Natural Language Engineering **3** (2/3): 147–169. Printed in the United Kingdom

Using a semantic network for information extraction

ROBERT GAIZAUSKAS and KEVIN HUMPHREYS
Department of Computer Science, University of Sheffield
Regent Court, Portobello Road, Sheffield S1 4DP, UK
e-mail: {r.gaizauskas,k.humphreys}@dcs.shef.ac.uk

(*Received 25 April 1997*)

Abstract

This paper describes the approach to knowledge representation taken in the *LaSIE* Information Extraction (IE) system. Unlike many IE systems that skim texts and use large collections of shallow, domain-specific patterns and heuristics to fill in templates, *LaSIE* attempts a fuller text analysis, first translating individual sentences to a quasi-logical form, and then constructing a weak discourse model of the entire text from which template fills are finally derived. Underpinning the system is a general 'world model', represented as a semantic net, which is extended during the processing of a text by adding the classes and instances described in that text. In the paper we describe the system's knowledge representation formalisms, their use in the IE task, and how the knowledge represented in them is acquired, including experiments to extend the system's coverage using the WordNet general purpose semantic network. Preliminary evaluations of our approach, through the Sixth DARPA Message Understanding Conference, indicate comparable performance to shallower approaches. However, we believe its generality and extensibility offer a route towards the higher precision that is required of IE systems if they are to become genuinely usable technologies.

1 Introduction

This paper describes the approach to Knowledge Representation (KR) taken in the *LaSIE* system (Large Scale Information Extraction system), an information extraction system initially developed for participation in the Sixth Message Understanding Conference (DARPA 1995; Grishman and Sundheim 1996), and currently being used as a 'core information extraction engine' underlying a number of language engineering projects (Cunningham 1996).

Information Extraction (*IE*), as the term is coming to be used in the NLP community (see, for example, Cowie and Lehnert (1996)), is the mapping of short natural language texts (such as newswire reports) into predefined, structured representations, or *templates*, which, when filled, represent an extract of key information from the original text. A typical IE task, for example, might involve processing business newswire texts containing announcements of joint ventures and extracting from them the names and nationalities of the participating companies, the activity of the venture, the start date of the venture, its capitalisation, and so on.

The approach taken towards KR within the *LaSIE* system has been conditioned by our interest in IE. Focusing on IE within the context of the DARPA Message Understanding Conferences (MUC), has meant fully implementing a system that:

- processes unrestricted 'real world' (*Wall Street Journal*) text containing large numbers of proper names, idiosyncratic punctuation, idioms, etc.;
- processes relatively large volumes of text in a reasonable time;
- needs to achieve only a relatively shallow level of understanding in a predefined domain area;
- can be ported to a new domain area relatively rapidly (details of the MUC-6 scenario template were given to participants only 4 weeks before the evaluation, to encourage portability).

While MUC is an artificial exercise designed to foster technology development, these constraints are also likely to hold of any real IE application.

The processing in an IE system can be divided roughly into two components: a syntactic component that works on single sentences of the input and a discourse-level component that integrates information from the syntactic analyses of multiple sentences. The former typically does tokenisation, part-of-speech tagging, phrasal pattern matching or parsing and produces a regularised form which may be anything from a partially filled template to a full logical form. The latter takes whatever regularised form has been produced by the former and, perhaps using more general knowledge of domain, attempts to integrate information from the individual sentence representations into a larger scale structure which ultimately either is, or serves to provide, the information for the final template.

This paper focuses on the discourse component of *LaSIE*. In particular, it describes our approach to representing the meaning of individual sentences in a text, to representing general knowledge needed to resolve ambiguities in a text, and to representing the meaning of an entire short text or discourse (of course in all cases the meaning captured is only partial). It also describes some of the mechanisms that use these representations, in order to demonstrate their motivation and utility. Finally, it explains how the knowledge currently used by the system has been acquired, and discusses new approaches we are now investigating for deriving more extensive knowledge bases from the WordNet semantic net. But first, we place our work in the context of current efforts in IE.

2 Background and context: Shallow *vs* deep IE

Given the pragmatic constraints imposed by the IE task, the relatively limited understanding required, and the competitive aspects of MUC which can encourage 'maximal gain with minimal pain' solutions, many developers of MUC IE systems have, in recent years, opted for engineering solutions that de-emphasise the substantial body of theoretical work both in computational syntax and semantics and in knowledge representation and reasoning. This de-emphasis is perhaps most dramatically illustrated by the SRI team who abandoned, quite consciously, the theoretically motivated TACITUS system after MUC-3 in favour of the pragmatically motivated FASTUS system which they have used for MUC-4 through MUC-6. TACITUS (Hobbs 1991) attempted a full syntactic analysis, using a large scale grammar of English, performed semantic interpretation to produce first-order predicate calculus representations, and then used abductive reasoning to interpret the semantic representations of individual sentences in the context of a schema pertaining to the scenario of interest. FASTUS (Hobbs *et al.* 1992; Appelt *et al.* 1993, 1995), by contrast, uses a cascade of finite-state transducers that successively tokenise, recognise names, recognise phrases, recognise template patterns, and then combine or merge partially filled templates to generate the final template. SRI have been keen to stress that this change in direction has not happened because they concluded that the TACITUS approach was faulty, but because they believed it was inappropriate for the task. TACITUS did *text understanding*, FASTUS *information extraction*, the latter, on their view, a much simpler task that does not require the theoretical and computational sophistication of TACITUS. The chief gain from the switch has been speed (from 36 hours to 12 minutes for 100 texts between MUC-3 and MUC-4) and to some extent ease of porting to new domains. Though performance results, in terms of combined precision and recall, are not strictly comparable between MUCs, it is worth noting that FASTUS scores (on the key template filling task) surpassed TACITUS scores by about 16% between MUC-3 and MUC-4, mostly due to increased recall.

SRI have not been alone in moving away from a more powerful, linguistically motivated approach towards a more restricted, task-specific, engineering-driven approach. Between MUC-4 and MUC-5 the GE team moved from a system which combined linguistic analysis based on chart parsing with a unification-style grammar and domain-driven conceptual processing to a system that used finite-state pattern matching to annotate sentences, then did semantic interpretation of annotated sentences, and finally merged the semantic fragments into the templates (Krupka *et al.* 1992; Jacobs *et al.* 1995). Between MUC-5 and MUC-6 the NYU team abandoned the Proteus system which had been based on a large scale grammar of English developed at NYU over more than twenty years, to adopt an approach using a finite-state pattern matcher, which, however, still produced a logical form which was then integrated into a concept hierarchy for reference resolution and then template generation (Grishman and Sterling 1993; Grishman 1995).

