IJCTA, 9(17) 2016, pp. 8659-8663

A Problem Solver Mechanism for Mathematical Text Predicates Using Ontology

Swathy Vodithala[*] and Suresh Pabboju[**]

ABSTRACT

Most of the problem solving application domains uses mathematics as its base background. The mathematical domain which solves numerous problems uses a number of predicates, known as mathematical predicates. The problem with the mathematical predicate is that a single predicate say '+' has many text predicates such as sum, total, add etc. In order to use these mathematical predicates, we need to develop a method for describing them semantically. The semantical description can be clearly done by using ontology. This paper describes different ways of developing an ontology that further can be applied to any application which uses mathematical predicates.

Keywords: ontology; ER diagram; slot and filler; semantic network; list; predicate.

I. INTRODUCTION

The meaning of ontology in philosophy is generally considered to be as "theory of existence" [1] . In computer science, ontology is specifying the terms and the relationship among them in a particular domain formally. Ontology was initially used by AI laboratories but now it is gradually moving to the desktop of the domain experts [2]. The development of ontology can be divided into two phases Specification and Conceptualization [3][6]. The former phase constitutes acquiring of informal knowledge about the relevant domain and in the later phase the structure and organization of the above acquired knowledge is achieved by external representations. These representations must be free from the implementation languages and environments.

There are different categories of ontology's based on the degree of conceptualization [4].

- Top-level ontology: It defines general notations, that are not dependent on a particular domain. Ex: time, space etc.
- Domain ontology: It defines notations of concepts which are particular to a domain. This as well includes the relationships among the concepts. Ex: banking, Forestry etc
- Task/application ontology: It defines notations of concepts that are particular to a domain or even a particular task. Ex: Alumni information system of a college.

II. RELATED WORK

2.1. Knowledge Representation

In previous years the usage of tables or database scheme would be sufficient to describe the relationship between the data that yields answers to many queries [8][7]. Even though it solves many queries the database

[*] Assistant Professor, Dept. of CSE, KITS, Warangal, *E-mail: swathyvodithala@gmail.com*
[**] Professor, Department of IT, CBIT, Gandipet, Hyderabad, *E-mail: plpsuresh@gmail.com*

scheme's data is only at the physical level. Table 1 can answer only simple queries such as "which branch XYZ belongs to" etc.

The disadvantage of above representation is that the inferential capabilities are very low. It cannot answer a simple query as "who has scored the best", without providing a set of additional rules to the database. But, nowadays more inferential information is required to solve many problems i.e. semantically inferring the data is given the prominent role. There are different mechanisms to provide the semantic level information, and one among such is "ontology".

Table 1
Simple relational knowledge

Name	*Roll No*	*Branch*	*Grade*
ABC	120	CSE	9
XYZ	121	ECE	10
MNO	122	EEE	8

2.2. First order predicate

The prepositional logic which was used previously has a limited power to express as it considers only the facts [5]. Whereas, First order logic (FOL) considers objects, relations and functions. Since FOL considers the world entities it is widely used as a problem solving technique in knowledge representation.

Example: X killed Y

The above sentence is represented as killed(X, Y) in the first order logic [9]. Here killed is known as "predicate symbols" or the relationship between X and Y. X and Y are considered to be the objects or "constant symbols". Since the FOL is using predicates, it is also known as first order predicate.

III. PROPSED WORK

There are many rules to build an ontology. One such rule to build ontology is that, it should be a repetitive or iterative process. But the important thing to be considered while building an ontology is there is no exact way to model a particular domain, i.e. the best solution always depends on the application on which the user is working.

