chain. In the collaboration process, on the one hand, the member enterprises in the supply chain should spread the knowledge, on the other hand, they should absorb the knowledge from other member enterprises to realize the complementation of knowledge resource [1]. Scientifically managing and applying the knowledge in the supply chain can optimize the business association among the member enterprises, and improve the agility, flexibility and innovative efficiency of the supply chain, thus improving the whole performance, and achieving sustainable competitive advantages in the fierce market competition.

Ontology technology can be used to achieve the knowledge sharing of supply chain [2]. However, in the ontology, most of the knowledge is implicit rather than explicit declaration. In addition, the ontology application can not explicitly express all the implicit knowledge, for it can make the ontology huge and inefficient in that way. In order to solve these problems, the inference engine can be used to reason and acquire the knowledge. The efficiency of knowledge acquisition is determined by the complexity of ontology and optimization degree of knowledge organization, and the concrete realization needs some production rules cooperated with search algorithm to query related contents.

2. Organization of the Text Reasoning based on Semantic

The reasoning based on the ontology semantic can be directly used the rules and axiom to reason, such as subClassof and equivalentClass. In the reasoning process, it should consider the relationship between classes, class and individual (instance), instance and instance (correlation, contain and custom), and object attribute relationship (parent, transitive, reciprocal, and symmetry relations) [3]. In the reasoning process, it can determine whether an individual is the instance of one or more classes, whether it is the instance of some certain class, the relationship between two instances, the hierarchical architecture reasoning of the instance and class (attribute) having a specific relationship with certain instance[4].

This paper has mainly studied four kinds of semantic relations. One is Is-a relation among the concepts at different logic levels; One is Instance-of relation between the concepts at different logic level and the extension individual of the concept; Thirdly is Instance-Instance relation among concepts at the same logic level or different logic levels; Fourthly is SubAttribute-of relation reflecting the relations at different abstraction degrees[5][6].

2.1. Is-a Relation

Is-a relation is the typical binary relation among the concepts, which can be used to point out the affiliation relation of the abstraction concepts among things. It can form the logical hierarchical classification structure among concepts, and it is similar to the parent relationship of the objects. The formal definition of Is-a relation is as follows:

Definition 1 For the concepts C1, C2∈Sc in the ontology concept set Sc, if the connotation I(C1) of concept C1 contains the connotation I(C2) of C2, namely I(C1)⊃I(C2), and the extension E(C1) of the concept C1 contains the extension E (C2) of C2, namely E (C1)⊃ E (C2), then the relationship between the concepts C1 and C2 is Is-a relation, marked as Is-a (C1,C2). The concept C1 can be called the sub-concept or species concept, while the concept C2 can be correspondingly called father concept or genus concept.

Is-a relation cannot satisfy the symmetry, while it has reflexivity, anti-symmetry and transitivity. The knowledge reasoning rule (quasi natural language description) based on Is-a relation is as follows:

Transitivity rule: (Is-a (C1, C2)∧Is-a(C2, C3))→Is-a(C1, C3)

Object attribute inheritance rule: (Is-a (C1, C2)∧HasAttribute(C2, A))→HasAttribute(C1, A)

Data attribute inheritance rule: (Is-a (C1, C2)∧HasProperty(C2, P))→HasProperty(C1, P)

Instance transitivity attribute rule: (Is-a (C1, C2)∧Instance-of(e, C1))→Instance-of (e, C1)

Parent reciprocal relation rule: Subclass Of (C1, C2)→superClassOf (C2, C1)

The transitivity rule of is-a relation can be mainly used to determine the parent hierarchy among multiple concepts. The transitivity attribute rule of instance can check the instances of the concept. The attribute inheritance rule can be used in the inheritance of subclass on the parent class attribute.

2.2. Instance-of Relation

Instance-of relation is the typical binary relation between the concept and the instance, which is similar to the relation among the object-oriented class and object. Suppose the concept extension set of any concept C in the concept set Sc is E(C) ={x|x⊂C}; For any element Ci∈E(C) in E(C), if Ci's extension set E(Ci)={Ci}, Ci is the instance of the concept C, while the instance set Sin(C) can be defined as Sin(C)= {x|x∈E(C)∧E(x)={x}}. The above formal definition on the instance has distinguished the sub-concept and instance of the concept from the concept extension. The formal definition of concept C, instance set element e, and Instance-of relation is as follows: for the concept C and the instance set Sin, the relationship between element e（e∈Sin) in the instance set Sin and the concept C can be called the instance relation, marked as Instance-of (etc.). The knowledge reasoning based on the instance relation can be achieved through the inheritance rule.