We have dwelt at some length on this movement away from the more theoretically motivated work of the 1980s because this movement has engendered considerable debate (and rhetoric) about ‘shallow’ versus ‘deep’ approaches to information extraction. This debate is ongoing and needs to be referenced when introducing an information extraction system, as we shall be shortly be doing, in order to position the system properly in the current context. In particular, it could be maintained that, since the trend has been going in the direction of the ‘shallow’ and away from the ‘deep’, and since general approaches to knowledge representation are associated with deep analysis it follows that there is no need for a general approach to knowledge representation in the context of IE at all. This argument is clearly fallacious; we also believe its conclusion remains very much an open question.

It is important to distinguish at least two ways in which processing in an IE system can be shallower or deeper, corresponding to the two components of IE systems identified in section 1. First, the syntactic analysis the system performs can be more or less thorough. At one extreme there are systems which employ formally weak mechanisms (finite-state pattern matchers) to apply domain-specific lexically-triggered patterns; at the other extreme there are systems which employ formally stronger mechanisms (complete parsers for context-free or even more expressive formalisms) to apply general grammars of natural language. Second, discourse-level processing, by which we mean the process of integrating the information obtained from the syntactic analysis of the multiple individual input sentences making up the text, can be more or less general. Thus, the semantic representation derived from the syntactic analysis can be expressed in a more or less general formalism and manipulated by more or less general algorithms which attempt to integrate it into a more or less general model of the text and domain. There may or may not be any attempt to use declaratively represented world and domain knowledge to help in resolving ambiguities of attachment, word sense, quantifier scope, and coreference, or to support inference-driven template filling. At one extreme there are information extraction systems which produce semantic representations that are fragments of the target template for just those sentences that yield template relevant information, and then merge these using *ad hoc* heuristics to produce the final template; at the other extreme there are systems that use abductive theorem provers and axiomatisations of the domain to compute the least cost explanation of the first order logic expressions derived from every sentence in the input, and then generate the template from the resulting underlying logical model.

Given these distinctions we are now in a position to begin to locate the *LaSIE* system in the space of IE systems. *LaSIE*'s approach to syntactic analysis is what is best described as fragmentary parsing: a bottom-up chart parser applies in sequence two simple unification-based grammars (one for proper names, one for phrases) to yield a set of partial parses, from which a 'best' is selected and then semantically interpreted to yield a predicate-argument like representation (see section 4.1 for more details). Semantic information is subsequently used to extend these partial analyses.

On the conventional shallow-deep scale of syntactic analysis *LaSIE* is somewhere in the middle, clearly deeper than the SRI, NYU, SRA (Krupka 1995) and MITRE (Aberdeen *et al.* 1995) MUC-6 systems, but shallower than the earlier TACITUS and Proteus systems and the MUC-6 PIE system (Lin 1995) which uses a broad coverage government and binding grammar. Perhaps closest to the *LaSIE* approach to syntactic analysis is the BBN PLUM system (Weischedel 1995) which also uses a domain-independent grammar to do fast partial parsing.

Nothing in the subsequent discourse processing stage of *LaSIE* is committed to its current approach to syntactic analysis: all that is required is that semantically interpretable phrases (Ss, NPs, VPs and PPs) be identified and converted to a canonical predicate-argument form. This could be achieved by finite-state pattern matching techniques (as the NYU and MITRE MUC-6 systems do) or by even deeper syntactic analysers. For these reasons, while we believe the *LaSIE* approach to syntactic analysis to be worthy, we are not deeply committed to it.

The *LaSIE* approach to discourse processing is to construct a single meaning representation for a text, which we call a *discourse model*, from which the information required to fill an IE task-specific template may be derived. This is done by extending a semantic net which declaratively represents the system's domain knowledge prior to processing the text into a new semantic net which represents the system's domain knowledge plus the knowledge it has obtained by processing the text (the approach is explained in detail in the following sections). Thus, from the *LaSIE* perspective, IE is a process of integrating a new text into a semantic network and then deriving, by inferential processing, a task-specific template from the semantic net. The key difference between the *LaSIE* approach and shallower IE approaches to discourse processing is that the discourse model and intermediate representations used to derive it in *LaSIE* are less task- and template-specific than those used in other approaches. However, while we are committed to deriving a richer meaning representation than many IE systems, a representation that goes beyond the template itself, we are still attempting to achieve only limited, domain-dependent understanding. Hence, the representations and mechanisms employed in *LaSIE* still miss much meaning, and cannot be said to be attempting 'full understanding' (whatever that might mean).

On the shallow-deep scale of discourse-level processing *LaSIE* is, once again, somewhere in the middle, though perhaps in this case towards the deeper end. *LaSIE* is clearly deeper than template merging approaches such as FASTUS and shallower than full text understanding approaches like TACITUS. However, it is difficult to assess the relative deepness of what we are calling discourse-level processing in many of the MUC IE systems due to the extremely partial descriptions of this component given by the systems' creators. At least one MUC-6 system, the LOLITA system (Morgan *et al.* 1995), also uses a semantic network into which the semantic representations of input sentences are integrated. The NYU MUC-6 system produces a logical form following finite-state syntactic analysis and uses a classification hierarchy to assist in coreference resolution. The MITRE *Alembic* system also produces a propositional logical form following its phrase-finding syntactic analysis and then does limited forward inference using declaratively represented domain-specific inference rules to assist in discourse interpretation. The relative deepness of these approaches is hard to assess given the limited amount of effort that has so far gone into articulating and analysing these techniques.

While not particularly committed to *LaSIE*'s approach to syntactic analysis, we are committed to its approach to discourse interpretation. There are two principal reasons for this. First, we believe that obtaining high levels of precision in the IE task is of critical importance[1] and simply will not be achieved without attempting a deeper understanding of at least parts of the text. Such an understanding requires, given current theories of natural language understanding, the availability of general and domain specific world knowledge together with a reasoning mechanism that allows

[1] Cowie and Lehnert (1996) suggest that 90% precision will be necessary to satisfy information analysts. Current high precision scores in the MUC scenario extraction tasks are around 70%.

this knowledge to be used to resolve ambiguities in the initial text representation and to derive information implicit in the text. *LaSIE* provides mechanisms for this knowledge to be represented and applied.

Second, since discourse processing is not well understood, mechanisms which are both general and transparent are required so that a better understanding of the practical significance of various discourse phenomena for the IE task can be gained. The approach we have adopted to KR in *LaSIE* does allow us to address in a general way the use of world knowledge presupposition, coreference resolution, robust parsing and inference-driven derivation of template fills. This approach may not be as efficient as more restricted forms of processing, but does allow a wider range of phenomena and approaches to dealing with them to be modelled. It should result in deeper understanding which, once obtained, can be used to develop new, more accurate techniques which can then be optimised for efficient processing.

Insofar as results from the MUC evaluations can reliably be used to assess approaches, MUC-6 results show that the *LaSIE* approach can be as effective as shallower approaches. The *LaSIE* system described here obtained (jointly with NYU) the highest raw precision score in the scenario template filling task at MUC-6 and its combined precision and recall score for this task was only significantly poorer than one (of 10) systems (see section 7 below and DARPA (1995) for more details). We believe that its generality will, in the long run, lead to the significantly higher levels of precision which will be needed if IE systems are to break through their current barrier of mid-50's combined precision and recall scores and become a genuinely usable NL technology.

