

Towards MORK: Model for Representing Knowledge

S. Praveena Rachel Kamala

School of Computer Science and Engineering, VIT University, Chennai, 600 127, India.
E-mail: praveena.rachel2013@vit.ac.in

Dr. S. Justus

School of Computer Science and Engineering, VIT University, Chennai, 600 127, India.
E-mail: justus.s@vit.ac.in

Abstract—Smart world needs intelligent system for effective and timely decision making. This is achieved only through a knowledge based system with functional knowledge representation units. In this paper, two models are proposed for representing knowledge. This process involves in getting the data and placing the information in the correct location. Logical notations are used for taking the clauses and graph is used for putting the entities. In Model one, the data is translated into logical statements using predicate logics, later the knowledge is stored in conceptual graph and retrieved. Whereas in Model two, the given information is translated using First Order Logic (FOL), by applying description logic concept rules are defined and as a result reasoning is done. Storage is done by using concept-relation graph. The main aims of our models are to have easy and simple access over the information. These models return the required exact answer, for the higher order query posted by the end user to the intelligent system.

Index Terms—Knowledge Representation, Knowledge Retrieval, Conceptual Graph, Description Logics, Predicate Logic.

I. INTRODUCTION

Knowing and understanding an individual or being well aware and familiar about a system, and being familiar with their properties, data, information, particulars, details, task and so on are derived through practice and intelligence. This knowing through gaining and reasoning facts is termed as Knowledge. This can be achieved by a practical talent (tacit) or theoretical fluency (explicit), proper (formal) or methodical (systematic), personal or procedural or propositional way.

Knowledge plays a vital role in all sectors – from education to healthcare to aeronautical to financial and e-commerce systems. To design a successful knowledge based system (KBS) understanding how, where and when the knowledge flows is essential.

Knowledge is defined as a “Known True Belief or Fact” of a System [1]. If a system believes some concept to be true, then that becomes knowledge to that system. For example: if I believe “Sky is blue”, and it is a true

fact. Now this is knowledge to me. Consider another example: “The table has four legs” – a known true fact – and is a knowledge. This is form of knowledge is called “Tacit Knowledge” – knowledge which cannot be expressed, and is in the human brain. Our objective is to convert this tacit form of knowledge to an explicit form of knowledge – that which takes an expressible form – so that any system, either human or any intelligent or expert systems would understand to process it. Hence converting a tacit knowledge into an explicit knowledge is called Knowledge Representation. Our work focuses on designing knowledge representation models for an Expert System.

Our proposed two models for representing knowledge are based on conceptual graphs. Here the information is represented as concept-relation graph, making confirmation and interpretation easier. These models enable the concepts to effortlessly incorporate into the graphical structure.

This paper covers the Related Works, Motivation and Problem Statement in section 2, Processes in Knowledge Based Systems in section 3, Knowledge Representation Models in section 4, and our planned Concept Based Models for Knowledge Representation in section 5, Conclusion and Future Works in section 6.

II. RELATED WORKS

A. Existing Knowledge Representation Models

In the recent years, there are a number of researches taking place in the area of knowledge representation. The various technologies followed for representing knowledge are fuzzy Petri nets [16],[17], hybrid representation, production rules, hierarchical representation, neural networks, semantic networks [3], logical representation, ontology[10], and knowledge webs.

Fuzzy Petri nets model proposed by X. Li and F. Lara Rosano [13], for dynamic knowledge representation and inferences. This model aimed for more generalized expert system trained using neural networks. A novel hybrid intelligent system (HIS) proposed by Phayung Meesad and Gary G. Yen [14], provides a unified integration of numerical and linguistic knowledge representation. A

hierarchical integration of an incremental learning fuzzy neural network and a linguistic model (i.e.) fuzzy expert system are deployed and genetic algorithm is used for optimization.

Tsang E.C.C et al, proposed a model based on rules, the “If..Then” condition is widely used for manufacturing and scheduling knowledge representation. Victor R L Shen also presented a representation model based on fuzzy production rules where high level fuzzy Petri nets are used. Hong Sen Yan proposed a new complicated knowledge representation approach based on knowledge mesh and agent mesh [15]. Knowledge Management based approach focus on macro knowledge, structure, functions and information flow of an advanced manufacturing system.

K. Petridis et al., proposed knowledge representation and semantic annotations of multimedia content, where Ontologies are being extended and enriched to include low-level audiovisual features and descriptors. Piero P. Bonissone, Thomas R. Kiehl proposed knowledge representation model based on Evolutionary algorithm, using genetic algorithm and fuzzy logic. Jianlui Luo et al., presented a graph based representation model [4][8].

