Distributedness and Non-Linearity of LOLITA's Semantic Network

Short S., Shiu S., Garigliano R.
Laboratory for Natural Language Engineering
School of Computer Science
University of Durham
South Road
Durham DH1 3LE, United Kingdom.
sengan.short@durham.ac.uk

Abstract

This paper describes SemNet the internal Knowledge Representation for LOLITA[1]. LOLITA is a large scale Natural Language Engineering (NLE) system. As such the internal representation must be richly expressive, natural (with respect to Natural Language), and efficient. In network representations knowledge is gleaned by traversing the graph. The paper introduces two properties, (**distributedness** and **non-linearity**) of networks which directly relate to the efficiency by which knowledge is obtained. SemNet is shown to have the specified properties thus distinguishing it (in terms of efficiency) as a suitable representation for large scale NLE.

1 Introduction

Natural Language Engineering (LRE, 1992) (Smith, 1995) is a more pragmatic approach to Natural Language Processing than traditional Computational Linguistics. It involves seeking a large scale solution to NLP by applying engineering principles to utilise all available resources. This is in contrast to trying to scale up domain specific applications, or by first attempting to obtain a general theory of language.

A core problem for NLE is the design of the internal representation. An ideal representation should have several features including: rich expressiveness, readability, efficient storage/retrieval of information. Semantic networks have long been recognised as having the potential to fulfil many of these requirements. This paper introduces two new criteria for semantic networks **distributedness** and **non-linearity** and discusses their relevance to NLE. They are particularly relevant in large networks where search efficiency is vital to real-time system operation.

The large scale NLE system LOLITA (Long, 1993) (Smith, 1995) has been designed and implemented following an NLE methodology. Its internal representation, SemNet, is a semantic network satisfying the above features. The system analyses complex text, and expresses its meaning in SemNet. This information can then be used to perform reasoning, information retrieval, or translation. Knowledge held in the network can be expressed for users by generating natural language from SemNet.

The fundamental principle of Semantic Networks is that information is stored as nodes and arcs, which represent concepts and relationships respectively. Within this framework a wide variety of networks exist, e.g. KL-ONE based systems (Woods, 1992), SNePS/ANALOG (Ali, 1993), and Conceptual Graph Theory (Sowa, 1984). Direct comparison with these would not be justified as each has been designed with different objectives. However, the paper does discuss aspects of these representations in order to highlight differences and why the authors believe SemNet is a powerful (with respect to search) representation for large scale NLE.

The rest of this paper is organised as follows. Section 2 introduces distributedness and non-linearity as criteria for judging networks and explains their significance for NLE. Section 3 describes the core of SemNet. Section 4 discusses the distributedness and non-linearity of SemNet and some other well known network representations. Section 5 draws conclusions.

2 Distributedness and Non-Linearity

A syntactic representation will have a semantic model. The degree to which such a representation

[1] Large-scale, Object-based, Linguistic Interactor, Translator, and Analyser

is **distributed** depends on the proportion of sections of the representation which are both syntactically legal and give information which is sound with respect to the model. A network is said to be **non-linear** if reading from any node and in any direction gives information which is sound with respect to the model.

In a large knowledge base, the amount of information that must be accessed in order to retrieve a particular fact is critical. In a semantic network information is not accessed directly as in a table, but by traversing its arcs as a graph. Retrieval therefore corresponds to searching for a particular type of information from a known node in the net. For instance, if the problem is to determine John's height, the origin-node where the search starts is "John", and the type of information is "height". In such a model, the efficiency of retrieval is determined by:

- **topological distance** Since the graph is traversed arc by arc, the number of arcs that must be traversed to reach the relevant piece of information determines the efficiency of retrieval. It is therefore important to ensure relevant information is represented locally.

- **determinism of the search** Although information may only be a few arcs away from the thing described, there may be many paths leading from the thing, and of equal distance. Thus the potential search space to be explored before finding the relevant information may be quite large. This can be reduced by making the path to traverse uniquely recognisable.

- **non-linearity** In order to ensure efficiency, it is important that the shortest path possible will be that traversed when searching for the required information. This cannot be achieved if the semantic network must be traversed in any pre-established order, for a meaning to be assigned to it. This absence of prescribed order corresponds to 'non-linearity'.