3 *LaSIE* system overview

LaSIE has been designed as a general purpose IE research system, initially geared towards, but not solely restricted to, carrying out the tasks specified in MUC-6: named entity recognition, coreference resolution, template element filling, and scenario template filling (see DARPA (1995) for further details of the task descriptions). In addition, the system can generate a brief natural language summary of any scenario it has detected in the text. All of these tasks are carried out by building a single rich model of the text – the discourse model – from which the various results are read off.

The high level structure of *LaSIE* is illustrated in figure 1. The system is a pipelined architecture which processes a text one sentence at a time and consists of three principal processing stages: lexical preprocessing, parsing plus semantic interpretation, and discourse interpretation. The overall contributions of these stages may be briefly described as follows:

Lexical preprocessing reads and tokenises the raw input text, tags the tokens with parts-of-speech, performs morphological analysis, performs phrasal matching against lists of proper names;

Parsing and semantic interpretation builds lexical and phrasal chart edges in a feature-based formalism then does two pass chart parsing, pass one with

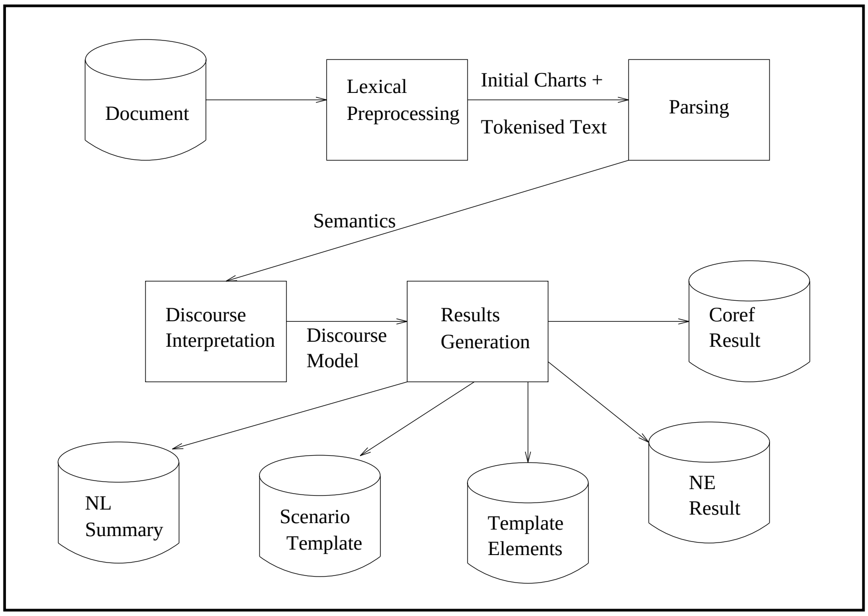

Fig. 1. *LaSIE* system architecture.

a special named entity grammar, pass two with a general grammar, and, after selecting a 'best parse', constructs a predicate-argument representation of the current sentence;

Discourse interpretation adds the information from the predicate-argument representation to a hierarchically structured semantic net which encodes the system's world model, adds additional information presupposed by the input, performs coreference resolution between new and existing instances in the world model, and adds any information consequent upon the new input.

For further details of the overall system see Gaizauskas *et al.* (1995).

4 Knowledge representation in *LaSIE*

Semantic interpretations are assigned to each sentence in a text during parsing using what is essentially a classical compositional approach – each phrase structure rule has a corresponding semantic rule which specifies how a semantic representation is to be built up (see, for example, Cann (1993)). The result is a *quasi-logical form* or QLF, much cruder than that used in Alshawi (1992), but sharing the characteristics of retaining various proximities to the surface form and of postponing some disambiguation, e.g. full analysis of quantifier scope and word sense disambiguation.

In this section we explain the central knowledge representation formalisms used in *LaSIE*: the QLF which is used as an initial representation of the meaning of individual sentences and the semantic net formalism, XI, used to represent both world models and discourse models.

4.1 QLF

Syntactically, QLF expressions are simply conjunctions of first order logical terms. The predicates in the QLF representation are either derived from the appropriate lexical morphological roots of head words, or come from a closed class of relation predicates that express modification or semantic role relations. To be more specific:

1. NPs lead to the introduction of a unary predicate whose functor is the morphological root of the head of the NP and whose argument is a unique index which serves as an identifier for the entity referred to, e.g. *company* will map to something like `company(e22)`.
   (a) Determiners such as *the*, *some* and *many* lead to the introduction of a `det` relation whose first argument is the index introduced by the head noun and whose second argument is the actual determiner. For example, *the company* becomes `company(e22), det(e22,the)`.
   (b) Cardinal quantifiers such as *three*, *10 million* lead to the introduction of a `count` relation. For example, *three companies* becomes `company(e22), count(e22,3)`.
   (c) Adjectives such as *big* and *old* are treated in the same way as determiners, by introducing an `adj` relation with the adjective itself as an argument. For example, *big company* becomes `company(e22), adj(e22,big)`.
   (d) Noun modifiers introduce new indices which are treated as the second argument to the `qual` relation, so that, for example, *computer company* becomes `computer(e21), company(e22), qual(e22,e21)`.
2. VPs lead to the introduction of a unary predicate whose functor is the morphological root of the head of the VP and whose argument is a unique index which serves as an identifier for the event referred to, e.g. *hired* will map to something like `hire(e34), time(e34,past)` [2].
3. Where complement structure has been recognised in the parser this is recorded in the QLF representation using binary relations of the form `lsubj(e34,e22)` (for logical subject), `lobj(e34,e25)` (for logical object) and, in the case of prepositional phrase complements, *prep*`(e34,e29)` (where *prep* is the actual preposition, e.g. `beside(e34,e29)`).

4.2 XI and world models

The discourse interpretation stage of *LaSIE* relies on an underlying 'world model', a declarative knowledge base that both contains general conceptual knowledge and serves as a frame upon which a discourse model for a multi-sentence text is built. This world model is expressed in the XI knowledge representation language (Gaizauskas 1995; Gaizauskas and Humphreys 1996*b*) which allows straightforward

[2] This treatment of VPs is in the tradition of Davidson (1967).

definitions of cross-classification hierarchies, the association of arbitrary attributes with classes or individuals, and the inheritance of these attributes by individuals. XI is an instance of the broad class of knowledge representation formalisms referred to as semantic networks – "a structure for representing knowledge as a pattern of interconnected nodes and arcs" (Sowa 1991, p. 1) – and bears similarities to many of the approaches discussed in Sowa (1991).