All kinds of instances and the object class declared in the domain ontology can satisfy the object attribute inheritance rule and data attribute inheritance rule. The object attribute inheritance rule of Instance-of relation can be used to deduce the possible relationship between the instance and other object instances.

2.3. Instance-Instance Relation

For the semantic relationship among the instances at same concept or different concepts, such as the instance X1, X2 and relation R, the formal definition of Instance-Instance relation is: for certain instance X1∈SA in the instance set SA of the ontology concept A, there is at least one instance X2∈SB satisfying R relation in the instance set SB of the ontology concept B, namely R(Xl, X2), while the relation R is the relation between concept A and instance B.

The generalization form of simple rule of the relation among instances is:

Symmetrical relation reasoning rule: (SymAttribute(A)∧A(e1,e2))→A(e2, e1)

Transitivity relation reasoning rule: TraAttribute(A)∧A(e1,e2)∧A(e1,e2)→A(e1, e3)

Reciprocal relation reasoning rule: (AthAttribute(A1, A2)∧A1(e1,e2)) →A2(e2, e1)

SymAttribute (A) shows that the relation (object attribute)A has symmetry, which is an instance of the symmetric relation class. TraAttribute (A) shows that relation A has transitivity, which is an instance of the transitivity relation class. While AthAttribute (A1, A2) shows that the object attributes A1 and A2 are the reciprocal relation. ei (i∈N) shows the ontology instance.

2.4. Subattribute-of Relation Reasoning Rule

SubAttribute-of relation is the typical binary relation among the relations (object attributes), which can be used to point out the affiliation relation in the relation attributes of things, and it forms the logical hierarchical classification structure among the relationships. For the given relations of R1 and R2, the formal definition of SubAttribute-of relation is: for the relations R1, R2∈S in the ontology relation set SR, if any one instance (xi,yi) in the instance set R1(x,y) of relation R1 has R2 relation, while any one instance (mi,nj) in the instance set R2(m,n) of relation R2 does not necessarily have R1 relation, or namely the instance set R1(x,y)⊂R2(m,n), then the relationship between R1 and R2 is SubAttribute-of relation, marked as SubAttribute-of(R1, R2), the relation xx is called the sub-relation, and xx is called the parent relation.

The SubAttribute-of relation do not satisfy the symmetry, while it has reflexivity, anti-symmetry and transitivity. The reasoning rule based on SubAttribute-of relation is as follows:

Transitivity rule: (SubAttributeOf (A1, A2)∧SubAttributeOf (A2, A3)→SubAttributeOf (A1, A3)

Attribute extension rule: (HasAttribute (C, A1)∧SubAttributeOf (A1, A2)→HasAttribute (C, A2)

Attribute extension generalization rule: (A1 (e1, e2)∧SubAttributeOf (A1, A2))→A2(e2, e1)

Parent attribute reciprocal relation rule: SubAttributeOf (A1, A2)→SuperAttributeOf (A2, A1)

In the attribute extension generalization rule, A1 (e1, e2) shows that the instance e1 has the object attribute A1 with the value of e2. The attribute transitivity rule is mainly used to determine the hierarchical relation among multiple object attributes. In the reasoning process, it is usually used as the intermediate rule preparing for other rule reasonings. The attribute extension rule is mainly used to determine the semantic relation between multiple object attributes and single class. The attribute extension generalization rule is used to determine the relationship of two instances in two parent object attribute level. In the SubAttribute-of relation, based on the consideration of the data attribute of instance, the most important thing is to combine the parent hierarchy structure of the relation to reason and achieve more specific and detailed relationship between two instances. In addition, the detail degree of instance relation information is corresponding with the level of sub-relation.

3. Tool of Establish Reasoning Rules

Jena of HP can be used to establish the reasoning rules. Jena is a Java framework for building Semantic Web applications. It provides a extensive Java libraries for helping developers develop code that handles RDF, RDFS, RDFa, OWL and SPARQL in line with published W3C recommendations. Jena includes a rule-based inference engine to perform reasoning based on OWL and RDFS ontologies, and a variety of storage strategies to store RDF triples in memory or on disk [3]. Jena provides a collection of tools and Java libraries to help you to develop semantic web and linked-data apps, tools and servers. The composition and reasoning principle of Jena is as shown in Figure 1.