Our proposed work demonstrates on how to build ontology for the mathematical predicates. A predicate is a self explaining hypothetical language. Some examples where the mathematical predicates are used are

Example 1:

Natural language Sentence: x adds with y (1)

Mathematical notation of (1): x+y (2)

Conversion of "(2)" to First order predicate
(mathematical symbol predicate) : +(x,y) (3)

Different users can interpret the above mathematical symbol predicate into different ways of mathematical text predicate. Some of them are shown below

Conversion of "(3)" to First order predicate
(mathematical text predicate): add(x,y) (4)

Conversion of (3) to First order predicate
(mathematical text predicate) : sum(x,y) (5)

Conversion of (3) to First order predicate

(mathematical text predicate) : total(x,y) (6)

The problem with the conversion to text predicate from symbol is that even though “(4)”, “(5)” and “(6)” are semantically equal they cannot be traced by the machine. In order to solve the problem mentioned we should build an ontology for the text predicates.

Example 2:

Sentence : average of x and y (7)

Mathematical notation: $\mu(x,y)$ (8)

First order predicate : average(x,y) (9)

First order predicate : mean(x,y) (10)

“Example 2” refers to the average of two numbers i.e. it involves both addition as well as division i.e. average =(x+y)/2. The need of ontology is equally important to describe the involvement of other terms within a single term.

The following work explains how to build an ontology for the mathematical predicates in all the different possible ways.

3.1. Entity relation diagram

ER diagram is one of the traditional way to represent the tables in database schemes. This ER method can also be used to represent the ontology . In the below figure, the rectangles represent the entities or terms and the directions represent the relations between them. This is also called as slot and filler or semantic network, and in artificial intelligence it is used to represent knowledge. In slot and filler structure, links are considered to be the attributes and Boxed nodes as objects, values or terms. “1”on the link represents that both terms are equal and is represented with a bidirectional arrow. “2” represents that one term is part of the other term, so it is represented with a single directional arrow.

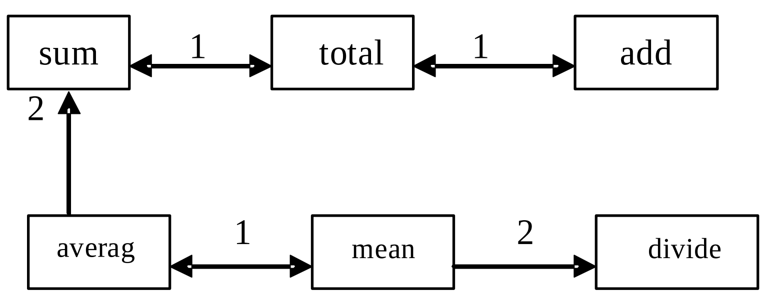


Figure 1: Slot and filler /ER diagram

3.2 Listing

The ontology can also be represented as a list of terms ,i.e. average has both addition as well division of numbers. Average and mean are synonyms whereas sum and divide are subparts of average or mean. Further the synonyms of sum and divide are also listed as shown below

1. a) Average
   b) Mean
   1.1 a) Sum
   b) summation

c) total

d) add

e) addition

1.2 a) divide

b) division

Figure 2: Listing

3.3. Tree representation

The Tree diagram is one of the most important data structure representation, that can also be used to represent the ontology. In tree representation the root node becomes the key term i.e., Level 0 and level 1 are considered to be the synonyms of level 0 and level 2 are partially related to the root node. As the level number increases the similarity or relationship between the terms corresponding to that level and root term decreases.

