International Journal of Applied Business and Economic Research

ISSN : 0972-7302

available at http: www.serialsjournals.com

Volume 15 • Number 22 • 2017

Concept-Relation Constructs for Knowledge Representation

S. Praveena Rachel Kamala[1] and Justus Selwyn[1]

[1]School of Computer Science and Engineering, VIT University, Chennai Campus, Chennai – 600 127, Tamil Nadu, India. E-mail: praveena.rachel2013@vit.ac.in, justus.s@vit.ac.in

ABSTRACT

In the recent days, the data is available in huge volume. For accessing the exact information it takes a longer time for retrieval or the result remains irrelevant to the search - due to redundancy. This scenario can be overcome by having a systematic way of representing knowledge. We have proposed models based on Concept Relation (CR) graph. This enables us to easily embed the data and readily take back the information. The novelty in our model is identifying the concepts from the given document, assigning a unique token for each concept reducing redundancy and extracting the knowledge related to the context. This process involves creation of a knowledge base to represent and store the concepts along with their relationship with the other concepts. This enables easy extraction of meaningful knowledge. This model can easily adapt to any domain without vigorous training of data sets or pre defining the concepts10. These features will make our model successful.

Keywords: Knowledge Representation; Conceptual Graph; Knowledge Based System; CR Graph; Knowledge Retrieval; Knowledge Storage.

1. INTRODUCTION

The flow of data has become very common and catering the needs of people is very easy, if the necessary resources are available. From buying medicine to any electronic appliances, different online and mobile shopping apps are available. But no complete model for representing knowledge is available.

The existing knowledge representation models demonstrate on the following tools. For getting the data a sequential or hierarchical method, for storing the information a network or table format and for reasoning fuzzy Petri-net technology are used.

These conventional techniques for capturing data, take a longer time to process and store. Moreover same data will be placed in different locations based on their order of occurrence. This also results in presence of the same data under different categories. Searching and sharing of information becomes more complex and complicated because of these problems. Absence of correlation of data ends up in delivering mismatched answers for the query posted by the user.

Selecting a more appropriate mode for storage plays an important role in adding, modifying and searching of information. In terms of representation, a systematic structure is required for settling the data. For retrieval of information, a less complex and less time consuming method is well suited for the scenario. These requirements are widely absent in the existing models.

In case of reasoning, mostly only two extreme options are considered. The intermediate or the minor issues which could bring major changes to the situation are not measured. The outcome of the question can be different from the answer which was predicted by the system. Thus the reasoning module in a knowledge based system is always a flaw.

To overcome all these disadvantages, a detailed study on all the available models and techniques was done. Resulting is the proposed model for representing knowledge. We have concentrated on a systematic way to interpret the data, and by carefully placing it in a well-organized structure abiding the rules and restrictions.

Our proposed models for representing knowledge use the logical concepts like predicate logics, and first order logic for the data to be interpreted. Description logic is applied to form rules and restrictions. These identified concepts are mapped to a graphical structure to determine the relationship between each clause. This enables the construction of Concept Relation (CR) graph to be simple.

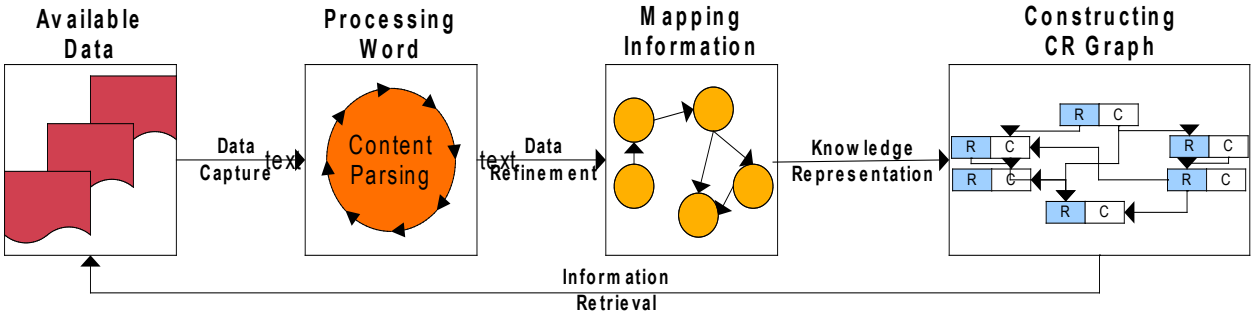

Figure 1: A Model for Representing Knowledge

The figure 1 explains the overall working of the system. Our model supports any kind of data from any source. After the data is fed into the structure, content parsing is done where each and every word in processed. Depending upon the nature of their occurrence (*i.e.*) verb, noun, connector, etc. their role is determined. Then the graph is built based upon their association with the neighboring nodes.