The *world model* consists of an *ontology* plus an associated *attribute knowledge base* [3]. The ontology is a directed acyclic graph with a unique top node. The nodes in the graph are either class nodes or instance nodes, with instance nodes occurring only as leaf nodes. Any non-leaf node may be subclassified across n dimensions, that is, may be at the root of n orthogonal trees. Each of these trees divides into mutually exclusive branches – thus, while any node may be immediately dominated by multiple nodes, no two of these nodes may be alternatives in the same classificatory dimension. For example, wines can be classified by colour and by nationality, so that a given wine can be white and French (dominated by the white and French nodes) but cannot be both red and white (dominated by two nodes in the same classificatory dimension).

In *LaSIE* the ontology consists mostly of classes or 'concepts' directly relevant to a specific template filling task. For MUC-6, the template filling tasks were to do with extracting information concerning *management succession events* from financial newswire articles. So, details about persons, posts, and organisations, and also about events involving persons leaving or taking up posts in organisations needed to be extracted. The ontology used for this domain required only 85 concept nodes though, as described below, new class nodes may be created dynamically during processing. Much of the initial ontology was derived directly from the MUC task specifications, ensuring that distinctions required in the template slots were reflected in the ontology. The manual development of the ontology for the MUC domain was not therefore a major task – see section 6.1 for more details. While an 85 node hierarchy may seem small, preliminary experiments which extended its size by two orders of magnitude have not led to improved extraction results (see section 6.2).

The *attribute knowledge base* is the set of attribute-value structures associated with the ontology. Attribute-value structures are just sets of `attribute:value` pairs where the value for an attribute may either be static, as in the pair `animate:yes`, which is associated with the `person` node, or dynamic, where the value is dependent on various conditions, the evaluation of which makes reference to other information in the model. Certain special attribute types, `presupposition` and `consequence`, may return values which are used at particular points to modify the current state of the model, as described in the following section.

The higher levels of the ontology for the MUC-6 management succession extraction task are illustrated graphically in figure 2, along with some very simple attribute-value structures. The very same ontology and associated attribute-value

[3] The distinction between ontology and attribute knowledge base was inspired by the similar distinction between ontology and generic knowledge base in Dahlgren (1988).

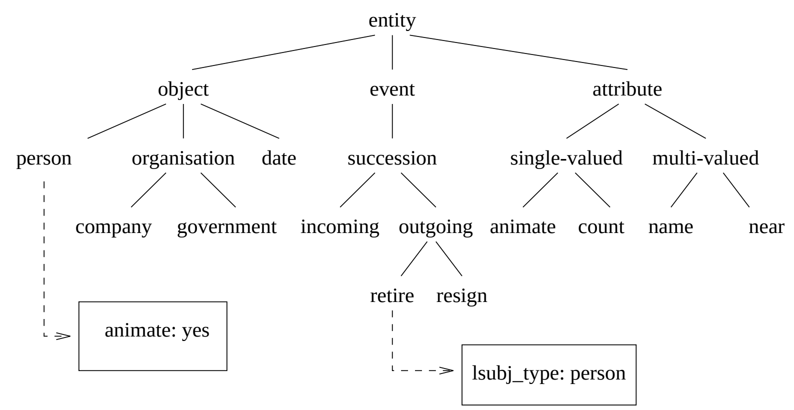

Fig. 2. A fragment of the *LaSIE* ontology and associated attribute knowledge base.

```
% Ontology
entity(X) ==> object(X) v event(X) v property(X).

object(X) ==> person(X) v organization(X) v date(X).
organization(X) ==>  company(X) v government(X).

event(X) ==> succession_event(X).
succession_event(X) ==> incoming_event(X) v outgoing_event(X).
outgoing_event(X) ==>  resign(X) v retire(X).

attribute(X) ==> single_valued(X) v multi_valued(X).
animate <-- single_valued(X).
count <-- single_valued(X).
near <-- multi_valued(X).
name <-- multi_valued(X).

% Attribute Knowledege Base
props(person(X),[
     animate(X,yes) ]).
props(name,[
     (presupposition(name(E,Name), [object(E)])) ]).
props(retire(X),[
     lsubj_type(X,person) ]).
```

Fig. 3. XI Definitions of the world model in figure 2.

structures are shown as defined in XI in figure 3 (note: `animate`, `count`, `near` and `name` are *instances* of the `single_valued` or `multi_valued` attributes – hence the differing form of their definitions in the ontology; all other terms in the ontology are *classes*).

4.3 Discourse models

The world model described above can be regarded as an empty shell or frame to which the semantic representation of a particular text is added, populating it with the classes and instances mentioned in the text. The world model which results is then a model specialised for the world as described by the current text; we refer to this specialised model as the *discourse model.*

The QLF representation produced by the parser for a single sentence is processed by adding its classes and instances, together with their attributes, to the discourse model which has been constructed so far for the text. One node is added to the ontology for each index in the QLF representation of the sentence: if the index is for a plural NP then a class node is added to the ontology; otherwise an instance node is added [4].

Figure 4 illustrates how a QLF representation is added to the world model, specialising it to convey the information supplied in a simple text. The resulting discourse model corresponds to the text:

Mr. Jones will retire.

to which the parser assigns the QLF:

```
person(e1), title(e1,'Mr.'), proper_name(e1,'Jones'),
lsubj(e4,e1),
retire(e4), mode(e4,will)
```

Indices which have their semantic class specified in the input (via unary predicates) are added directly to the discourse model, provided the class already exists as a node in the ontological hierarchy (e.g. `person(e1)`). If, however, the class specified in the input does not exist in the ontology (say, `penguin(e23)`), a new class node (`penguin`) is created dynamically and automatically placed within the existing hierarchy. Such additions permit, at least in a limited way, coreference relations to be identified between objects and events that are not modelled in the pre-defined, domain-specific world model created for the extraction task. The automatic placement of new nodes is done simplistically at present: the new nodes are added either immediately below the `object` node or the `event` node as determined by the presence or absence of event-like attributes, i.e. `time`, `lsubj` or `lobj`. None of these dynamically added classes is permanently retained following the processing of the text.

Attributes – binary predicates in the QLF in which the first argument is always an index – are added to the attribute-value structure(s) associated with indices occurring within them (e.g. `lsubj(e4,e1)` is added to the attribute-value structures of both `e4` and `e1`).