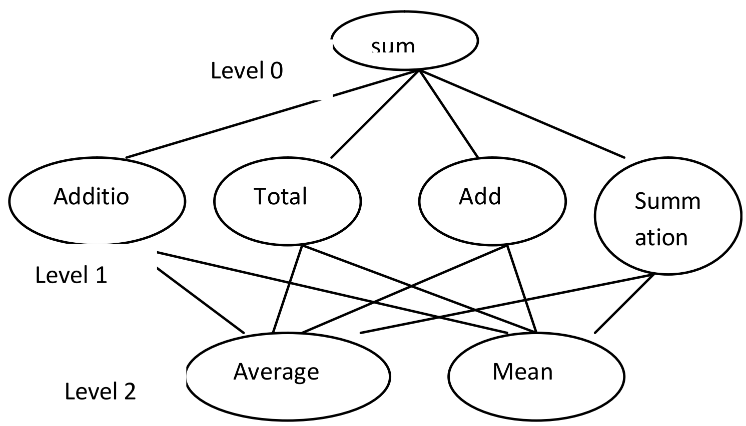


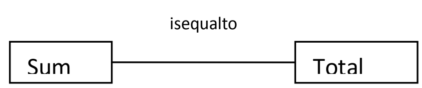
Figure 3: Tree representation

3.4. Predicate representation

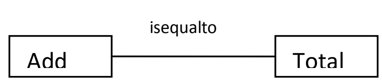
The ontology for mathematical terms can also be represented by using first order predicate.

Examples:-

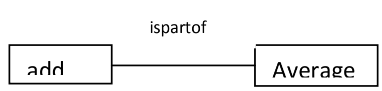
(1) First order predicate: isequalto (sum, total)



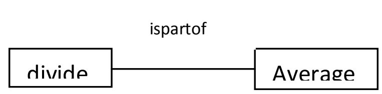
(2) First order predicate: isequalto (add, total)



(3) First order predicate: ispartof (add, average)



(4) First order predicate: ispartof (divide, average)



IV. CONCLUSION

There are different approaches for developing an ontology and this paper focuses on the development of ontology for the mathematical text predicates. The ER diagram shows the entities as well relations among the entities. It can be also viewed as semantic network or slot and filler. Some of the other ways for representing the ontology are tree, list and first order predicate. There are even many other ways to create an ontology and the choice of ontology mainly depends on which problem , and the solution that we want to derive for a particular application.

The problem with building ontology is that it is very expensive. The better solution to this problem is to reuse the existing ontology's.

REFERENCES

[1] Tom Gruber, "Ontology", entry in Encyclopedia of Database Systems, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2008.

[2] Natalya f. noy and Deborah l. mcguinness, "Ontology development 101: a guide to creating your first ontology",Stanford university, Stanford, CA, 94305.

[3] Graciela Brusa, Ma. Laura Caliusco and Omar Chiotti,"A Process for Building a Domain Ontology: an Experience in Developing a Government Budgetary Ontology", Australasian OntologyWorkshop (AOW 2006), Hobart, Australia. Conferencesin Research and Practice in Information Technology, Vol. 72. M. A. Orgun and T. Meyer, Eds.

[4] Frehiwot Fisseha, "The Basics of Ontologies", Nordic Agricultural Ontology Service (AOS) Workshop, Royal Veterinary and Agricultural University, Copenhagen, Denmark, February 28, 2003.

[5] Nicola Guarino, Massimiliano Carrara and Pierdaniele Giaretta,"An Ontology of Meta-Level Categories", Knowledge Representation and Reasoning: Proceedings of the Fourth International Conference (KR94), Morgan Kaufmann, San Mateo, CA.

[6] Andrew Krizhanovsky and Feiyu Lin, "Related terms search based on wordnet / wiktionary and its application in ontology matching", Proceedings of the 11th Russian Conference on Digital Libraries RCDL'2009, Petrozavodsk, Russia, 2009.

[7] Elaine Rich and Kevin Knight, "Artificial intelligence", Tata McGrawHill.

[8] Ayesha Banu, Syeda Sameen Fatima and Khaleel Ur Rahman Khan, "Semantic – based querying using ontology in relational database of library management system", International Journal of Web & Semantic Technology (IJWesT) Vol.2, No. 4, October 2011, DOI : 10.5121/ijwest.2011.2402 21.

[9] Swathy vodithala, Dr Suresh Pabboju, "A clustering technique based on the specifications of software components", IEEE International Conference on Advanced Computing and Communication Systems (ICACCS -2015), 5th to 7th January, 2015, Coimbatore, India.