Conceptual Graph (CG)[7] concept designed by John Sowa1, defines the conceptual form of representing knowledge. The concepts are represented as nodes in the graph having logic based relationship. A CG based knowledge representation model facilitates a clear and structured approach for placing the data. CG also permits a more systematic move towards reasoning. The graphical arrangement of CG enables the user to clearly monitor the structure and operational procedure of the system. Studying the advantages of CG, we have incorporated this graphical concept into our model for more clarity, functional structure, logical entailment, and for user assistance.

In this paper, the processes involved in a knowledge based system are discussed in Section 2. In Section 3, algorithms for acquiring the content, segregation of concept and relation, modification of concept and searching for existence of a concept are discussed. In Section 4, the summary and conclusion is given for the model designed.

2. PROCESSES IN KNOWLEDGE BASED SYSTEMS:

The different stages involved in the process of knowledge based system are knowledge representation, knowledge reasoning, knowledge storage and retrieval.3

Knowledge Representation involves in receiving the data and placing the information in the repositories. The concepts are identified and their relationship with the other concepts is associated.

Knowledge Reasoning is gaining knowledge from information which is plotted in the knowledgebase. The efficiency of the system is measured from the accuracy of reasoning.

Knowledge Storage is placing the information in a location from where it can be accessible for later use.

Knowledge Retrieval plays an important role in extracting the correct knowledge depending upon the context and the user's requirement.

2.1. Proposed Model

In this paper, the different phases of the process of knowledge based system are discussed. Initially the fetching of data from any source, splitting of words obtained through content parsing, mapping the clause of information with each other and storing the grouped concept into the CR graph as nodes are done.

Several algorithms are designed for operating the graphical structure. The different operations done in a Concept Relation (CR) graph are adding new nodes, modifying the existing nodes and finding the existence of the desired node.[2]

The design used for retrieving the knowledge from the graphical structure is like a magnetic ball effect. When a concept is searched and found, then all the related concepts (nodes) will be fetched depending upon their level of relationship. Thus the retrieval of the stored concept is carried out following the path of their relation.

3. ALGORITHM

Our Model comprises of various stages in producing meaningful extraction of knowledge. This is achieved by processing the data through different levels. The working procedure and algorithms is discussed below.

3.1. Algorithm for Content Parsing

This method of capturing data4 plays a vital role in knowledge representation. The meaning of the source should remain the same while obtaining the facts. There should be no loss of records. The location of storage should support easy fetching of information for further manipulation.

Content_Parsing(word_document)
Input: Data content as .txt or .doc
Output: Parsed content in .csv
Procedure:
OPEN the text file READ the content of the file SPLIT each and every word separately OPEN an excel file WRITE the words in different column STORE the content CLOSE the files