On the addition of each sentence's QLF representation, the model is checked for any inheritable attributes. In this case the `animate` and `lsubj_type` attributes

[4] There is at present no attempt made to distinguish distributed from collective readings of NPs, and hence no attempt to add classes of events for distributed readings of plural subject NPs, e.g. *Each man bought a house.*

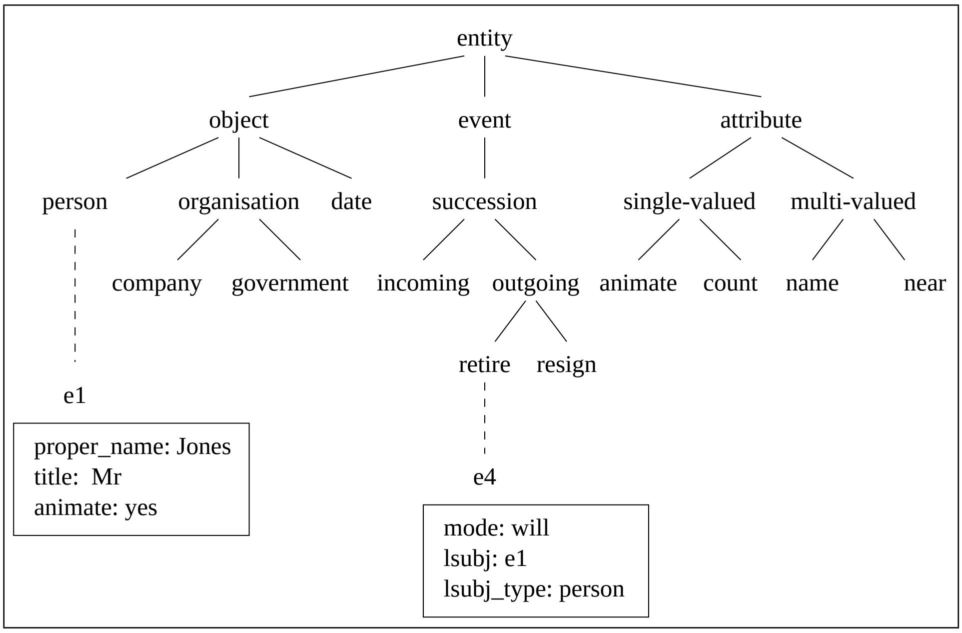

Fig. 4. A fragment of the *LaSIE* discourse model.

are inherited from the `person` and `retire` nodes respectively. If any `presupposition` attributes are inherited their values may be used to add (or remove) further information in the model. One use of this mechanism is to permit missing semantic class information for indices to be derived from type restrictions on attribute arguments. For instance, a `presupposition` attribute associated with the `proper_name` attribute node in the ontology records that this attribute holds only of entities of type `object`. When attempting to add say `proper_name(e1,Jones)` to the model, then in the absence of any more specific information about the type of `e1`, such as that `e1` is a `person`, `e1` will be added as an `object`. That is, the default semantic type of named entities is `object`, as opposed to, say, `event`. Such a derivation is not required in the example above because the presence of the title *Mr.* in the input permits the assignment of the semantic class `person` by the semantic interpretation rule associated with the NP grammar rule which says that an NP can consist of a person title followed by a proper name.

The use of English words to label nodes in the ontology (as shown in figures 2 and 4) is done purely for readability; nodes are actually labelled with unique identifiers and a mapping is carried out, via an index, between the predicates in the QLF and the appropriate ‘word sense’ or ‘concept’ nodes in the ontology. For this mapping to be done correctly, word sense disambiguation must be carried out. Or rather, it would need to be carried out if the ontology contained nodes corresponding to multiple senses of the same English word. But, for the small, domain-restricted, manually constructed ontologies we have used to date this problem has not arisen: all occurrences of a word are simply mapped onto the same node in the ontology. For words for which a single sense has been manually added to the ontology this approach amounts to an assumption of 'one sense per corpus'; for words for which no senses are recorded in the ontology, but for which nodes are added dynamically during processing (as explained above), this amounts to an assumption of 'one sense per discourse' (Gale *et al.* 1992). Clearly, our approach is naive. Nevertheless, it has not led to serious problems in the information extraction application.

It is worth noting that introducing a level of indirection between the predicates in the QLF (which, recall, are derived from morphological roots of words in the input) and concept nodes in the ontology provides a mechanism not only for dealing with multiple word senses and synonymy, but also for supporting multilinguality. The same concept node in the world model can be used as a target onto which synonymous lexical items from multiple source languages are mapped. This idea is pursued in Azzam *et al.* (1997) and Gaizauskas *et al.* (1997).

5 Knowledge use in *LaSIE*

The approach to KR outlined in the previous section allows a number of mechanisms to be applied during and after the construction of the discourse model.

5.1 Coreference resolution

After the QLF representation of a sentence has been added to the discourse model, all new indices (those introduced by that sentence) are compared with other indices in the model to determine whether any pair can be merged into a single class or instance, representing a coreference in the text. The comparison of indices is carried out in several stages:

1. new indices with `proper_name` attributes are compared with all existing indices with `proper_name` attributes, i.e. named entity coreferences can range over the whole text;
2. all new indices are compared with each other (intrasentential coreference resolution);
3. new indices introduced by pronouns are compared with existing indices from the current paragraph, i.e. pronoun coreferences are intra-paragraph only;
4. all other new indices are compared with existing indices from the current and previous paragraphs, i.e. all other coreferences are restricted to a span of two paragraphs.

Each comparison involves first determining if the indices' classes lie on the same branch in the ontology (type-compatibility). If not, then the indices are not considered further for coreference. If they are on the same branch then the attributes of the indices are compared to ensure there are no conflicts (attribute-compatibility). Certain attributes, such as `animate`, are defined in the ontology as taking single, fixed values for a particular index and so indices with conflicting values for these attributes cannot be the same. If such conflicts are discovered then the comparison is abandoned. The `proper_name` attribute is treated specially, using a semantic type-specific name match, to determine the compatibility of the newly input index's name with the known names of the existing index.

If no attribute conflicts are found between two indices, a similarity score is calculated based on the number of common attributes and on a semantic distance measure, determined simply in terms of the number of nodes in the path between them. After a newly input index has been compared with all others in a particular comparison set, it is merged in the world model with the index with the highest similarity score, if one exists.

Further details, and an evaluation, of this coreference algorithm may be found in Gaizauskas and Humphreys (1997).

5.2 Semantic parse extension

One of the perennial problems with IE systems that rely on constructing intermediate semantic representations by carrying out significant syntactic analysis is developing a sufficiently wide coverage grammar to process the range of real world text to which they are exposed. Our approach to this problem has been two-fold: on the one hand, we continue to refine our grammar and semantic interpretation mechanism; on the other hand, we store semantic role information about verbs and verb classes in the world model and then use this information, together with the coreference mechanism described above, to extend the frequently partial QLF representations that are produced when grammatical coverage is inadequate. In this section we describe the semantic parse extension mechanism as it illustrates the utility of the semantic net KR scheme in supporting robust analysis.

Where complement structure can be recognised by the parser, the appropriate relations between complement and verb are created during the semantic interpretation. Consider, for example:

Mr. R. Jones will succeed J. M. Greb.

