IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 3, Ver. V (May-Jun. 2016), PP 89-93
www.iosrjournals.org

Design of Layers in Knowledgebase for Expert Systems

Yugandhara A. Nimbalkar[1,] Dr. Justus S[2]
[1,2]*School of Computer Science and Engineering VIT University* Chennai, India

Abstract: *In any Expert System, Knowledge is the basic functional unit for building a knowledgebase [1]. Hence, Expert Systems are totally/partially depended on Knowledge bases for its intelligent functionality. In our proposed design of Knowledgebase, we have divided it into 3 layers: Logical or Design Layer, Knowledge Layer and Storage Layer.In Logical layer or Design layer a high-level statement or a file is the input, which is processed, and clauses (functional & predicate symbols) are extracted [3]. These will form the basic unit for designing a knowledge unit (KU). These KUs are validated for their scope with identified relations. Then we will create various set of domain-dependent predicates and functions to represent the Knowledge in the Knowledge layer. This representation will be converted to Object Relational Database and will be stored in Storage Layer [2]. Researchers working in design and development of Knowledgebase will be benefitted by this.*

I. Introduction

Data is the collection of raw facts. The facts or the data about the world are put into Knowledgebase. Knowledgebase can be used to infer to various facts of the world with their proper reasoning. We use these facts, some logic rules and other logic forms to deduce new facts. We can also provide the inconsistencies of some facts available in knowledgebase.

The Knowledgebase in Experts Systems are layered as follows: Design or Logical Layer, Knowledge Layer and Storage Layer. Design or Logical Layer gets its input from Knowledge Acquisition phase is completed by extracting the data or the raw facts from one source usually human experts and then organizing and storing this data. The next layer is an intelligent layer of the system-Knowledge Layer. In this layer, new facts are deduced from the existing facts and also some inconsistencies can be deduced from them. The third and the last layer of the system is Storage Layer. The extracted facts in acquisition phase are classified into types: Predicate Symbols, Functional Symbols and connectives. These symbols are then stored in the form of Object Relational Database Systems in the Storage Layer. All these symbols together form Ontology for an Organization.

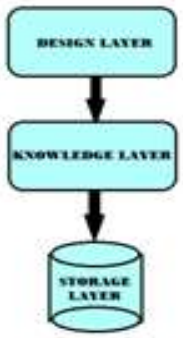

Figure 1. Knowledge Management System

II. Related Work

Three layered Database Management Systems is compared with Knowledge Management Systems and many similarities are found between their concepts. The Presentation Layer of the Database Management System can be compared with Design Layer of Knowledge Management System. In Knowledge Management Systems, Design Layer acts as an interface between the system and user while in Database Systems, Presentation Layer acts as interface [1]. Similarly, Application layer of DBMS is used to perform operations while Knowledge layer of KMS is used to deduce new facts from existing ones. Both Storage layer and Database layer of KMS and DBMS are used to store the knowledge and data. In the design layer of the KMS, knowledge is acquired from various trusted sources such as Experts in that particular field, various Magazines and Journals, meetings and Forums. The knowledge collected is then classified and stored. This knowledge can later be updated by Experts when needed using the concept of Knowledge Reuse [5]. Refer Figure 2.

III. Proposed Work

The entire work is divided into three phases, each associated with the three layers of the KB. In this work we are focusing on the first layer – the design layer.

The phases involved in the design of any Knowledge Management System are:

1. Knowledge Acquisition
2. Knowledge Representation
3. Knowledge Storage & Retrieval
4. Knowledge Sharing
5. Knowledge ReuseRefer Figure 3.

A. Knowledge Acquisition

Knowledge Acquisition is done in the Design Layer of the KMS model.

The functions of the design layer are:

1) Processing of input text or file
2) Extract tokens from the input string/file
3) Identify the clauses and predicates
4) Represent the text in the form of FOL.

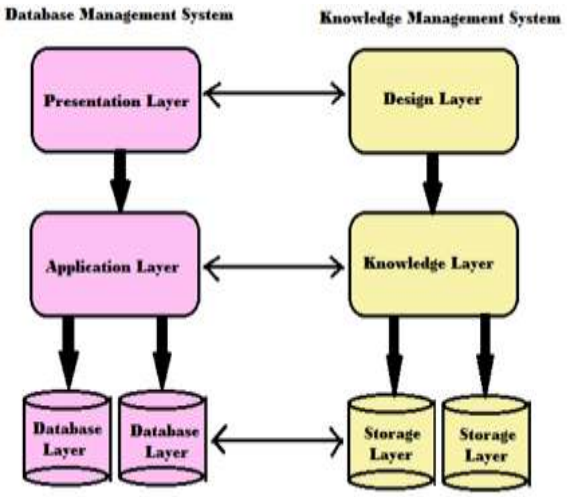

Figure2. Comparison Between Database Management Systemand Knowledge Management System.