Frame based knowledge representation was proposed by Jose ´ Manuel Go ´mez-Pe ´rez [18], [19]. Phivos Mylonas proposed semantic representation for knowledge. Christopher Brewster proposed representation model based on Ontology. These are a few works related to knowledge representation.

B. Motivation & Problem Statement

Several works on Knowledge Representation has successfully proved their credibility in the functional platform. However, as far as we know, there is not a complete work on proposing knowledge as a model. A knowledge representation model should be ambient intelligent, detecting conflict, and resolving conflict.

III. PROCESSES IN KNOWLEDGE BASED SYSTEMS

In this section we’ll present some of the essential processes that governs any Knowledge based Systems (KBS).

A. Knowledge Representation

Knowledge representation is all about identifying concepts (clauses or entities) in an unstructured repository or in a tacit knowledgebase, and assigning relationships to connect these clauses. In natural language, people represent and store their ideas about objects by mapping using symbols, signs, numbers and characters. Whereas, for a computer based system whether it is a simple fact or complex relationship, mathematical formulas and rules are used for representing the natural syntax. An expressible form of ideas and facts about the real-world environment has to be represented, which is called explicit knowledge.

These programs (agents) are used to associate related concepts and exhibit inheritance hierarchy. These agents may use the information to derive that which is implied

by it, to communicate with people in natural languages, to decide what to do subsequently, to plan future activities and to solve problems in area that usually require human knowledge [1].

B. Knowledge Reasoning

Knowledge reasoning is obtaining new knowledge from knowledgebase which is already present, or inferring to a conclusion from the set of already collected knowledge based on the facts and rules of the system. Knowledge representation is worthless without the ability to reason with them. An efficient knowledge representation scheme should be easy to use, easily modified and extended. The different models for knowledge representation are Procedural method, Declarative method, Relational method, hierarchical method and complex network graph.

C. Knowledge Storage & Retrieval

This gives knowledge a more structural form [2], which means it involves structured storage of knowledge units in the knowledgebase, effective and accurate retrieval of knowledgeunits for the high-order query posted by the user. The knowledge retrieval process aims in arriving at best set of knowledge unit. This activity is performed in the region where represented knowledge is stored. Once meaningful information with proper reasoning is obtained, this information can be shared.

Knowledge sharing is exchanging information with others. This enhances the overall performs by transferring the assets of one entity to another entity, reducing the redundancy, reworking and enables smooth knowledge flow. The Figure 1 gives the relation between knowledge representation, storage, retrieval and sharing in knowledge based system.

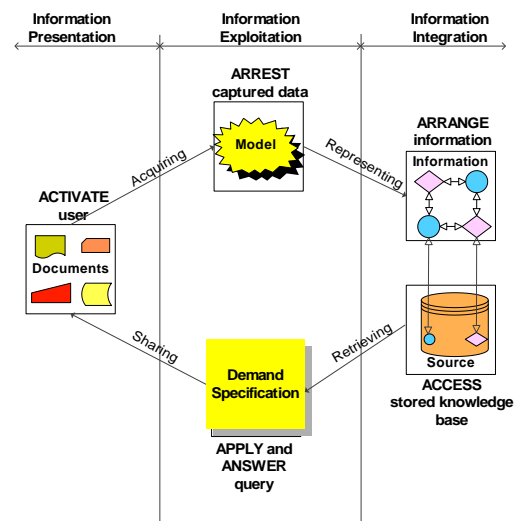


Fig.1. Big Picture of Knowledge Process.

The knowledge based system has the following stages:

- (1) Activate Stage: In this stage, knowledge base is created and initialized,

- (2) Arrest Stage: The data is captured from the source,
- (3) Arrange Stage: Information is organized,
- (4) Access Stage: Storing and fetching details, and
- (5) Apply (Answer) Stage: Using the exact inference for sharing.

IV. KNOWLEDGE REPRESENTATION MODELS

In this paper, the main focus is on proposing models for knowledge representation. Knowledge representation seeks to design real world activities into a machine interpretable approach, which is similar to human way of thinking. The representation models have components such as objects, events, relationships and the domain covers all related data about any real world or system details [9]. A knowledge based system maintains a knowledge base which has a collection of statements about the domain, and manipulates information with suitable reasoning [12].

The knowledge representation can be done in many different ways, they are Concept based knowledge representation, Logic based knowledge representation and Rule based knowledge representation.

A. Logic Based Knowledge Representation

In the olden days, Syllogism was the only way for knowledge representation and reasoning. In this method the logical conclusion is derived from one or more propositions that are exactly equal or assumed to be true based on logical analysis. Posterior and Prior analysis are the two types of logical analysis.

