KRIS: Knowledge Representation Formalism

Rajendra Akerkar  
Chh. Shahu Central Institute of Business Education and Research  
Kolhapur – 416004, INDIA &  
Technomathematics Research Foundation  
Kolhapur – 416001  
INDIA

D. Rajesh Duthie  
Department of Computer Studies  
Chh. Shahu Central Institute of Business Education and Research  
Kolhapur – 416004, INDIA  
Email: drduthie@yahoo.com

Abstract

In this paper, we are dealing with representation of natural language sentences in the "new" propositional semantic network form. Our formalism gives more "natural" representation of any natural language sentences. We discuss its usefulness in our KRR system called Knowledge Representation Interactive System (KRIS) and also present some examples in new formalism.

Introduction

There are many language-related issues, which must be tackled while designing knowledge representation and reasoning formalism for natural language. The issues consists of representation of complex sentences, treatment of quantification and semantic of pragmatic aspects of natural language. First Order Predicate Logic (FOPL) and Semantic Network are widely used choices of representation language for natural language. There are a variety of expressions that can be used in ordinary language that is not expressible in standard knowledge representation languages, especially those based on FOPL and Semantic network. This section discusses some of the knowledge representation systems, which are already existing.

1. KL-ONE

Ronald Brahman (1979) designed a network notation for his Knowledge Language One (KL-One), which is the ancestor of many frame systems. Woods and Schmolze (1992) reported that about 20 system have been implemented in the KL-One tradition.

The network has nine ovals for concept nodes and nine arrows, which represent different kinds of links. The white ovals represent "generic concepts" for the types, as distinguished from the shaded oval, which is an "individual concept" for the instance 18. The oval marked with an asterisk (*) indicates that integer is a built-in or primitive type. The concepts Truck and Trailer, WtMeasure and VolMeasure would have to be defined by other KL-ONE diagrams.
Brahman drew a clear distinction between the different kinds of links in a semantic network. The double line arrows represent subtype-supertype links from Trailer Truck to Truck and from Truck to Vehicle. The arrow with a circle in the middle represents roles, which corresponds to the slots other than supertype in the previous frames.

Figure 1: KL-ONE representation for Truck and Trailer Truck

The Truck node has four roles labeled *UnloadedWt*, *MaxGrossWt*, *CargoCapacity* and *NumberofWheels*. The Trailer Truck node has two roles, one labeled *HasPart* and one that restricts the *NumberofWheels* role of Truck to the value of 18. The notation v/r at the target end of the role arrows indicates value restrictions of type constraints on the permissible values for those roles.

2. Schubert's Representation

In the work of Schubert *et al.*, which uses a semantic-network-based formalism (lately named ECO), variables are atomic nodes in the network. Type (and other) restrictions are specified by links to the variable nodes. There are no explicit universal or existential quantifiers. Free variables are implicitly universally quantified; Skolem arcs specify existentially quantified variable nodes. In Figure 2, the dotted arrow denotes a scope dependency between the universally quantified “man” and the existentially quantified “woman”. The implication (as in the FOPL representation) is represented by the “⇒” connected to the variables and the predicate “likes.” Because this is a nonlinear notation, sentences with branching quantifiers can be represented. However, the separation of variables from their constraints causes the representation
to be not “natural” relative to the original natural language. Moreover, since restrictions on possible fillers for variables appear to be simple type restrictions, there is no representation for noun phrases with restrictive relative clause complements, and consequently no representation for donkey sentences. The work of [Sowa, 1984; Sowa, 1992] with conceptual graphs, a form of semantic networks, is similar and has the same difficulties. Sowa does discuss the donkey sentence and his treatment of it resembles the Discourse Representation Structure (DRS) approach.

\[ (∀x)(∃y) [[x \text{ man}] ⇒ [[y \text{ woman}] & [x \text{ likes } y]]] \]