This is easily parsed as

(*S* (*NP Mr. R. Jones*) (*VP* (*V will succeed*) (*NP J. M. Greb*)))

and is given the QLF interpretation:

```
succeed(e1), mode(e1,will),
person(e2), title(e2,'Mr.'), proper_name(e2,'R.Jones'),
lsubj(e1,e2),
person(e3), proper_name(e3,'J.M.Greb'), lobj(e1,e3)
```

The relations of the subject and object complements to the verb are encoded in the parse tree and translated into the predicate-argument representation by the standard semantic interpretation rules associated with the phrase structure grammar rules which generated the tree.

Now consider:

Mr. R. Jones who headed Foo Corp will succeed J. M. Greb.

and suppose, for the sake of illustration, that our grammar lacks the appropriate relative clause rule to generate a spanning parse. A parse consisting of a sequence of three sub-trees is produced:

(*NP Mr. R. Jones*) (*VP* (*V headed* (*NP Foo Corp*))) (*VP* (*V will succeed*) (*NP J. M. Greb*))

and is given the interpretation:

```
succeed(e1), mode(e1,will),
person(e2), title(e2,'Mr.'), proper_name(e2,'R.Jones'),
person(e3), proper_name(e3,'J.M.Greb'), lobj(e1,e3),
head(e4), time(e4,past),
company(e5), proper_name(e5,'Foo Corp'), lobj(e4,e5)
```

Notice that we have lost the `lsubj(e1,e2)` relation linking the subject and the main verb. It is at this point that the parse extension mechanism takes effect, relying on semantically typed role information associated with specific event nodes in the world model (this may be thought of as semantic type constraints on verb arguments in syntactic subcategorisation frames). When the QLF representation produced above is passed on to the discourse interpretation stage, `e1` is added to the world model as an instance of the class of `succeed` events. Assuming the appropriate semantic role information is present for the `succeed` class (or is inheritable by it from a superordinate class), then `e1` inherits the information that it has a logical subject role which must be filled by a `person` object. This information is used to hypothesise the existence of a new entity, say `e6`, about which nothing is known save that it is a person and that it is the logical subject of `e1`. At this point the interpretation looks like:

```
succeed(e1), mode(e1,will),
person(e2), title(e2,'Mr.'), proper_name(e2,'R.Jones'),
person(e3), proper_name(e3,'J.M.Greb'), lobj(e1,e3),
head(e4), time(e4,past),
company(e5), proper_name(e5,'Foo Corp'), lobj(e4,e5),
object(e6), animate(e6,yes), lsubj(e1,e6)
```

The intrasentential component of the coreference resolution mechanism (stage 2 of the algorithm described in the previous section) is then brought into play to attempt to unify the hypothesised entity with entities in the input. The knowledge that active verb subjects occur before their verbs in the same sentence is represented as an attribute of the `lsubj` node in the world model, and the coreference mechanism then attempts to unify `e6` with `e5` (*Foo Corp*) and with `e2` (*R. Jones*). The former is not type-compatible with `person` (i.e. it is not a super- or sub-ordinate class of `person`) and hence the unification fails. The latter, on the other hand, is unifiable with `e6` (again we assume the `animate` attribute is inheritable by `person`) and hence is unified with it, leading to the final interpretation:

```
succeed(e1), mode(e1,will),
person(e2), title(e2,'Mr.'), proper_name(e2,'R.Jones'),
lsubj(e1,e2),
person(e3), proper_name(e3,'J.M.Greb'), lobj(e1,e3),
head(e4), time(e4,past),
company(e5), proper_name(e5,'Foo Corp'), lobj(e4,e5)
```

```
<TEMPLATE-jnle.eg-1> :=
        DOC_NR:                 "jnle.eg"
        CONTENT:                <SUCCESSION_EVENT-jnle.eg-1>
<SUCCESSION_EVENT-jnle.eg-1> :=
        SUCCESSION_ORG:         <ORGANIZATION-jnle.eg-8>
        POST:                   "president"
        IN_AND_OUT:             <IN_AND_OUT-jnle.eg-6>
        VACANCY_REASON:         DEPART_WORKFORCE
<IN_AND_OUT-jnle.eg-6> :=
        IO_PERSON:              <PERSON-jnle.eg-2>
        NEW_STATUS:             OUT
        ON_THE_JOB:             YES
<ORGANIZATION-jnle.eg-8> :=
        ORG_NAME:               "Foo Corp."
        ORG_TYPE:               COMPANY
        ORG_LOCALE:             Los Angeles CITY
        ORG_COUNTRY:            United States
<PERSON-jnle.eg-2> :=
        PER_NAME:               "Jones"
        PER_TITLE:              "Mr."
```

Fig. 5. A sample MUC-6 template.

where the key `lsubj` relation has been restored. If the `head` event class also has a logical subject type constraint of `person` then a new `person` entity will be hypothesised as its subject and will also be coreferred with Jones, allowing us to add `lsubj(e4,e2)` to our discourse model.

Further details, and an evaluation, of this parse extension mechanism may be found in Gaizauskas and Humphreys (1996*a*).

5.3 Template filling

After all coreference and partial parse extension has been carried out for a text, any task specific `consequence` attributes of instances in the discourse model are evaluated. As described in the following section, the ontology will define a class of event types which may give rise to a template fill for the current task, and the `consequence` attributes of these classes will specify new attribute values to represent information from the discourse model that is relevant to the template.

Once all consequences have been expanded, the final state of the discourse model is checked for complete descriptions of scenario events. Each instance of the scenario event will be examined to ensure that all the information required to produce a complete template has been established, e.g. that each succession event instance has a related person, post and organisation. If any of the required information was not found in the current text, then the instance will be ignored. Otherwise a formatted template will be produced, with values for the various slot fills read from the attributes added during consequence expansion.

Figure 5 shows the MUC-6 management succession scenario template generated for the single sentence *Mr. Jones has retired as president of Foo Corp., the Los Angeles based firm.*

The verb *retire* gives rise to an instance of a `succession_event` in the discourse model. This instance is associated, possibly via the parse extension mechanism, with a person and a post in its `lsubj` and `lobj` relations. The consequence expansion stage then uses these relations to add scenario specific attributes (`succession_post`) and abstract objects (`in_and_out`) associated with the event. There may be consequences of these new additions, causing further attributes such as `io_person` to be added. Then, if a `succession_event` instance has all the required attributes (to allow a template to conform to the MUC-6 definitions), a template will be generated.

6 Knowledge acquisition in LaSIE

The preceding section has made it clear how the knowledge held in the *LaSIE* world model is used in coreference resolution, in robust interpretation, and in deriving template fills in the IE task. In this section we address the issue of how useful world models are to be constructed.

6.1 Manually constructed world models

To date, world models have been constructed by hand and their construction has been guided by the template defining the extraction task. Typically, *scenario extraction tasks*, in MUC parlance, involve filling a template that consists of an object defining a scenario event such as a management succession event, a labour negotiation event, a joint venture event, and so on. This high level object is made up of slots which define the participants in the event (companies, persons) and possibly other attributes of the event. These participants may in turn be objects in the template with slots whose values are to be extracted. So for instance a company object may have slots for `name`, `aliases`, `nationality`, and so on. Figure 6 shows the BNF definition of the MUC-6 management succession scenario template (as specified in Appendix F of DARPA (1995)).