Figure 3. Phases involved in design of Knowledge Management System

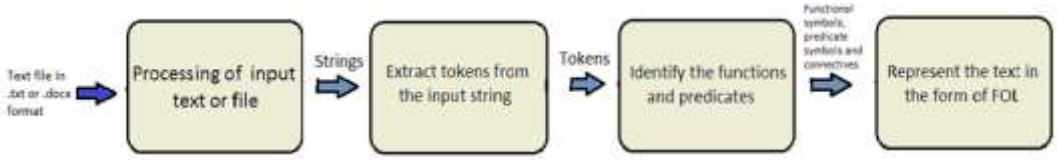

Figure 4. Functions of Knowledge Acquisition Phase

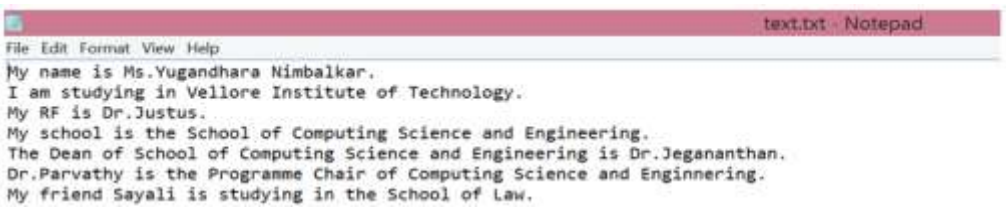

Figure 5. Input text to the System in the '.txt' format

We will explain these functionalities in detail in the following sub-sections. Refer Figure 4.

1. Processing of input text or file

The input to the system is a text. We can either import a file or use a string of characters as an input. This text may be a paragraph or a mere sentence. The format of the text taken is '.txt'. The text of '.docx' format can also be given to the system. Refer Figure 5.

2. Extract tokens from the input string/file

The imported file is read and stored it in buffer to tokenize the clauses. Clause is nothing but a collection of literals [3]. The tokens of words are created in Java using built-in class *StringTokenizer*. Token is a single atomic element in a string. We are using space as the condition to make tokens out of a string.

***3*. Identify the Functional and Predicate symbols**

The next step is to classify various function symbols, predicate symbols and connectives. All the Connectives are predefined in the system so that they can be sorted easily. So our main aim in this step is recognize Functional symbols and Predicate symbols. In the above example, functional symbols are 'Name', 'Yugandhara Nimbalkar', 'Vellore Institute

Technology', 'RF', 'Dr.Justus', 'My friend Sayali', 'Dr.Parvathy', 'Dean', 'School of Computing Science', 'School of Law'. Predicate symbols are 'studyingIn', 'doing', 'under', 'from', 'of', 'is'.

It is hereby proposed the idea for representing the high level text in FOL:We will use the Natural Language Processing concepts are used for linguistic and semantic knowledge from the text for learning.

1. Using the Natural Language Processing Parser, divide the sentence into
- Subject
- Verb
- Object
2. Exclude all the stop-words like 'the', 'a' and 'an'.
3. Make the separate clauses of subject, verb and object.
4. Verb present will form the Predicate Symbols while the Subject and the Object will form the Function Symbols.

'Is' is an important verb in English language. Sometimes it acts as a supporting verb and sometimes it acts as a main verb. So based on this verb, this algorithm is developed which can create clauses and distinguish whether it is Functional symbol or Predicate symbol

The Proposed Algorithm is:

```
Input: Text in the form of sentences related to
Ontology.
Output: FOL form of the sentences.
Variables:
    objectInitiator='the' or 'a' or 'an' or 'Dr. or
'Mr.' or 'Mrs.' Or 'Ms.';
    dropOff ='the' or 'a' or 'an';
    Clause1, Clause2, Clause3;

Algorithm:
For (each sentence in paragraph)
  Search for 'is' in the sentence
  Put all the strings in Clause1 until 'is' is
found
  If the start of the Clause1 is dropOff
       Drop the dropOff;

  If (no string present between 'is' and
objectInitiator)
      Put all the strings after objectInitiator in
      Clause2
      Return Clause1(Clause2);
Else if (some string is present in between 'is' and
objectInitiator)
    Put all the strings after 'is' (including it) in
    Clause2
    Put all the strings after objectInitiator
    (including it) in Clause3
    If (objectInitiator == dropOff)
          Drop the dropOff from Clause3
    Return Clause2(Clause1, Clause3)
```

5. Represent the text in the form of FOL

The higher level text or query that is processed in the first step is converted to the First-Order Logic (FOL). This is necessary in the design layer so that the concepts extracted are validated for correct and concrete representation. The functional symbols are represented as capital letters (P, Q etc.) and the Predicate symbols are represented as capital letters followed by parenthesis. Example X(A,B) or Y(P,Q).