Figure 2: Schubert's representation for: Every man likes a woman

3. Fahlman's NETL Representation

NETL is a semantic network language, for connectionist architectures. It is a system for Representing and Using Real-World Data. Fahlman's work specified two representations for variables. The first representation is as a *TYPE-node that represents the “typical-member” of a set. This approach suffers from the typical inheritance problems (what Fahlman calls the “copy--confusion” problem) associated with such representations. The second representation for variables corresponded to universally quantified nodes called *EVERY-NODEs. Restrictions on these nodes can be general type or relative clause restrictions and are directly linked to the *EVERY-NODE. Existentially quantified variables are of type *INDV with Skolem arcs to *EVERY-NODE variables to indicate scope dependencies. An example is shown in Figure 3 of his representation for Any person that owns a dog that hates Fred also hates Fred. In Figure 3 the node with the black center dot is the universally quantified person that owns a dog that hates Fred. The labeled unfilled filled node pairs correspond to classes. The arcs with double arrowheads correspond to what Fahlman calls clauses, which correspond directly to clausal
restrictions on variables. The wavy line expresses the scope dependency between it and the *dog that hates Fred*. Finally, the *hates* relation of the top-level sentence is expressed by the arrow from the *person* node to the *Fred* node. This is more general than the variable representation of Schubert et al., because of the representation of complex clause restrictions. However, the form of the representation and its interpretation (as defining a set, rather than stating a proposition) is, again, unnatural relative to the sentence it represents. In principle, Fahlman's representation of variables appears capable of representing donkey sentences and branched quantifiers sentences; however, he does not discuss this, and semantics for such representations are unclear.

![Diagram](image)

Figure 3: **NETL representation for** "Any person who owns a dog that hates Fred also hates Fred".

4. **Kamp's Discourse Representation**

An alternative atomic variable representation theory is Discourse Representation Theory (DRT). DRT offers a representation for natural language discourses. A DRS (Discourse Representation Structure) consists of two parts: a set of *discourse markers* which are used to represent objects introduced in the discourse; and *conditions* on these objects. DRSs are characteristically drawn as boxes. The markers are shown in the upper part of the box with the conditions below. A typical DRS for the sentence "*A man walks*" is
Conditionals and the determiner “every” represented as a relation between two DRSs. Thus a DRS for the sentence "every man loves a woman" would be drawn as

\[
\begin{array}{c|c}
\text{x} & \text{man(x)} \\
\hline
& walk(x)
\end{array}
\quad \Rightarrow 
\begin{array}{c|c}
\text{Y} & \text{woman(Y)} \\
\hline
& \text{love(X,Y)}
\end{array}
\]

The interpretation of a DRS can be defined as follows. A DRS, V is true in a model M, when

- There exists a binding B which assigns each discourse marker in the top part of D to individuals in M such that each condition in D with respect to B becomes a true proposition in M.
- A \( \Rightarrow \) condition is defined to be true if for all bindings \( B' \) (which are an extension of \( B \), the binding the \( \Rightarrow \) condition is within, for just those markers introduced in the left DRS) that make the left DRS true in \( M \), there exists \( B'' \), an extension of \( B' \) for just those markers introduced in the right hand DRS, which makes the right hand DRS true in \( M \).

Donkey sentences offer a problem to standard logical treatments of language. The classic example is

\[\text{Every man who owns a donkey beats it.}\]

The problem can be seen when we try to give a first order representation of the sentence. In translating indefinite noun phrases in simple sentences like "a man walks", the indefinite introduces an existential quantifier, while in the donkey sentence such a simple translation might give

\[\forall x \exists y \ [(\text{man}(x) \land \text{donkey}(y) \land \text{owns}(x, y)) \rightarrow \text{beats}(x, y)]\]