The example given explains Posterior analysis, “If all brave animals are majestic, and all lions are brave animals, then all lions are majestic”. This example is categorized into three parts. The first part is “If all brave animals are majestic” representing the major premise, the second part is “all lions are brave animals” indicating minor premise and final part is “then all lions are majestic” signifying the conclusion.

In the case of Prior analysis, there are two types namely non – negative and negative analysis. “Cat is a mammal, mammals have skin, so Cats have skin”, is an example for non – negative Prior analysis. This is of the type A implies B, B implies C, therefore A implies C. “Cars are not trucks, trucks are huge, hence cars are not huge”, this is grouped under negative Prior analysis meaning A implies negation of B, B implies C, so A implies negation of C.

1. Predicate Calculus:

These classification yields to a more complex situation, therefore rules of inference (\forall) overcame the problem. The various forms are universal affirmation (all P is Q), particular affirmation (some P is Q), universal negation (no P is Q) and particular negation (some P is not Q) [5].

Later the concept of Boolean algebra came into existence. Boolean takes only two values and the result generated will also be only of the kind with those two values, either 0 (False) or 1 (True). Consider the example, X – “Sunday is a holiday” and Y – “there is school”.

The different interpretations for Boolean algebra are:

- $X \times Y$ – Sunday is a holiday and there is school.
- $X + Y$ – Sunday is a holiday or there is school.
- $\neg X$ – Sunday is not a holiday.

In the case (1), ‘and’ operation is performed resulting only $1 \times 1 = 1$ (True), whereas $0 \times 1 = 1 \times 0 = 0 \times 0 = 0$ (False). In case (2), ‘or’ operation is achieved producing $0 + 0 = 0$ (False) and $1 + 1 = 0 + 1 = 1 + 0 = 1$ (True). Finally in case (3), ‘not’ operation is carried out yielding $\neg 1 = 0$ and $\neg 0 = 1$.

Boolean algebra is applied in many fields like digital logic circuits, Boolean operations, Boolean search, etc... This can be represented using truth table.

2. First Order Predicate Logic:

“Tom is a cat”, “Whale is a mammal” are facts. In propositional logic, these facts can only be symbolized by P and Q respectively but no proper conclusions can be derived though they sound similar. These facts can be represented as CAT (Tom) and MAMMAL (Whale). Hence it is very difficult to represent a simple sentence using propositional logic [5].

This drawback is overcome by Predicate logic (PL), which is a logical extension of propositional logic. The predicate calculus contains three logical notations namely term, predicate and quantifier. Term refers to the constants and functions like

$$\begin{aligned} \text{LESSER}(X, Y) &= \text{TRUE}, \text{ if } X \leq Y \\ &= \text{FALSE}, \text{ Otherwise} \\ \text{LESSER}(\text{height}(s), \text{height}(t)) \end{aligned}$$

In this ‘LESSER’ is the predicate and ‘height’ is the function and ‘x, y, s, t’ are constants.

Predicate links the term to truth table, for example $\text{OWNS}(\text{Elizabeth}, \text{Parrot})$ and Variables are combined using quantifiers. There are two types of quantifiers, they are “there exist (\exists)” and “for all (\forall)”. Suppose, there are two variable a, b denoting a person and an object respectively, and let $O(a, b)$ mean “a owns b”. These are a few translations using quantifiers in predicate logics.

$\forall a O(a, \text{parrot})$ – Everybody owns parrot.

$\forall a \exists b O(a, b)$ – Everybody owns something.

$\exists b \forall a O(a, b)$ – There is something that everybody owns.

$\exists b \neg O(\text{Elizabeth}, b)$ – There is something that Elizabeth doesn’t own.

$\exists b \forall a \neg O(a, b)$ – There is something that no one owns.

The First Order Logic (FOL) is a well formed language based on the rules used in predicate logics. The FOL like the PL contains quantifiers and in addition has logical connectors like conjunction (\wedge), disjunction (\vee),

implication $\langle \rightarrow / \supset \rangle$, bi-conditional $\langle \leftrightarrow / \equiv \rangle$, and negation $\langle \neg / \sim \rangle$.

3. Description Logic:

Description logic is used in the field of artificial intelligence for formalized knowledge representation and reasoning. This is based on the idea of semantic networks and frame language. The real world situation is represented in the form of a network where nodes describe concepts and links characterize the relationship between concepts [6], [7].

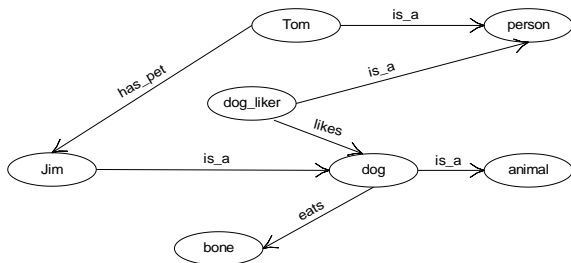


Fig.2. Illustration for Description Logic.