- **distributedness** Information is expressed as a cluster of nodes and arcs in the semantic net. For retrieval, the type of the cluster must be identified. The efficiency of this step depends how information is ordered within each cluster. Each cluster may express a separate piece of information. Alternatively separate pieces of information may be expressed as a single more complex cluster. In the first case, the cluster will be small and easily recognisable, whereas in the second case a lot of effort will be required to recognise the larger cluster. Furthermore, extracting the relevant piece of information from a complex cluster requires identifying and filtering out irrelevant information. This step is not necessary for simple clusters which only express the relevant information. Thus smaller clusters expressing separate pieces of information ensure more efficient retrieval. This leads to the definition of distributedness as the degree to which independent pieces of information are expressed as independent clusters.

  Full distributedness can obviously be obtained by expressing every piece of information that could possibly be conceived independently as a separate cluster. However, as separate pieces of information are usually used in conjunction, it may be advantageous to use one more complex cluster rather than many simple ones: this will reduce the number of clusters to find, and the amount of net to search. A simple but effective method of penalising the complete flattening approach is to consider the ratio of distributedness to number of nodes and arcs for statements expressed in the net.

This discussion will focus more specifically on the last two criteria. Although a quantitative measure of the criteria is available, to simplify the discussion, only their qualitative definitions will be used.

3 SemNet: LOLITA's Semantic Network

SemNet has been designed specifically for large scale NLE. This section describes some of the core aspects needed for this discussion. SemNet is a graph of nodes and arcs which can be read/traversed in either direction. Associated with each node are controls. Controls hold structured information about their nodes. Because they are internal to each node they are not subject (with respect to SemNet) to the search properties mentioned previously.

There are three types of nodes: entities, events (assertions) and actions (roles). There are three types of directed arcs: subject, object and action [2] which can be read/traversed in either direction.

[2] The names of these arcs should neither be confused with their grammatical counterpart, or with the case analysis of (Fillmore, 1968). They can be thought of as argument$_1$, argument$_2$ and argument$_3$.

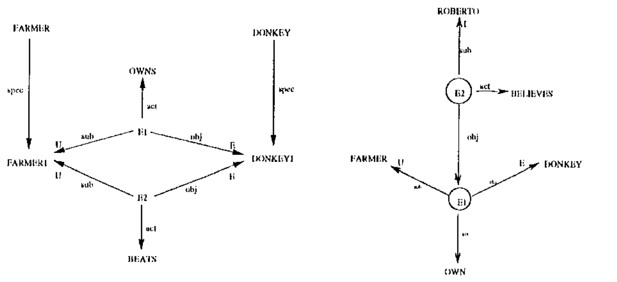

Figure 1: Figure 1: (a) SemNet event for "Every farmer that owns a donkey beats it." (b) SemNet epistemic event for "Roberto believes that every farmer owns a donkey."

Only event nodes can have a subject, object or action. Only action nodes can be an action for an event node. A control for each node specifies its type. E_3 in Figure 1(a) asserts that two entities ($FARMER_1$ and $DONKEY_1$) are in an beating relationship. The subject/object arcs ensure that it is understood that farmers beat donkeys and not vice versa.

A fundamental principle of the design is that concepts are not reduced to primitives. The meaning of any node is defined in terms of its relationship with other nodes, so ultimately each node is only fully defined by the whole semantic network. It should be noted that the event nodes can be the subject or object of another event so that SemNet is 'propositional' in the sense used by (Kumar, 1993).

3.1 Quantification

A problem for networks is to ensure that relationships refer to concepts unambiguously (Woods, 1991). For example without reference information, E_3 in figure 1(a), could mean any of: a farmer beats a donkey, all farmers beat a donkey, all farmers beat a (the same) donkey, or all farmers beat all donkeys. In SemNet this ambiguity is resolved by attaching the following quantification[3] labels to arcs:

- **Universal U** refers to the instances of the concept and says that all the instances of the concept are involved in relationship specified by the event.

- **Individual I** refers to the concept as a whole and says that it is involved in the relationship specified by the event.

- **Existential E** refers to the instances of the concept, but the instance involved depends on the particular instance of some other universally quantified concept which is involved in the event.