Constructing a world model relevant to an extraction task such as this is done in stages:

1. The top level ontology is always the same and involves the top two levels shown in figure 2, i.e. the top level split in the ontology is always into `object`, `event`, and `attribute`. Furthermore, the `attribute` class is always split into `single_valued` and `multi_valued`. Various algorithms, such as the coreference algorithm, depend upon these high level divisions.
2. The next levels in the ontology are determined by the template.
   (a) Event objects in the template (such as `succession_event`) are made into `events` in the *LaSIE* ontology.
   (b) Non-event objects in the template (such as `person`, `organization` are made into `objects` in the *LaSIE* ontology.
   (c) All slots in objects in the template (such as `succession_org`) are made into `attributes` in the *LaSIE* ontology (binary predicates) whose first argument is the index of the `object` or `event` to which they pertain and whose second argument is the slot value.

```
<TEMPLATE> :=
        DOC_NR:                 "NUMBER" ^
        CONTENT:                <SUCCESSION_EVENT> *
<SUCCESSION_EVENT> :=
        SUCCESSION_ORG:         <ORGANIZATION> ^
        POST:                   "POSITION TITLE"|"no title" ^
        IN_AND_OUT:             <IN_AND_OUT> +
        VACANCY_REASON:         {DEPART_WORKFORCE, REASSIGNMENT,
                                 NEW_POST_CREATED, OTH_UNK} ^
<IN_AND_OUT> :=
        IO_PERSON:              <PERSON> ^
        NEW_STATUS:             {IN, IN_ACTING, OUT, OUT_ACTING} ^
        ON_THE_JOB:             {YES, NO, UNCLEAR}
        OTHER_ORG:              <ORGANIZATION> -
        REL_OTHER_ORG:          {SAME_ORG, RELATED_ORG, OUTSIDE_ORG} -
<ORGANIZATION> :=
        ORG_NAME:               "NAME" -
        ORG_ALIAS:              "ALIAS" *
        ORG_DESCRIPTOR:         "DESCRIPTOR" -
        ORG_TYPE:               {GOVERNMENT, COMPANY, OTHER} ^
        ORG_LOCALE:             LOCALE-STRING {{LOC_TYPE}} *
        ORG_COUNTRY:            NORMALIZED-COUNTRY-or-REGION STRING *
<PERSON-9301190125-6> :=
        PER_NAME:               "NAME" ^
        PER_ALIAS:              "ALIAS" *
        PER_TITLE:              "TITLE" *
```

Fig. 6. A sample MUC-6 template definition.

3. A development corpus and, if available, manually filled development templates are examined to determine which verbs give rise to the events and objects of interest in the scenario. These are added to the ontology beneath the template defined events and objects. If these events or objects themselves fall naturally into subclasses then these subclasses may be imported into the ontology (for instance, succession events can be divided into `incoming` and `outgoing` event classes and verbs tend to reflect this (e.g. *appoint*, *hire*, *name* versus *resign*, *quit*, *retire*).
4. After the `event` nodes are defined, appropriate `presupposition` attributes are associated with them so as to have the effect that when an event of this class is added to the discourse model, objects filling the roles of the event will be hypothesised. These hypothesised objects are then available to be bound to other objects by the coreference mechanism (see sections 5.1 and 5.2).
5. Scenario-specific `consequence` attributes, as described in section 5.3, are added to permit slot fills for template objects to be derived during the final template generation stage.

This approach to constructing a world model permits *LaSIE* to be ported reasonably rapidly to a new domain, given a template. The first two stages of the above procedure can be carried out mechanically provided a template has been specified; the third stage can also be carried out rapidly if manually filled development templates are supplied; the final two stages require more careful coding, but can be carried out quickly in a preliminary fashion then refined as time permits. *LaSIE* was ported to the management succession domain with approximately four person-weeks of effort.

6.2 Extending world models with WordNet

Preliminary investigations into the use of a general purpose lexical semantic network, WordNet (Miller 1990), have been carried out in an attempt to produce a more general world model which would cover more than the task domain required by a template definition. This was mainly expected to improve coreference and, as a consequence, improve performance in template filling. Coreferential phrases in the text will not always be between classes directly related to the scenario task, e.g. *Soon the chairman will be able to concentrate on his duties as rear commodore at the New York Yacht Club* where *rear commodore* is unlikely to be included in any financial world model. A more general model also has the potential to expand the range of lexical forms relevant to a particular task beyond the set found in a training corpus, thus allowing the production of templates from expressions not seen in training. For example, an instance of the verb *fire* in training data could be used to extend the world model to also cover all WordNet synonyms of *fire*, e.g. *dismiss*, *sack*, etc.

The WordNet hierarchy includes hypernym (superclass) and hyponym (subclass) relations between groups of synonymous lexical items, or 'synsets', which effectively represent concepts. A mechanism to translate these relationships into the XI notation has been developed (Poulos 1996) allowing the construction of a WordNet based world model which can be interchanged with a manually constructed one. To do this the problem of word sense disambiguation must be addressed, since WordNet includes a synset entry for each possible word sense. The current solution is to select a single sense per word manually, based on a small set of training texts. For each token found in WordNet, a sense selection is prompted for, then the hypernym and hyponym chains are followed to add the subtrees above and below the selected sense to the XI world model. An index is also generated to map each lexical item in each selected synset to the corresponding node in the XI representation. The mapping is used during the construction of the discourse model, when processing the root derived predicate names in the QLF representation.

Two alternative WordNet-derived world models were constructed: one in which senses were only selected for tokens considered to be relevant to the scenario task, and one which included, in addition, the most common senses for all other tokens in the training set. Five texts were used in training and a test set of 30 texts produced the following coreference results (using the MUC-6 scoring software):

```
Manually constructed WM:     Recall = 845/1627 = 52%
                          Precision = 845/1175 = 72%
WordNet-derived WM:          Recall = 850/1627 = 52%
(task specific senses)    Precision = 850/1250 = 68%
WordNet-derived WM:          Recall = 849/1627 = 52%
(task + general senses)   Precision = 849/1262 = 67%
```

The numbers of nodes in the three world models used above were, in order, 85, 1221 and 2943. The latter two would also have significantly greater lexical coverage due to the use of the mapping from root forms to concept nodes (i.e. the lexical coverage will be the sum of the number of words in each of the synsets corresponding to the nodes).

The results show that, within the current coreference algorithm at least, the availability of a much broader range of semantic classes in the ontology has very little effect on the number of coreferences that are identified correctly. However, the number proposed incorrectly increases noticeably. Given the small number of training texts, these results must be viewed as suggestive rather than conclusive and further experimentation is needed. The impact of these extended world models on the template filling task scores also needs to be established.