In addition to these the set operators (Union, Intersection), relational operators (<, >, =), and logical operations (subsumes, presumes, entails etc.). The given text needs to be converted as the FOL grammar for deriving logical entailments of the concepts and their relations.

B. Knowledge Representation

Using various rules and logics, relations are developed between these symbols. Here, in this layer, the relations between this symbols can be represented using various methods.

1. First Order Logic
2. Frames
3. Description Logic

1) First Order Logic

In First Order Logic, quantified variables are used. Instead of propositions, variables are used in this type of representation. In FOL, facts and rules are made up of Predicate Symbols and Functional Symbols.The above text can be represented in FOL as

- MyName(Yugandhara)
- studyIn (I, Vellore Institute of Technology)
- myRF (Dr.Justus)
- mySchool (School of Computing Science and Engineering)
- deanOf (Dr. Jegananthan, School of Computing Science and Engineering) and so on.

Frame is another way of representing knowledge.There are two types of frames-

- Individual Frame- to represent single object.
- Generic Frame-to represent categories or classes of objects.

Above text can be represented in the form of Frames as follows:

- (**School**

<:Name SchoolName>
<:Dean Person>
<: Location Address>
<: NoOfFaculties (IF NEEDED
{let x<-SELF:1}
…)

- (**Student**

<: IS-A Person>
<: RegistrationNo Number>
<: Year year>
…)

- (Yugandhara

<: INSTANCE-OF Student>
<: PhoneNo Number>
<: Hobbies Reading>…)

- (Person

<: myName name>
<: Age Number>
<: BirthDate Date>
<: City Location>
IF-ADDED :myName(name)
…)

3) Description Logic (DL)

Description Logic is used in Knowledge representation of important concepts from application domain. The important terms used in this type of representation are concepts, role and individuals, and their relationships.

Above text can be represented using DL as follows:

- (School =
  [**EXISTS**20: Students]
  [**EXISTS**1 Dean]
  [**EXISTS**1 Course]
  [**ALL** Faculty Course]
  [**ALL**ResearchFacilitator Faculty]
  …)

- (Student =
  [**EXISTS** 1 RegistrationID]
  [**FILLS** Pass Criteria]
  [**EXISTS** 27 Credits]
  …)

- (Person=
  [**EXISTS** 1 Name]
  [**EXISTS** 1 Address]
  [**FILLS** Finite Age]
  …)

C. Knowledge Storing

The symbols in the Conceptual graphs is then stored in the Storage layer in the form of Object Relational Database System.

D. Knowledge Sharing& Reuse

The Knowledge stored in Storage layer of one KMS can be shared with any other KMS. Knowledge Units are modeled to concepts and objects in Object-relational database. This will enable Object-based storage (OBS) of KUs.When some new facts are added to the system, the existing knowledge is updated or modified accordingly. Knowledge Reuse is related to the reusability of the concepts from the existing KMS.

IV. Conclusion

Thus we have extracted the clauses from the text. These clauses are domain dependent. Then the clauses are identified as the Functional symbols and Predicate symbols using the 'YIS' algorithm. The clauses are then represented using FOL form and then stored in the database or knowledgebase. If 'Is' is supporting verb in the sentence, then clauses will form the Functional symbol and if 'is' is main verb in the sentence then clauses will form the Predicate symbols.

References

[1]. 1.Ronak Kharadkar and Justus S," Analytical Validation of 3 layer knowledgebase design", Conference on Science Engineering and Technology, VIT University,2014.
[2]. RonakKharadkar and Justus S," Designing Knowledge Representation Framework for ICD-10", 1st International conference on futuristic trend in computational analysis and knowledge management (ABLAZE 2015),2015
[3]. Ronald J. Brachman and Hector Levesque," Knowledge Representation and Reasoning",1st Edition May,2004
[4]. Wei-Tek, Rama Vishnuvajjala and Du ZhangYan, "Verification and Validation of Knowledge Based Systems," IEEE Transactions on Knowledge and Data Engineering, vol. 11, no.1, January 1999
[5]. Amar Arbouh, Mohamed Menacer and Yasser M. Alginahi, "Design Concepts of a KnowledgeBase System for Research and Studies in Quran and Its Science",2013 Taibah University International Conference, December 2013