The semantic network structure in Figure 2 expresses the following cases. There are two types of assertions about an individual; they are concept and role assertions. The statements about a concept can be classified as basic concept definition, relationship among concepts and using axioms.

The concept assertion is expressed as

- $person(Tom)$ – “Tom is a person” and
- $dog(Jim)$ – “Jim is a dog”.

An example for role assertion is

- $has_pet(Tom, Jim)$ – “Tom has a pet which is Jim”.

A concept can be basically defined as

- $dog_liker \equiv person \wedge \exists likes.dog$ – “A dog liker is a person and there exists a dog that the person likes”.

Relationship between concepts can be given as

- $likes(dog_liker, dog)$ – “(Every) dog liker likes a dog”.
- $eats(dog, bone)$ – “(Every) dog eats bone”.

Axiom is a formula for deriving the starting point of reasoning or a statement which is the truth without controversy such as

- $dog \subseteq animal$ - “dog is an animal”. (hierarchy of concepts)
- $dog \subseteq animal \wedge \forall eats.bone$ - “a dog is an animal which eats bone”.

The description logic based knowledge representation system [11] is made up of a knowledgebase containing

TBox (terminology) and ABox (assertion about individuals) as in Figure 3.

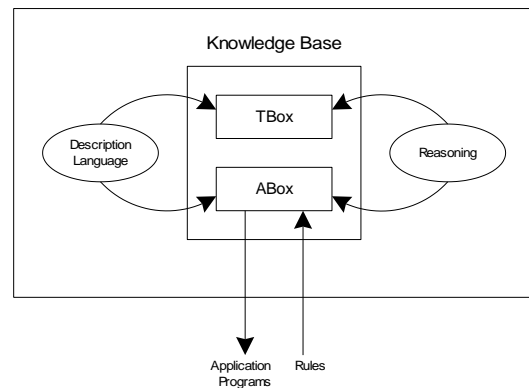


Fig.3. Description Logic based Knowledge Representation System.

B. Rule Based Knowledge Representation

Inference is a way by which knowledge is obtained from existing knowledge. Inference engine is a widely used tool for deriving conclusions from the facts by expert systems. These facts and known conclusions are stored in the knowledge base. Forward and backward chaining are two modes of action in inference engine. In forward chaining the facts are studied and combined to arrive at the conclusion, whereas in backward chaining the goal is backtracked with the available facts to attain the conclusion.

In Figure 4, an example for taxonomy is illustrated. In the case of forward chaining, the goal is achieved by inferring each and every step. One should be above 18 years of age, then male, married and should have at least a child to be a father, which is the goal. In backward chaining to be a father (goal), one will have to have a child, be married, be male and should also be above 18 years old.

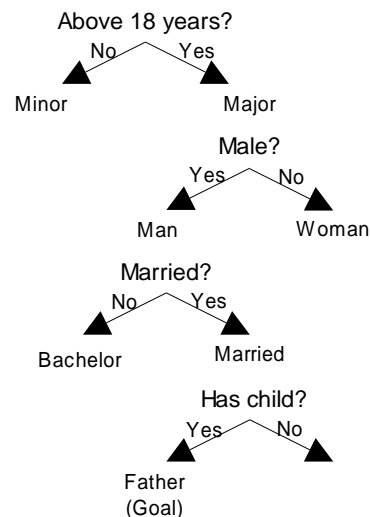


Fig.4. Example for Taxonomy.

Rule based representation requires less informational power than logic based representation. Rule based system will not contain rules of inference and all the quantifiers, but only focus on the task to be completed. The major strength of rule based representation is the ease to work on system based on human psychology. Another advantage is the use of human understanding and skill to naturally describe in terms of rules. The drawback is that rule based system is not easy to formalize in machines.

1. IF THEN:

Representing knowledge is most popularly done by using rules. Rules with IF – THEN statements are used, to express different types of complex statements. In such statements the value followed by IF is called as the body or antecedent and the value followed by THEN is called as the head or consequence. These are the different types of IF – THEN statements.

- (a) IF condition THEN action: (e.g.) IF we switch off the light THEN the light will stop burning.
- (b) IF premise THEN conclusion: (e.g.) IF queen lives in London and Elizabeth is a queen THEN Elizabeth lives in London.
- (c) IF proposition p1 and proposition p2 are true THEN proposition p3 is true: (e.g.) IF it is raining and we take umbrella THEN we will not get wet.