Existential arcs can be thought of as existentially quantified variables in First Order Logic (FOL), which are necessarily scoped by some universal. To represent an existential that is not scoped by a universal we use the individual rank. Thus for E_2 in figure 1(a), the donkey that is involved depends on the farmer. This could be interpreted[4] into FOL as:-

$$\forall x \exists y (Farmer1(x) \rightarrow Donkey1(y) \wedge Beats(x, y))$$

To demonstrate how SemNet can represent complex expressions, consider the well known donkey sentence: "Every farmer that owns a donkey beats it." Of course to capture this unambiguously the meaning has to be agreed. It is assumed that it is correctly represented by the FOL statement:-

$$\forall x \forall y ((Farmer(x) \wedge Donkey(y) \wedge Owns(x, y)) \rightarrow Beats(x, y))$$

SemNet represents this as shown in figure 1(a). The event E_2 is an 'observing' event, it represents the assertion of the donkey sentence.[5] E_1 is a 'defining' event used to build the complex concepts $FARMER_1$ (farmers that own (and so beat) donkeys) and $DONKEY_1$ (donkeys that are owned by these farmers). For clarity the events linking hierarchies of farmers and donkeys have been written as spec (for specialisation).

3.2 Representation of Belief and Intensional Knowledge

It is important to emphasise that the information which is recorded within SemNet is intended to reflect the world as it is to be understood by the agent that uses the network. No claim is made that the representation reflects the world as it really is (if there is such a thing), nor even that the representation reflects some consensus view of the way the world is. Thus from an external viewpoint the concepts should be interpreted as intensional. However from the agent's viewpoint, they

[3] It should be noted that this paper presents a simplified account of the quantification scheme used in SemNet. The full scheme is described in (Short, 1996).

[4] A current project is looking at providing a formal, type theoretic, semantics for SemNet (Shiu, 1996)

[5] Note that Farmer1 in the first formula above represents "farmers that own donkeys" so this formula is inferred by second (donkey sentence) formula, as would be expected.

constitute the world it believes in, and thus may be either extensional or intensional. As it is cumbersome to repeat that we are dealing with the agent's beliefs, this shall be taken as read in the rest of this section. Similarly, the agent will be referred to by the name LOLITA, as this is the only agent so far which uses SemNet.

It is possible for LOLITA to believe that another agent believes some relation to hold. For example, LOLITA may believe that "Roberto believes that every farmer owns a donkey.", see figure 1(b). Distributedness requires that one may read E_1 and E_2 independently from the other. According to the description given so far, there is no difference between the way E_1 is represented when LOLITA believes it, and when it is there merely as a part of some other event which LOLITA believes (of course it could be both). Thus if E_1 is read on its own, all that would be said is that some agent potentially believes in the relation it expresses. To identify any such agent would require some form of search which would be inefficient as very often the agent will be LOLITA. Distributedness can be better exploited by using a control. A status control makes this distinction, it takes two values: real (when LOLITA believes in the event), and hypothetical (otherwise).

Statements may either be made about concepts or about the things concepts refer to. These cases need to be distinguished. For example, consider the three concepts "the morning star", the "evening star" and "Venus". The morning star is the last point of light in the sky to disappear at dawn, the evening star is the first point of light in the sky to appear at dusk, and Venus is a particular planet of the solar system. Thus, although they have the same extension they are different intensionally. Since the representation represents different concepts by different nodes, there must be a means to state that two concepts refer to the same object. This is done using an extensional synonym event to connect the concepts. The synonym event has no effect on distributedness or non-linearity but affects topological distance and determinism of search adversely.

This price is justified as distinguishing intensional and extensional concepts is important in many situations. For example, if one tells LOLITA "I need a hammer", one does not want her to answer that she has found a hammer: "the hammer that you need". Such misunderstandings will occur unless the hammer is correctly understood as intensional and distinguished in the representation from extensional hammers. This is done using a 'tensional' control stating whether the node has an extension in the world, an extension in some other frame of existence, such as Agatha Christie's fictional world where the hammer was the murder weapon, or an unknown extension. Note that 'tensionality' and belief are independent. A relation may be not only hypothetical, but also intensional: "John believes he needs a hammer".

3.3 Features exploiting the search properties

If controls were written as events, they would be uni-directional, involving an uni-directional subject or object arc, i.e. if a control refers to a node of the network, there need not be any information on that node back to the control. Such uni-directional events are beneficial to the determinism of search since they restrict the number of arcs that can be traversed from any node. Controls represent a further improvement on distributedness since they reduce the number of required event nodes without affecting richness. The information expressed as controls is never referred to by other events.

Controls allow defaulting, which is illegal for the network. Defaulting consists of assuming some fact, when no information of that fact's type is expressed explicitly. This means that the information expressed by some section of SemNet can be unsound with respect to the full semantic net. It might appear sufficient to check all the events attached to a node to determine whether a default applies, but it should be remembered that events can also be inherited from far up the inheritance hierarchy. Indeed, one of the practical advantages of distributedness is that it does away with the need of inheriting all a nodes 'ancestors' information while allowing the benefits of a hierarchical knowledge base.