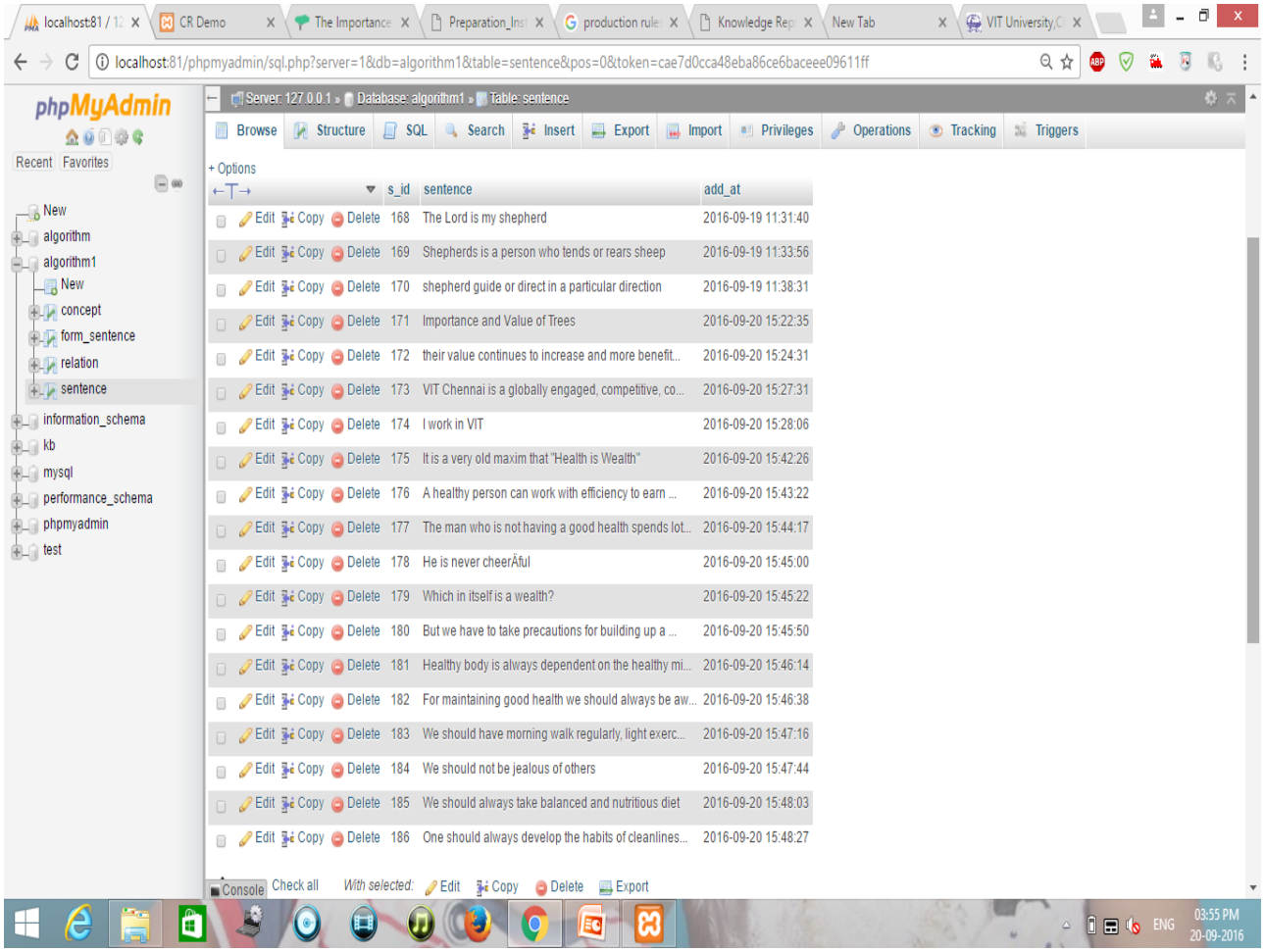

Figure 2: Content Stored in Sentence Table with Sentence ID

In this algorithm Content_Parsing the word document from where data has to be taken is treated as the input. This file is opened; all the lines in the file are read and stored in the cache memory. These data are split word by word and rewritten in the memory. A spreadsheet is opened5 and the records are added by writing the split words onto different cells in column wise. Then the newly created spreadsheet is saved and closed for later use. These captured words are added in the sentence table for computational purpose.

A unique number is generated based upon the existence of the word and a sequence number is also automatically generated as the words have to be arranged in correct order. The data captured is analyzed by checking the text document containing the list of relations defined. If the word is already defined in relation list then it is treated as relation otherwise it has to verify with the list of concepts enrolled in the concept table. If the concept is already present then the existing unique number is fetched and used. Adding new concepts to the list requires generation of new unique number.

The Concept Table is created, if no concepts are already present. Updating of Concept Table is done only one or more concepts are available and only if the concept is not present earlier. If the concept previously exists then the unique number is retrieved for completing the sentence table.

The result of this algorithm comprises of creation of excel or text document containing the words split from the document given and updating the sentence table with the word, identification of the word as concept or relation, generation of unique number with respect to concept and relation and sequence number indicating the order of arrangement for simpler computation.