This example demonstrates how knowledge is expressed as rule. The statement is “You want to arrange a treat during the weekend and you don’t want to wait in the queue”. By applying rule to the statement, “if you reserve seat then you don’t wait in the queue”. The conclusion for the query, “if the seats are unavailable on Saturday and are available on Sunday?” is “You arrange treat on Sunday”.

Knowledge is interpreted as rules. Rules are in black and white language with only two options. It is impossible to produce any intelligent system by using only rules because in order to make the computer machine to arrive with a valid conclusion, one has to program with all the possible rules. The range of all possible rules will be from 100 to 1000, but the need will be more than 100000. Therefore, formulate more rules for every problem to achieve proper and correct conclusion.

V. CONCEPT BASED KNOWLEDGE REPRESENTATION

These two newly proposed models for knowledge representation aim at simple addition and retrieval of information. Thus making the system to quickly append the obtained data and swiftly release the content demanded by the user. To satisfy these criteria, a graphical structure is used. For clear description and definition of the content acquired, logical interpretation is deployed.

Logical interpreters like Predicate logic and First Order Logic (FOL) are used for describing the scenario. Rule definer like Description logics and data manipulator like

graphs are jointly engaged in our models for effective performance. Combining these methods will improve the clarity in concept and their relation with the other already existing concepts.

Why we are choosing CG as a representation model for KU?

The concepts defined in the conceptual graph are rich supply of knowledge with special structure and content. Conceptual graphs are knowledge based representation structures grounded on philosophy, linguistic and object oriented principles [9].

Conceptual Graph

The conceptual graph provides a spontaneous and easy to understand method for representing knowledge. A graphical technology to represent knowledge expressed using a formal language along with the meaning of the concept developed by John Sowa, based on the work of C. S. Peirce is known as the Conceptual Graph [9].

The conceptual graph contains rectangle box to represent concept, circles for relations and arc from concept to relation or relation to concept. There can be no arc from concept to concept and relation to relation.

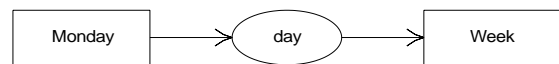


Fig.5. Simple Conceptual Graph.

The above Figure 5 is an example for conceptual graph stating “Monday is a Day of Week” or “Week has a Day which is Monday”.

A. Model-1: Predicate Logic and Conceptual Graph

To represent a given fact in the knowledgebase, the model-1 converts the data into a graph. The statement is taken as the input and interpreted into logical statement using predicate logic. Later these logical statements are translated into graphs by conceptual graphs. The following example exhibits the working of this model-1.

Input Statement:

Let us consider the statement “The sky is clear and blue, so it won’t rain”. In this situation, the input given is “the sky is clear and blue” and the required output after verification and reasoning is “so it won’t rain”.

Step 1:

The statement is taken and converted into valid logical phrase using Predicate logic.