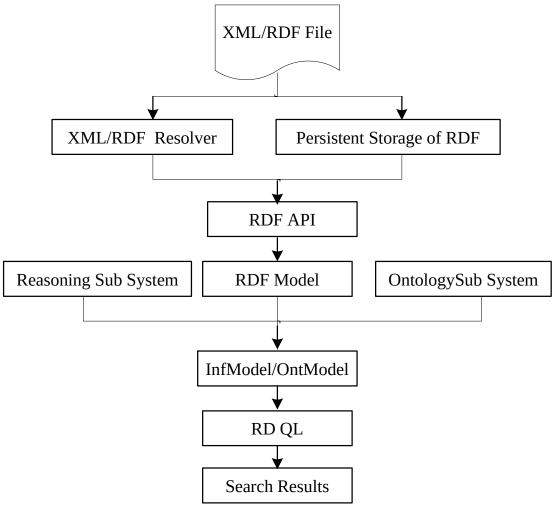

Figure 1. Composition and reasoning principle of Jena

The Jena Framework includes:

an API for reading, processing and writing RDF data in XML, N-triples and Turtle formats;

an ontology API for handling OWL and RDFS ontologies;

a rule-based inference engine for reasoning with RDF and OWL data sources;

stores to allow large numbers of RDF triples to be efficiently stored on disk;

a query engine compliant with the latest SPARQL specification;

Servers to allow RDF data to be published to other applications using a variety of protocols, including SPARQL.

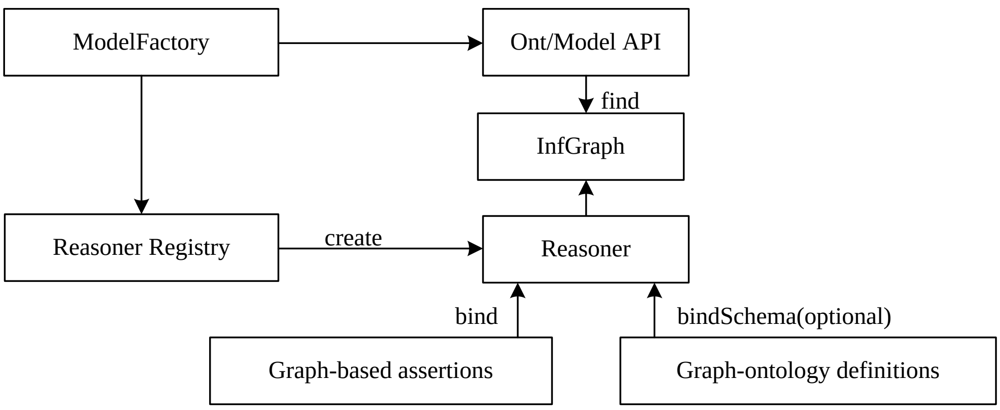

Figure 2. Structure of Jena

The rule-based inference engines Jena provided include RDF Reasoner, RDFS Reasoner, and OWLReasoner, and it contains the general inference function. Users can define the reasoning rule according to own needs, register and use the third-party inference engine. The working mechanism of the inference engine can be shown in Figure 2. The working principles of the inference engine are: 1) the reasoning registration mechanism creates the inference engine according to the triple and ontology of basic RDF. 2) The inference engine can generate the model object containing the inference mechanism. 3) Operate and process the model with Model API or Ontology API.

References

[1] Arshinder, Arun Kanda, S.G. Deshmukh. Supply chain coordination: Perspectives, empirical studies and research directions. Int. J. Production Economics 115(2008), pp: 316-335.

[2] Jaehun Joo, Sang M. Lee. Adoption of the Semantic Web for overcoming technical limitations of knowledge management systems. Expert Systems with Applications, 36(2009), pp: 7318-7327.

[3] Boris Motik, Ian Horrocks, Ulrike Sattler. Bridging the gap between OWL and relational databases. Web Semantics: Science, Services and Agents on the World Wide Web, 7(2009), pp: 74-89.

[4] Grigoris Antoniou, Frank van Harmelen. A semantic web prime, MIT press, 2004.

[5] C. Huang, S. Lin. Sharing knowledge in a supply chain using the semantic web. Expert Systems with Applications, 37(2009) , pp: 3145-3161.

[6] Y. Tian, N. Li, Y. Zhang. Research on office document processing based on ontology. Journal of Beijing information science and technology university. 25(Z2)(2010) 97-102.