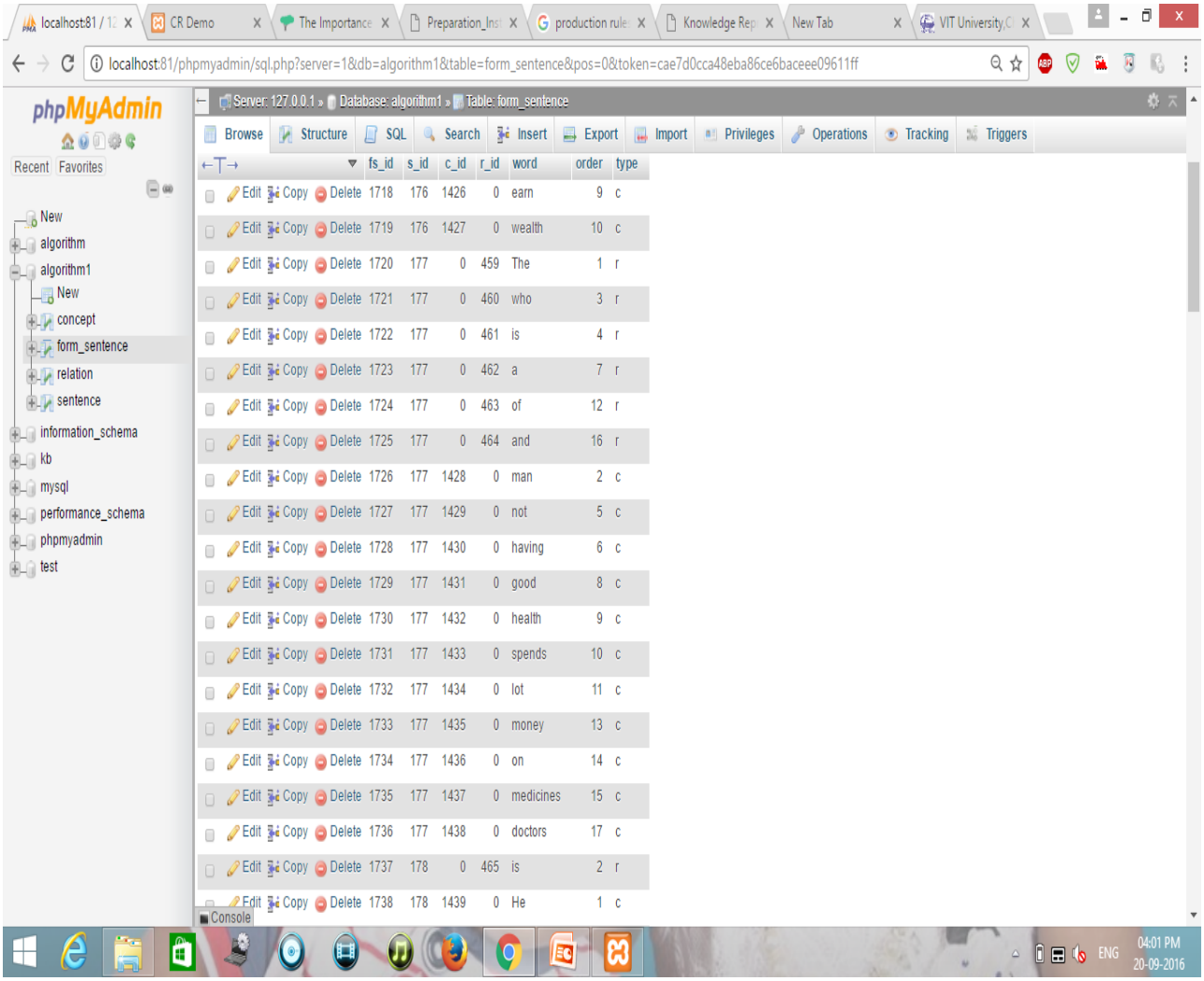

Figure 3: Parsed Content stored in Table

3.2. Algorithm for Adding Nodes

To solve the problems in knowledge representation an effective structure for symbolizing the data is required. A graphical structure is used to simplify this situation; hence Concept Relation (CR) graph is deployed.

Adding_Content(word1,..,wordn)

Input: Content (word1,.., wordn) to be included from .csv file

Output: Entire content added to the CR graph structure
/* Relation Table (RT) contains a set of relations */
/* Concept Table (CT) contains clauses, which are existing in the graph*/

Procedure:

```
DETERMINE the length of the content
INITIALIZE count=0
IF count < length of the content
   CHECK IF the word is in RT
      RETURN (relation)
      count++
   ELSE
      CHECK IF the word is present in CT
         RETURN (concept)
         count++
      ELSE
         CHECK IF CT ≠ ϕ
            FIND the position for the word to be added
            ADD the word to CT
            count++
         ELSE
            ADD the word to CT as position=1
            count++
END
```

As a pre requisite, two tables are created. One table known as Relation Table (RT) is filled with the relations. This RT is pre-defined and modifications cannot be made. The other table is Concept Table (CT), which keeps on appending concepts to the table as new undefined words are added.[8]

When the words captured are feed from the spreadsheet, the number of words is calculated. Checking the count to be lesser than the number of words obtained for the sheet, the procedure continues. Each and every word is taking separately and compared with the pre-defined verb list in the RT. If the word present, then that word is marked as a relation and its position is returned. Otherwise, that word is verified with the clauses which are already added into the CT. If the CT contains no word, then add this word as the first word. Moreover find out the most suitable location for adding the word in CT. Increase the count by one and repeat the same routine for the next word till all the words from the spreadsheet are placed in the correct position.