1. Sun, Moon, Stars and Clouds are in Sky.

$$\forall x[Sun(x) \rightarrow Sky(x)]$$

$$\forall x[Moon(x) \rightarrow Sky(x)]$$

$$\forall x[Star(x) \rightarrow Sky(x)]$$

$$\forall x[Cloud(x) \rightarrow Sky(x)]$$

2. Sun is in the Day and Moon and Stars are in the Night.

$$\forall x[Sun(x) \wedge Dayhas(x)]$$

$$\forall x[Sun(x) \wedge has(Day, x)]$$

$$\forall x, y[Moon(x) \wedge Star(y)]$$

$$\wedge Nighthas(x, y)$$

$$\forall x, y[(Moon(x) \wedge Star(y))$$

$$\wedge has(Night, (x, y))]$$

3. Day is bright and Night is dark.

$$\forall x[Sun(x) \wedge isBright(x) \rightarrow Day(x)]$$

$$\forall x[\neg Sun(x) \wedge isDark(x) \rightarrow Night(x)]$$

4. If Clouds are present, then it will rain.

$$\forall x[Cloud(x) \wedge Rain(x) \rightarrow$$

$$raining(x)]$$

$$\forall x[Cloud(x) \rightarrow Rain(x)]$$

5. If there are no clouds, then it will not rain.

$$\forall x[\neg Cloud(x) \rightarrow \neg Rain(x)]$$

$$\forall x[((\neg Cloud(x) \wedge \neg Rain(x)) \rightarrow$$

$$Sun(x)]$$

$$\forall x[((\neg Cloud(x) \wedge \neg Rain(x)) \rightarrow$$

$$\neg raining(x)]$$

6. If the Sun is in the Sky and the Sky is blue, then there are no clouds and it is a bright day.

$$\forall x[(Sky(x) \wedge Sun(x) \wedge is(x, blue) \wedge$$

$$\neg Cloud(x)) \rightarrow Day(x) \wedge isbright(x)]$$

$$\forall x[((Sky(x) \wedge Sun(x) \wedge is(x, blue)) \wedge$$

$$(\neg Cloud(x) \rightarrow \neg Rain(x))) \rightarrow$$

$$Day(x) \wedge \neg Rain(x)]$$

7. If the Sky is dark with no Clouds then it is a night with no rain.

$$\forall x[(Sky(x) \wedge is(x, dark) \wedge \neg Cloud(x))$$

$$\rightarrow (Night(x) \wedge \neg Rain(x))]$$

8. If raining then it is not sunny.

$$\forall x[raining(x) \rightarrow \neg Sun(x)]$$

9. If the Sky is clear, then it will not rain.

$$\forall x[(Sky(x) \wedge isClear(x)) \rightarrow \neg Rain(x)]$$

Step 2:

The generated logical expressions are represented as a formal Concept-Relation (CR) graph based on the ideas of conceptual graph as in Figure 6.

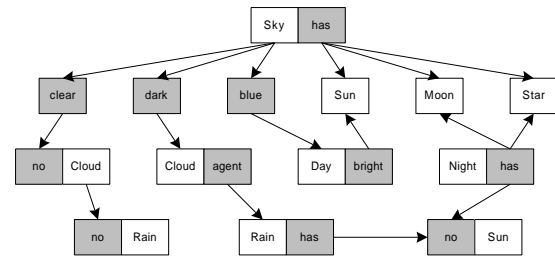


Fig.6. Concept-Relation Graph for Sky and Rain Example.

The Concept-Relation (CR) graph is extended from the work of Sowa. Each node contains both the concept along with their relationship to others nodes. If the mapping is done with a multiple concepts or relations then the node contains only a single concept or relation respectively. The nodes containing the concepts have clear background whereas the nodes or part of the nodes comprising the relation have shaded background. The edges are represented using arrows. These arrows are pointed either from concept to relation or relation to concept. The arrows cannot be pointed from concept to concept or relation to relation.

Step 3:

Based on the constructed conceptual graph, the process of reasoning can take place. For instance from the input given “the sky is blue and clear”, the following observations are made:

1. The Sky is blue when the Day is bright with the Sun.
2. The Sky is clear when there is no Cloud, where ‘no Cloud’ is an agent of ‘no Rain’.

From the above inference, the remarks are it’s a Sunny day with no Clouds. Hence, the conclusion is “it will not rain”.

Note for Model-1:

The data is processed by using predicate logic and constructs a conceptual graph. The predicate logic gives a methodical procedure for problem solving. The logical statements are concise, unambiguous, context insensitive, expressive, efficient and effective for inferences. Predicate logic by itself is extremely formal kind of

representation mechanism. The conceptual graph enables mapping properties into nested contexts. Later from the conceptual graph necessary and adequate information is retrieved. Thus the graphical structure gives an easy approach for representation.

B. Model-2: Description Logic and Conceptual Graph

In this Model-2, the knowledge is represented by converting the statement into description logic using first order logic. Rules are formulated and proper reasoning is done by applying conceptual graph structure. Thus required output is derived from the graph.

Input Statement:

Consider this statement: “If no man is reading the book, then is there someone who wrote the book reads it?”

Step 1:

The given statement is converted into logical statements using first order logics (FOL).

1. FOL for the given statement “If no man is reading the book, then is there someone who wrote the book reads it?”

$$\forall x[Man(x) \wedge \neg Read(x, book)] \supset \exists x[Author(x, book) \wedge Read(x, book)] \quad (1)$$

2. No man reads the book.

$$\forall x[\neg Man(x) \wedge Read(x, book)] \quad (2)$$

3. Adult is a group containing men and women. (if men are not reading then women are reading)

$$\forall x[Adult(x) \supset Man(x) \wedge Woman(x)] \quad (3)$$

4. Woman reads the book.

$$\forall x[Adult(x) \supset Woman(x) \wedge Read(x, book)] \quad (4)$$

5. Assuming negation approach: The author does not read the book.

$$\forall x, y[Author(x, y) \supset \neg Read(y, x)] \quad (5)$$

6. The author of the book is not Jane. (y=Jane)

$$\forall y[Woman(y) \wedge y \neq Jane \supset Author(book, y)] \quad (6)$$

7. A woman, who is not Jane, does not read the book. (from 5)

$$\forall y[Woman(y) \wedge y \neq Jane \supset \neg Read(y, book)] \quad (7)$$

Conclusion:

Jane is the author.