But the above is not a meaning of the English utterance. The above expression is true in the following model

\[
\begin{align*}
\text{farmer}(a) & \quad \text{own} (a, b) \\
\text{donkey}(b) & \quad \text{cat}(e)
\end{align*}
\]
where *a* owns a donkey but does not beat it. Other placements of the existential will also fail to capture the meaning. In order to get a reasonable translations it is necessary to translate the indefinite that is within the scope of the universal as a universal quantifier rather than a simply an existential. Thus

\[ \forall x \forall y [(\text{man}(x) \land \text{donkey}(y) \land \text{owns}(x, y)) \rightarrow \text{beats}(x, y)] \]

will give us a correct reading. Although this problem can be solved by translating indefinites as universals in such a context it is considered strange to deal with indefinites in different ways depending on their context. DRT however solves this by offering a consistent treatment of indefinites irrespective of their context. Indefinites are always simple added to the DRS (there is an implicit existential for all discourse markers). Universal quantification is not over variables but over DRSs, thus existential in the sub-DRS of a => relation will effectively be treated as universals.

For example given the simple discourse "A *man* walks. *He* talks." After the first sentence we would have

```
<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>man(X)</td>
</tr>
<tr>
<td>walk(X)</td>
</tr>
</tbody>
</table>
```

In dealing with the pronoun "*He" at the start of the following sentence we must find an accessible marker that already exists in the DRS which we can related this pronoun too. Here we will introduce a new marker for the pronoun and use the condition *is* to related to to an appropriate existing marker. Thus after the second sentence we would have a DRS of the form

```
<p>| |</p>
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Y</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>man(X)</td>
</tr>
<tr>
<td>walk(X)</td>
</tr>
<tr>
<td>is(X,Y)</td>
</tr>
<tr>
<td>talk(X)</td>
</tr>
</tbody>
</table>
```

The exact form of the DRS at the end of the first sentence depends on the particular treatment of DRT.

The underlying formal system is FOPL-based and as with all the atomic variable representations, there is a separation of constraints from variables and the form of DRSs in not
“natural” in the same sense that a proposition that represented sentence would be. Here variables are still scoped and not conceptually complete.

5. KRIS

Knowledge Representation Interactive System KRIS is an Internet based interactive system using Knowledge Representation and Reasoning (KRR) scheme, namely propositional semantic net. KRIS deals with health issues related to cough and diabetics. The aim of this work is to educate laymen (or patient) in health hazards and enable him to take proper medical care. KRIS provides only relevant information to the user by filtering his utterances. KRIS uses propositional semantic network approach for user utterance decoding. The tutorial and definitions of medical information creates awareness among common man. This system prepares laymen with relevant medical terminology.

KRIS system architecture has been designed to integrate three tasks: the collection of information about patients, selection of medical information that are most likely to address user requirement, and the explanation of how to use the system.

In this we discuss the knowledge representation and reasoning module of the KRIS system.

5.1 KRIS "Knowledge Representation and Reasoning" Module:

![Figure 5: KRIS "representation" module](image)

The "Knowledge Representation" module does the following in order to achieve the "more natural" knowledge representation and reasoning.
- It reasons the user’s natural language utterances or definition requests or help requests,
- It sends knowledge about the user, sickness and medical information to User / Medical Knowledge Module,
- It instructs Knowledge Reconstruction Module whenever necessary.

5.2 Algorithm for representation of sentences:

The following are the algorithmic schemes of the KRIS "Knowledge Representation" module for natural language discourse representation. It follows these procedures to represent the natural language sentences in to reconstructable computer representation.

**Representation:**

- **Step -I** Accept the natural language sentence.
- **Step -II** Give it to the Line Grammar Parser. If "it" is present reform the sentence by replacing "it" with intended reference word.
- **Step -III** Receive the optimum constituent tree representation of the original sentence
- **Step -VI** Build our representation of the sentence with the help of Line Property, Line Pattern, Dictionaries and Node Properties. If "it" replacement was done at Step-I, then give the memory reference for the intended reference word
- **Step -V** Represent and store the sentence in tree representation.

**Reasoning:**

- **Step -I** Query the tree representation for the desired knowledge by using the node number.
- **Step -VI** Rebuild the tree representation of the sentence in to "More close to natural" natural language sentence by back tracking the tree.
- **Step -V** Display in natural language format.