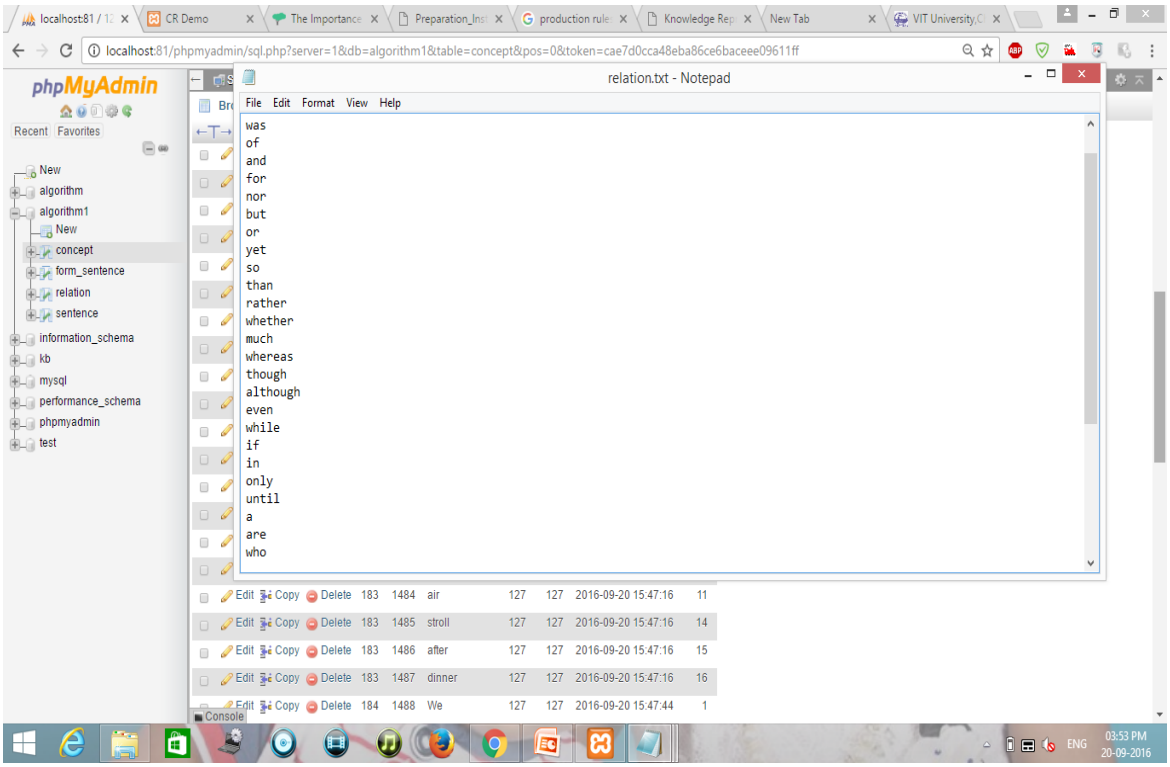

Figure 4: Pre-defined Relations

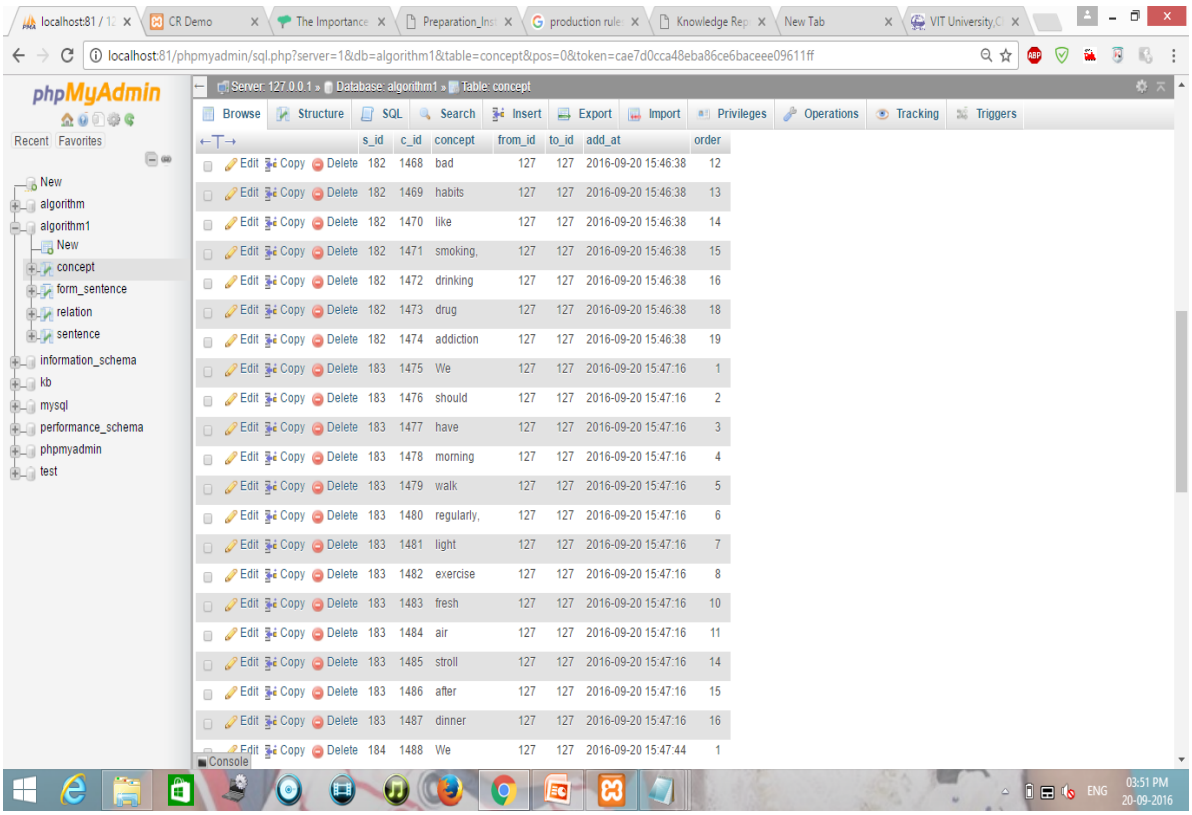

Figure 5: Concept Table (CT) updated with Concepts

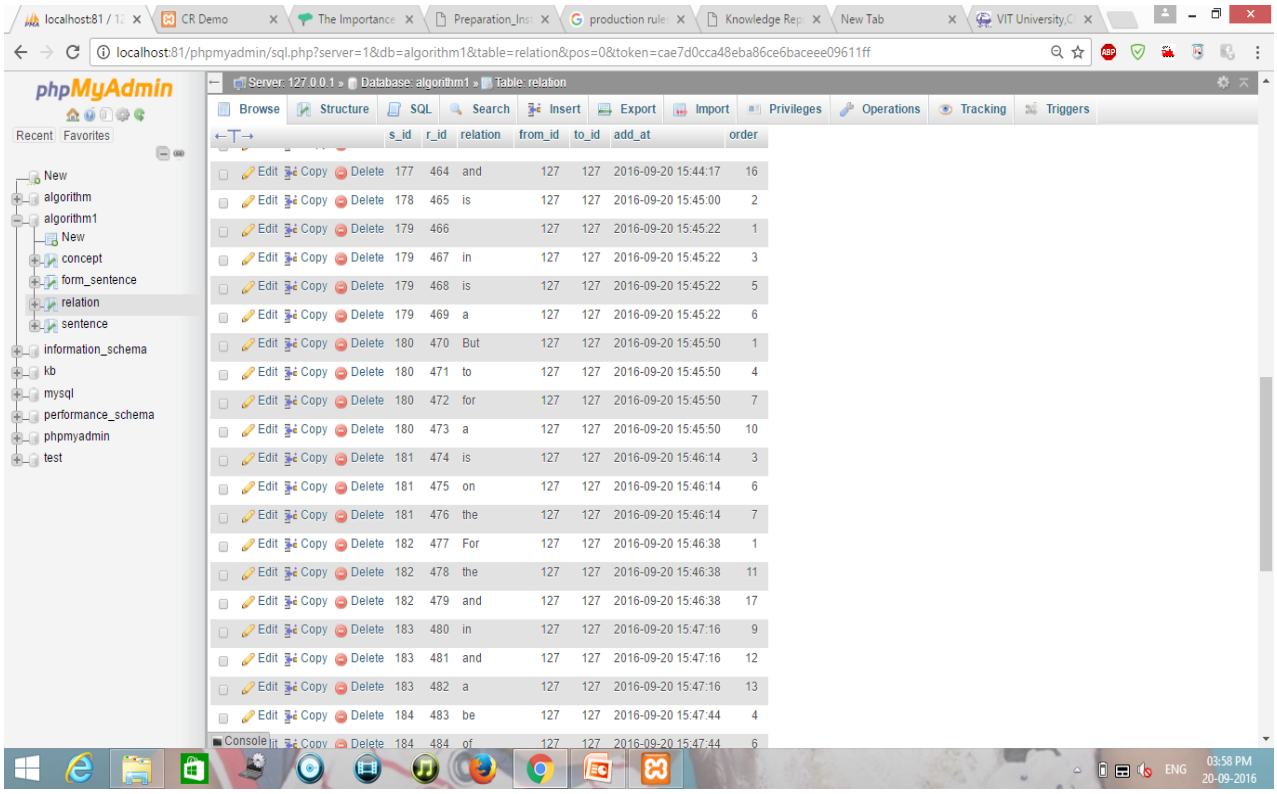

Figure 6: Relation Table (RT) with Relations

3.3. Algorithm for Modifying Nodes

The clauses in the graph can be changed or deleted. When such alternation takes place a proper replacement should happen. When a concept is modified, the relation existing between those nodes with the other nodes must also be carefully handled.

Modify_Concept(word,whatchange)

Input: Concept (C) to be modified

Output: Modified CR graph.

Procedure:

IF CT ≠ ϕ
  SEARCH IF Concept (C) ⊆ CT
    THEN the modification is done
    ALL the relation (*r*1,.., *rn*) nodes associated with C are marked
    CHANGES are made to C and (*r*1,..,*rn*)
  ELSE
    RETURN Concept (C) not found in CT
RETURN CT is Empty
SEARCH C ⊆ RT
    NO modification can be done
ELSE
    RETURN Concept (C) not found in RT

If any change has to be made to an entity then its availability in the CT is checked. If it is present then all the relation nodes emerging from this concept node are marked. Necessary changes like replacement or deletion can be performed to those marked nodes.