Step 2:

General properties for different concepts can be given by using restrictions and distinguish primitive definitions. These are a few definitions;

Primitive Concepts:

- Adult \sqsubseteq T
- Woman \sqsubseteq adult
- Man \sqsubseteq adult
- book \sqsubseteq \perp

Defined Concepts:

1. [ALL: Man [FILLS: hasn't Read]] From (2)
2. [ALL: Adult [AND Man, Woman]] From (3)
3. [AND Woman [FILLS: has Read]] From (4)
4. [AND Author [FILLS: hasn't Read]] From (5)(negation approach)
5. [ALL: Woman [EXISTS 1: Woman (Jane \rightarrow [FILLS: can't_be Author]) [FILLS: can_be Author]]] From (6)
6. [ALL: Woman [EXISTS 1: Woman (Jane \rightarrow [FILLS: can't_be Author]) [FILLS: hasn't Read]]] From (7)
7. [ALL: Woman [EXISTS 1: Woman [ALL: except Jane] [FILLS: can't_be Author]]] From (5) and (6)
8. [AND Woman [FILLS: Jane Author]] From (7) (conclusion).

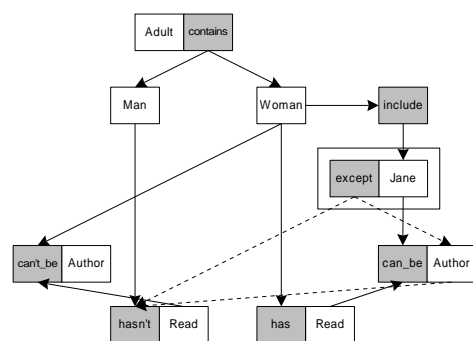


Fig.7. Concept-Relation Graph for Author – Reader Example.

Step 3:

Using the theory of conceptual graphs, the information is mapped using the concept – relation graph. Consider

the Figure 7, which gives the conceptual structure for the above mentioned statement.

In this CR graph, the assumptions are indicated by using a double border node as in case of Jane. Since the assumed concept along with the relation signifies a negative approach, they are indicated by a double bordered node.

Step 4:

The concept along with their relationship is represented as a single node. Mapping between nodes enable the graph to reason the sequence of action.

From the above CR graph, the problem is defined as Adult contains both Men and Women. Since no man reads the book, only a woman can be the Author of the book. By introducing a constant Jane under the category Woman and by applying negation approach to the situation, the conclusion is “Jane is the author of the book”.

Note for Model-2:

In this Model II, the information is converted into logical statements by using FOL and rules are based on description logic, adding restriction from the primitive experience. These facts are represented in CR graphical structure and reasoned knowledge is obtained. CR graph enables easy method of knowledge representation and retrieval.

Construct of a CG

Conceptual graph develops a straightforward and easy-to-implement methodology for transforming a structured concept definition into the corresponding representation breaking many limitations and problems in the extraction of semantic information. Defining the concepts for the situation, facilitate the implementation of a really complex environment to work easily.

Furthermore, conceptual graphs’ ability to identify and formalize similarities and differences between the defined concepts, enable to design a more efficient knowledge unit suitable for the environment. Conceptual graph ensures an implementation method for the semantic integration of all the available resources.

VI. CONCLUSION

The main contribution of this paper is to present two models for knowledge representation. The proposed two representation models are based on FOL, PL, DL and conceptual graph for knowledge representations. A list of theories related to knowledge representation and the importance of knowledge representation in the knowledge management system have been discussed.

These proposed models use predicate logic and description logic to translate the information into a valid logical statement. Checking the syntax and semantics at this stage, enables a meaningful representation of the knowledge.

Conceptual graphs defined by a formal mapping along with a variety of formal and informal extensions enable

mapping to and from other versions of logics [12]. The concepts along with their relationship can be clearly visualized. The interpretation of the conclusion is also easier due to the graphical structure. The meaning of the conclusion can be evidently and effortlessly derived.

Future Works:

In this paper we have opted to present framework for representing knowledge units, and this is the first step. An empirical or mathematical validation of these representational models of this paper, need to be done as the next step. Using these knowledge units in the form of conceptual graph, we need to work on the knowledge reasoning part and knowledge storage and retrieval part. These two works will be two separate research works which will complete the entire representation framework.