4 Distributedness and Non-Linearity in known Networks

This section begins with a discussion of the distributedness and non-linearity of SemNet. The latter part investigates the properties for other representations.

In SemNet a single node (say E_1 in figure 1(a)) tells us nothing, except that some concept exists. Its controls will specify its type (event, extensional, real in this case). Every arc attached to the node specifies E_1 further: the action arc specifies its type (an owning relation), the subject arc specifies that it is all the instances of $FARMER_1$ that participate in the owning relation in the subject role, and the object arc specifies that there is a (scoped) instance of DONKEY$_1$ which participates in the relation in the object role. This information can be combined into the interpretation that all instances of FARMER$_1$ own a (scoped) instance of DONKEY$_1$ Thus each arc conveys an independent piece of information which can be combined compositionally with other information known about the node. The interpretation assigned to a node need not be retracted when reading more information specifying it: rather it is augmented by this additional information. Further information can be obtained by reading more of the graph: FARMER$_1$ is a 'subset'[6] of FARMER. If the whole of the graph in figure 1(a) is traversed then the donkey sentence is inferred. E$_1$ is still not entirely defined: each node is only fully defined by the whole semantic network. This example illustrates the full distributedness of SemNet.

To demonstrate non-linearity consider again the highlighted section of figure 1(a). Reading from FARMER to DONKEY$_1$, gives:[7] "Entity FARMER is a 'superset' of FARMER$_1$, which is a universal subject of E$_2$, which has action BEATS, and existential object DONKEY$_1$". Alternatively reading from DONKEY$_1$ to FARMER, gives: "DONKEY$_1$ is an existential object for E$_2$, which has action BEATS, and universal subject FARMER$_1$, which has 'superset' FARMER". Clearly both readings convey the same information and each sub part would be sound information in its own right. SemNet is therefore non-linear.

The remainder of this section describes some initial investigations into the distributedness and non-linearity of other representations. This is done not as a criticism of other networks, but to test out the relevance of these new properties and also to try and show where SemNet differs from other well known networks.

The T-Box of KL-ONE based systems (Woods, 1992), (Beierle, 1992) is Semantic Net based, the A-Box usually consists of a subset of FOL. Since these assertions are expressed as ordinary logical statements, they must be read from left to right: there is a prescribed order for reading them so they are not non-linear. Similarly reading arbitrary sections of the statements is unlikely to give meaningful or sound statements. For example, reading part of the donkey sentence gives: $\forall x \forall y Beats(x,y)$ which is not sound with respect to the full reading. Thus assertions in KL-ONE are neither distributed nor non-linear.

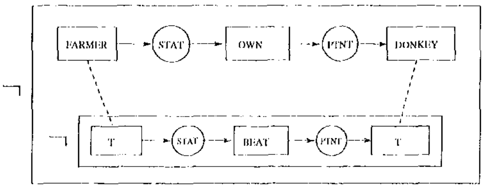

Figure 2: Figure 2: CGT - version of 'Donkey sentence'.

CGT (Sowa, 1984) builds complex logical assertions using contexts. Figure 2 shows how the donkey sentence is represented by CGT. This use of contexts requires the whole context to be read/traversed for any sense to be made. For example, the innermost sub-context is interpreted as "Farmers do not Beat Donkeys". If this is read independently from the rest, the interpretation derived is not sound with respect to that provided by the full context. Thus sub-contexts are not combined compositionally to form the full context. For CGT the independent pieces of network must be at the level of a context rather than its components. This is less distributed than SemNet, where arcs form the smallest independent pieces of the network.

Partitioned Networks (Hendrix, 1979) have a similar notion of context, called spaces. These spaces are collections of nodes and arcs of the full network. They are associated with nodes in the network, allowing them to be referred to. This allows the set of statements within a space to be negated, be the objects of someone's belief, or be treated in any other propositional way. A hierarchy of these spaces states which spaces have contents visible to which other spaces. A space, and the spaces visible from it, is called a vista. This leads to multiple views of a semantic net, where different vistas express possibly contradictory statements. Each vista is independent from the rest of the network in that the rest of the network is invisible from it. However within a vista, spaces may be negated. Indeed, if a space is negated, the space in which the negation is made is visible from it. As a result, the interpretation of parts of a vista is not guaranteed to be sound with respect to the vista itself. Partitioned networks thus have a low distributedness, but provide an alternative means of limiting the amount of information to be processed. Unlike distributedness however, the creation of vistas requires additional processing.