**Notations and explanation:**

In a network, P is a palpable node, and N is a structured individual which is not used previously. P is represented as a rectangle and N is represented as a circle. N suffixes a unique number to identify an entity. P filled with plain background depicts generic or common node and vertical line background depicts individual node. WHNP is considered as a "If" condition,
PNALL represent Universal quantifier, ACIN the LHS "acts" in the RHS, ACBY the RHS "acts" in the LHS, PN indicates proper name, AC indicates action and M indicates a member. Line with an arrow indicates the data flow. And the line property indicates the relationship between palpable nodes. Double arrow indicates an "is a" relationship. Line with a horizontal "-" strike indicate "not" and line with a "×" strike indicate "need not".

5.3 Examples (Representation in KRIS system):

Example - 1

Natural Language Sentence:

Lucy pets a dog

Link Grammar Parser Output:

(S (NP Lucy) (VP pets (NP a dog)))

KRIS sentence representation:

(S (NP Lucy-PN-I) (VP pets-AC (NP a dog-M-G)))

KRIS tree representation:

KRIS back tracking:

Lucy - PN pets - AC a dog - M

Lucy pets a dog
Example - 2

Natural Language Sentence:
Every farmer who owns a donkey beats it.

Link Grammar Parser Output:
(S (NP (NP Every farmer) (SBAR (WHNP who) (S (VP owns (NP a donkey)))))) (VP beats (NP a donkey))

Note: "it" is at present so sentence reformation is required. "it" is replaced with "a donkey"
Every farmer who owns a donkey beats a donkey.

KRIS sentence representation:
(S (NP (NP farmer-PNALL) (S (VP owns - AC-WHNP (NP a donkey-PN)))) (VP beats-AC-TH (NP [memory location of a donkey]-PN)))

Note: NP "Every" is added to PN as PNALL. WHNP is considered as IF condition and is added to the next verb.

KRIS tree representation:

KRIS back tracking:
Farmer PNALL owns - AC WHNP a donkey - M beats - AC a donkey - M

All farmer if owns a donkey beats a donkey

Note: Even though the donkey sentence seems ok, it does not give the intended meaning. In this case a farmer can own a donkey and beat another one. So still improvement is required in this representation.
Example - 3

Natural Language Sentence:
All malaria is caused by malarial-parasite, it need not be present in the smear.

Link Grammar Parser Output:

(S (S (NP All malaria) (VP is (VP caused (PP by (NP malarial-parasite)))))) and (S (NP malarial-parasite) (VP need not (VP be (ADJP present (PP in (NP the smear))))))

Note: "it" is at present so sentence reformation is required. "," is replaced with "and" and "it" is replaced with "malarial-parasite"

All malaria is caused by malarial-parasite and malarial-parasite need not be present in the smear.

KRIS sentence representation:

(S (S (NP malaria-PNALL) (VP is-IS (VP caused-ACBY-P (NP malarial-parasite-PN)))) and (S (NP [address of malarial-parasite]) (VP need not-NNOT (VP be (ADJP present-ACIN (NP the smear-PN))))))

Note: "All-ALL" is added to PN as PNALL. PP "by" and "IN" is added to the previous verb and SDJP. VP "be" considered with the next ADJP. "and" is used to start a new branch.

KRIS tree representation:

KRIS back tracking:

Malaria PN-ALL is cause - ACBY-P malarial-parasite - M and malarial-parasite - M present - NEED NOT -IN the smear - PN

All malaria is caused by malarial-parasite and malarial-parasite need not present in the smear.
**Future Direction:**

Some more complex natural language sentences can be represented where the backtracking could be more natural to the original sentence. The reasoning of the representation has to be worked on still further.

**Reference:**

1. [Sowa, 1984]  

2. [Sowa, 1992]  

3. [Shapiro, 1980]  

4. [John Lafferty, Daniel Sleator, Davy Temperley, 2001]  

5. [D. Rajesh Duthie, R. A. Akerkar, 2000]  

6. [D. Rajesh Duthie, R. A. Akerkar, 2001]  