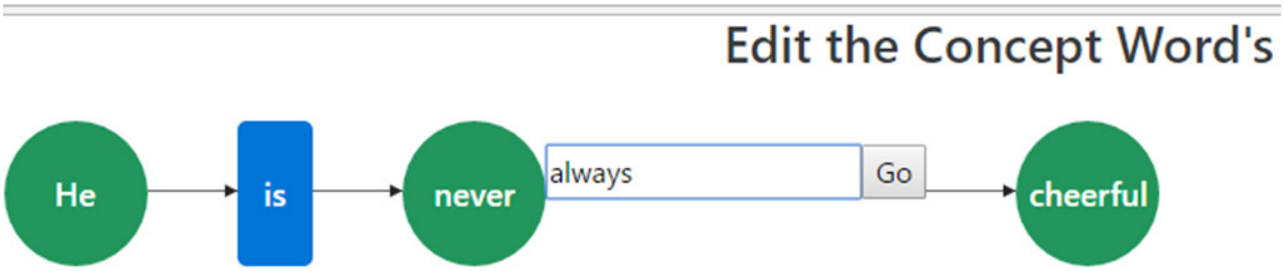

Figure 7: Concept for Modification

If replacement of concept has to be done, then the unique number has to be changed. The newly replaced concept exist the concept list, then the corresponding unique number has to be updated in the concept table and sentence table, otherwise new unique number has to be generated.

For deletion of concept, the concept name along with unique number and sequence number has to be removed from the sentence table. If the concept is in the beginning of the sentence formed, then the entire sentence is deleted from the sentence table and necessary updates like change of unique number, concept number are made in the concept table.

If the word is found in RT, then no modification can be done because this table is not created by appending data but by pre defining the verbs as relations.

3.4. Algorithm for Traversing Nodes

Reasoning[9] plays a greater role in the process of knowledge representation. Thus finding the occurrence of a concept and getting their relationship with the other concepts fetches the model.[6]

Traverse_Content(word)

Input: Concept (C) to be searched

Output: Displays the related information.

Procedure:

```
IF CT ≠ ϕ
  SEARCH IF Concept (C) ⊆ CT||RT
    THEN the related Nodes are fetched
    DISPLAY the concepts (C, C1, C2,..,Cn)&& their relation(r1,..,rn)
  ELSE
    RETURN Concept (C) not found
RETURN CT is Empty
```

This algorithm finds out whether the clause is present in CT or RT. Once the presence is confirmed then the clause along with the other associated neighbouring concepts and their relationship are taken out and displayed.[11] Or else the presence of the searched entity is denied.