[6] The terms subset and superset are used loosely here, formally concepts are interpreted as types and so the interpretation is not strictly correct

[7] LOLITA is of course able to generate English statements rather than the following.

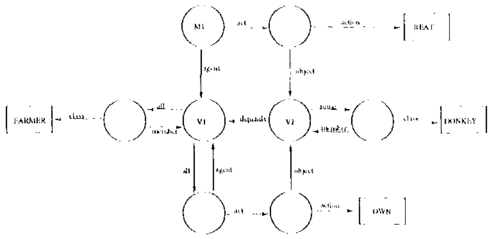

Figure 3: Figure 3: ANALOG version of the 'Donkey sentence'

SemNet does not have any such notion of context which can be negated. Instead, a nonaction arc replaces the action arc on the negated event. If a set of events are to be negated, as in the negation of "farmer Giles owns a donkey and likes a cat", it is the logical connective event which is negated. Nested negations are normalised into zero or one negations.

ANALOG (Ali, 1993) is a logic for natural language with structured variables. Figure 3 shows how ANALOG represents the donkey sentence. This representation seems quite close to SemNet and indeed comes close to achieving the level of distributedness and non-linearity which the authors seek. However, as argued previously, efficiency of search depends on the ratio of distributedness to the size of the graph required to read the statement. Expressing quantification on the arcs maintains the possibility to read or ignore the quantification, while reducing the graph's size. ANALOG also provides the possibility of reading the quantification independently from the relation in which it occurs. However the authors have not found any application in which this is or could be useful in their work building the LOLITA NLE system. Thus the distributedness achieved in SemNet provides a greater efficiency than ANALOG's[8].

5 Conclusions

Two new measures of efficiency for large scale NLE systems have been introduced: distributedness and non-linearity. SemNet has been designed with these properties in mind. The resulting representation has been compared with other widely used representations in the field of NLP. SemNet was found to satisfy these criteria best. It was also shown to be propositional and to have a rich syntax for addressing with problems such as quantification and intensionality. For these reasons, the authors believe that SemNet is an efficient and rich internal representation for large scale NLE systems, such as LOLITA.

[8] SemNet is able to represent the donkey sentence using fewer nodes and arcs, providing a better tradeoff between distributedness and node number.

References

S. S. Ali, and S. C. Shapiro. 1993. Natural Language Processing using a propositional semantics network with structured variables. In *Minds and Machines, 3, No 4.*

C. J. Fillmore. 1968. The case for case. In E. W. Bach, R. T. Harms, editors, *Universals in linguistic theory* Holt, Rinehart and Winston.

G. G. Hendrix. 1979. Encoding Knowledge in Partitioned Networks. In N. V. Findler, editor, *Associative Networks: Representation and Use of Knowledge by Computers* Academic Press.

D.Kumar, and H. Chalupsky. 1993. Guest Editorial for Issue on Propositional Knowledge Representation. In *Journal of Experimental and Theoretical Artificial Intelligence, 5, No 2.*

D. Long, and R. Garigliano. 1994. Reasoning by Analogy and Causality: A model and application. Ellis Horwood.

1992. LRE: Linguistic Research and Engineering european programme.

M. Smith. 1995. Natural Language Generation in the LOLITA system: An Engineering Approach. submitted as PhD thesis, Dept Computer Science, University of Durham, UK.

J. F. Sowa. 1984. Conceptual Structures: Information Processing in Mind and Machine. Addison Wesley.

C. Beierle, U. Hedstuck, U. Pletat, P. H. Schmitt, and J. Siekmann. 1992. An order sorted logic for knowledge representation systems. Artificial Intelligence, 55.

S. Shiu, Z. Luo, R. Garigliano. 1996 Type theoretic semantics for SemNet. Proceedings of the international conference on Formal and Applied Practical Reasoning (FAPR'96).

S. Short. forthcoming 1996. The Knowledge Representation of LOLITA. Phd. Thesis, Dept Computer Science, University of Durham, UK.

W. A. Woods. 1991. Understanding subsumption and taxonomy: A framework for progress. In J. F. Sowa, editor, *Principles of Semantic Networks: Explorations in the Representation of Knowledge* chapter 1. Morgan Kauffman.

W. A. Woods, and J. G. Schmolze. 1992. The KL-One family. In *Computers Mathematics and Applications, 23, No 2.*