REFERENCES

- [1] Russell S and Norvig P, “Artificial Intelligence: A Modern Approach”, Prentice Hall, Third Edition, 2010.
- [2] Yiyu Yao, YI Zeng, Ning Zhong and Xiangji Huang, “Knowledge Retrieval (KR)”, IEEE Computer Society, Silicon Valley, USA, 2007, pp 729 - 735.
- [3] Stephan Grimm, Pascal Hitzler and Andreas Abecker, “Knowledge Representation and Ontologies: Logic, Ontologies and Semantic Web Languages”, Semantic Web Services, 2007, pp 37 – 87.
- [4] Jianhui Luo, Haiying Tu, Krishna Pattipati, Liu Qiao and Shunsuke Chigusa, “Diagnosis Knowledge Representation and Inference: Understanding fundamental ideas behind graphical models”, IEEE Instrumentation and Measurement Magazine, 2006, pp 45 - 52.
- [5] David Makinson, “Sets, Logic and Maths for Computing”, Springer, Undergraduate topics in Computer Science, 2008.
- [6] Markus Krötzsch, František Simančík and Ian Horrocks, “Representation and Reasoning: Description Logic”, IEEE Intelligent Systems 2014, pp 12 -19.
- [7] Franz Baader, Ian Horrocks and Ulrike Sattler, “Description Logics”, Elsevier, Chapter 3, Handbook of Knowledge Representation, 2007, pp 135 - 180.
- [8] Michel Chein and Marie – Laure Mugnier, “Graph – Based Knowledge Representation: Computational Foundations of Conceptual Graphs”, Advanced Information and Knowledge Processing, Springer, 2009.
- [9] John F Sowa, “Knowledge Representation: Logical, Philosophical and Computational Foundations”, New York, PWS Publishing Co. , 2000.
- [10] Grega Jakes, Veljko Milutinović, Sanida Omerović and Sašo Tomažič, “Concepts, Ontologies, and Knowledge Representation”, SpringerBriefs in Computer Science, Springer New York, 2013.
- [11] Sebastian Rudolph, “Foundations of Description Logics”, Karlsruhe Institute of Technology, Germany, 2011.
- [12] Van Harmelen, V Lifschitz, and B Porter, “Handbook of Knowledge Representation”, Elsevier, chapter 5, Conceptual graph by John F Sowa, 2008, pp 213 – 237.
- [13] X. Li and F. Lara Rosano, “Adaptive fuzzy Petri nets for dynamic knowledge representation and inferences”, Elsevier, Expert Systems with applications, Volume 19, Issue 3, 2000, pp 235 - 241.
- [14] Phayung Meesad and Gary G. Yen, “Combined Numerical and Linguistic Knowledge Representation and Its Application to Medical Diagnosis”, IEEE transactions on systems, man, and cybernetics—Part A: systems and

- humans, Vol. 33, No. 2, March 2003, pp 206 - 222.
- [15] Hong-Sen Yan, "A New Complicated-Knowledge Representation Approach Based on Knowledge Meshes", IEEE transactions on knowledge and data engineering, vol. 18, no. 1, January 2006, pp 47 - 62.
- [16] Hu-Chen Liu et al., "Dynamic Adaptive Fuzzy Petri Nets for Knowledge Representation and Reasoning", IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 43, No. 6, November 2013, pp 1399 - 1410.
- [17] Hu-Chen Liu, Long Liu, Qing-Lian Lin, and Nan Liu, "Knowledge Acquisition and Representation Using Fuzzy Evidential Reasoning and Dynamic Adaptive Fuzzy Petri Nets", IEEE Transactions On Cybernetics, Vol. 43, No. 3, June 2013, pp 1059 - 1072.
- [18] Jose ´ Manuel Go ´mez-Pe ´rez, Michael Erdmann, Mark Greaves, and Oscar Corcho, "A Formalism and Method for Representing and Reasoning with Process Models Authored by Subject Matter Experts", IEEE Transactions On Knowledge And Data Engineering, Vol. 25, No. 9, September 2013, pp 1933 - 1945.
- [19] Milko Marinov et al., "Using frames for knowledge representation in a CORBA-based distributed environment", Knowledge-Based Systems, Elsevier Science Publishers B. V. Amsterdam, The Netherlands , Volume 21 Issue 5, July, 2008, Pages 391-397.

Authors' Profiles



S. Praveena Rachel Kamala is a full time Ph.D. Research Scholar at the School of Computer Science and Engineering, VIT University, Chennai, TN, India. Her areas of research include Knowledge Engineering, Knowledge Representation, Knowledge Storage and Retrieval.

She has received her undergraduate and postgraduate degrees in Computer Science & Engineering from Madras University and Anna University, Chennai. She has also teaching experience in handling post-graduate computing science courses.



Dr. S. Justus received his doctorate degree from Madurai Kamaraj University, Madurai, India. His research specializations include Object-relational data modeling, knowledge engineering and Big Data.

He has been into academic research and has published several of his research work results in International Journals and Conferences – including SwSTE in Israel and DASMA in Germany. He has also practitioner's experience while working with Software development companies.

He is a member of IEEE, ISTE, IAENG professional associations. He has served as research & project coordinator for PG studies at Engineering Institutes. Presently he is working as Associate Professor at VIT University, Chennai.