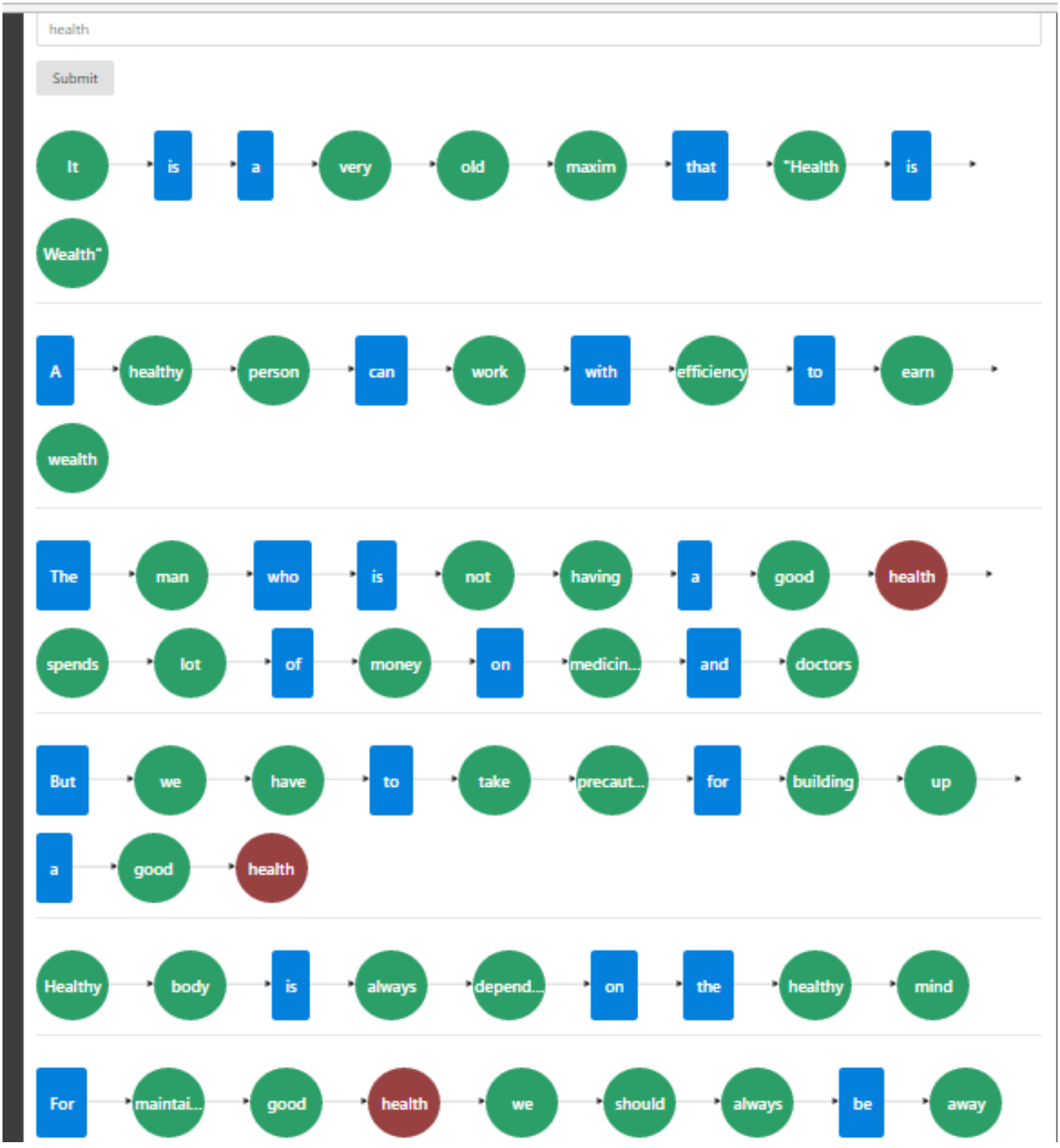

Figure 8: Search for Concept

The extracted concepts if linked with more than one concept through a relation, then the other concepts are also exhibited. Thus enabling the retrieval of all possible concepts connected with the selected concept.

4. CONCLUSION

In this paper, we have presented the working of our knowledge representation model with various algorithms for content parsing the document, adding, modifying and searching concepts. The objective of this paper is to fetch the content from a document which is semi structured, to place the content in a systematic way and to extract meaningful content from the knowledge base using Concept_Relation (CR) Graph.

References

John F Sowa, "Knowledge Representation: Logical, Philosophical and Computational Foundations", New York, PWS Publishing Co., 2000

S. Praveena Rachel Kamala, Dr. S. Justus, "Towards MORK: Model for Representing Knowledge", I. J. Modern Education and Computer Science, 2016, pp.45-53.

Frank van Harmelen Vladimir Lifschitz Bruce Porter, "Handbook of Knowledge Representation", 1st Edition, Volume 1, 2007.

Chuntao Jiang, Frans Coenen, Robert Sanderson, Michele Zito, "Text classification using graph mining-based feature extraction", Knowledge-Based Systems 23 (2010) 302–308.

Tiago A. Almeida, Tiago P. Silva, Igor Santos, , José M. Gómez Hidalgo, "Text normalization and semantic indexing to enhance Instant Messaging and SMS spam filtering", Knowledge-Based Systems 108 (2016) 25–32.

Misael Mongiovì, Diego Reforgiato Recupero, Aldo Gangemi, Valentina Presutti, Sergio Consoli, "Merging open knowledge extracted from text with MERGILO", Knowledge-Based Systems 108 (2016) 155–167.

Madalina Croitoru, Nir Oren, Simon Miles, Michael Luck, "Graphical norms via conceptual graphs",Knowledge-Based Systems 29 (2012) 31–43.

Julia Hoxha, Guoqian Jiang, Chunhua Weng, "Automated learning of domain taxonomies from text using background knowledge", Journal of Biomedical Informatics 63 (2016) 295–306.

Miao Fana, Qiang Zhou, Thomas Fang Zheng, Ralph Grishman, "Distributed representation learning for knowledge graphs with entity descriptions", Pattern Recognition Letters 000 (2016) 1–7.

Alberto Tonon, Michele Catasta, Roman Prokofyev, Gianluca Demartini,Karl Aberer, Philippe Cudré-Mauroux, "Contextualized ranking of entity types based on knowledge graphs", Web Semantics: Science, Services and Agents on the World Wide Web 37–38 (2016) 170–183.

Antonio Jimeno Yepes, Rafael Berlanga,"Knowledge based word-concept model estimation and refinement for biomedical text mining", Journal of Biomedical Informatics 53 (2015) 300–307.