7 Conclusions

In the two MUC-6 evaluations of most relevance here (the coreference task and the scenario template filling task) the *LaSIE* system performed well. In the coreference task *LaSIE* scored 71% precision and 51% recall, placing second out of seven systems in precision (precision scores ranged from 44% to 72%) and fourth of seven systems in recall (recall scores ranged from 36% to 63%). In the scenario task *LaSIE* scored 73% precision and 37% recall, for a combined P & R score of 49%, placing overall 7th of 10 systems in raw combined P & R score, though statistical analysis of these results showed that only one system scored statistically significantly better. Precision scores ranged from 34% to 73%, recall scores from 32% to 58% and combined P & R scores from 33% to 56%. Note that *LaSIE* obtained the highest precision score in the scenario extraction task (joint highest, in fact). DARPA (1995) should be consulted for full details of scoring procedures and results.

Of course, scores from artificial exercises such as MUC-6 are not all that matters. Technology which is deployable in real applications must result, and there are positive indications that this is happening. Several companies (SRA, SRI) are marketing systems derived from their MUC entries; *LaSIE* is being adapted by industrial collaborators for use in processing police reports and teletext (Cunningham 1996); other industrial IE systems such as TREE (Ellman *et al.* 1997) and NaviLex (Pietrosanti and Graziadio 1997) which perform extraction tasks significantly simpler than the MUC-6 tasks are filling real world needs. Thus, there are good reasons for believing the MUC exercises, while artificial, have been appropriate tasks for stimulating the development of genuinely useful technology.

The *LaSIE* approach described above, relying on integrating logical representations of surface forms into a semantic network or 'world model', which supplies background knowledge about classes of entities and their attributes, offers a number of advantages. Specifically, it:

- provides a general, knowledge-directed mechanism for template objects to be hypothesised from a triggering word or phrase in the text and then unified with objects mentioned elsewhere in the text;
- allows a declarative, hierarchical description of types and attributes of things

in the world, which serves to constrain coreference without making it overly specific;

- allows a declarative statement of inference rules (at the most general level possible, because of the inheritance mechanism) for determining template-specific attributes in terms of what is known about other objects and their attributes;
- allows a separate declarative statement of a general world model and of a task-specific world model, yet allows them to be integrated easily for a specific IE task.

The ability to easily exchange world models has allowed experimentation with semantic type hierarchies extracted from existing resources like WordNet, as reported in section 6.2. However, as yet, the use of such general purpose large resources has not produced any clear benefits over small task-specific hierarchies, for IE performance at least.

Drawing any firm conclusions about the value of an approach from the relative MUC scores is extremely hazardous, given the intrusion of extraneous issues such as resource allocation (how much effort was expended in collection of the relevant patterns/template-fill inference rules) and software engineering expertise (how reliably was the selected approach implemented). Nevertheless, it seems safe to say that the MUC-6 results demonstrate our approach to be worthy of further investigation and its high precision score in the scenario extraction task is encouraging. With further effort expended in failure analysis of the MUC-6 results and in the investigation of (semi-) automatic acquisition of world models, we are confident that the *LaSIE* approach can yield considerably better results.

References

Aberdeen, J., Burger, J., Day, D., Hirschman, L., Robinson, P. and Vilain, M. (1995) MITRE: Description of the *Alembic* System Used for MUC-6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 141–156. Morgan Kaufmann.

Alshawi, H. (ed.) (1992) *The Core Language Engine.* MIT Press.

Appelt, D. E., Hobbs, J. R., Bear, J., Israel, D., Kameyama, M. and Tyson, M. (1993) Description of the JV-FASTUS system as used for (MUC-5). *Proceedings of the Fifth Message Understanding Conference (MUC-5)*, pp. 221–235. Morgan Kaufmann.

Appelt, D. E., Hobbs, J. R., Bear, J., Israel, D., Kameyama, M., Kehler, A., Martin, D., Myers, K. and Tyson, M. (1995) SRI International FASTUS system: MUC-6 Test Results and Analysis. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 237–248. Morgan Kaufmann.

Azzam, S., Humphreys, K., Gaizauskas, R., Cunningham, H. and Wilks, Y. (1997) Using a language independent domain model for multilingual information extraction. *Proceedings of the IJCAI-97 Workshop on Multilinguality in the Software Industry: the AI Contribution (MULSAIC-97).*

Cann, R. (1993) *Formal Semantics: An Introduction.* Cambridge University Press.

Cowie, J. and Lehnert, W. (1996) Information extraction. *Communications of the ACM*, **39**(1): 80–91.

Cunningham, H. (1996) AVENTINUS, ECRAN and GATE – Information Extraction and Language Engineering. *Proceedings of the Workshop on Multi-lingual Access to Textual Information (MAIN-96)*, Volterra, Italy.

Dahlgren, K. (1988) *Naive Semantics for Natural Language Understanding*. Kluwer Academic.

Davidson, D. (1967) The logical form of action sentences. In: N. Rescher (ed.), *The Logic of Decision and Action*. University of Pittsburgh Press, Pittsburgh.

Defense Advanced Research Projects Agency (DARPA). (1995) *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann.

Ellman, J., Somers, H., Nivre, J. and Multari, A. (1997) Foreign Language Information Extraction: An Application in the Employment Domain. *Natural Language Processing: Extracting Information for Business Needs*, pp. 77–89. Unicom Seminars Ltd., UK.

Gaizauskas, R. and Humphreys, K. (1997) Quantative evaluation of coreference algorithms in an information extraction system. In: S. Botley and T. McEnery (eds.), *Discourse Anaphora and Anaphora Resolution*. University College London Press (in press).

Gaizauskas, R. and Humphreys, K. (1996*a*) Using verb semantic role information to extend partial parses via a co-reference mechanism. In: J. Carroll (ed.), *Proceedings of the Workshop on Robust Parsing*, Prague, Czech Republic, pp. 103–113.

Gaizauskas, R. and Humphreys, K. (1996*b*) XI: A simple prolog-based language for cross-classification and inheritance. *Proceedings of the 7th International Conference on Artificial Intelligence:Methodology, Systems, Applications (AIMSA96)*, Sozopol, Bulgaria, pp. 86–95.

Gaizauskas, R., Humphreys, K., Azzam, S. and Wilks, Y. (1997) Conceptions vs. Lexicans: An architecture for multilingual information extraction. In: M.-T. Pazienza (ed.), *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*. Lecture Notes in Artificial Intelligence 1299, pp. 28–43. Springer-Verlag.

Gaizauskas, R., Wakao, T., Humphreys, K., Cunningham, H. and Wilks, Y. (1995) Description of the LaSIE system as used for MUC-6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 207–220. Morgan Kaufmann.

Gaizauskas, R. (1995) XI: A Knowledge Representation Language Based on Cross-Classification and Inheritance. *Technical Report CS-95-24*, Department of Computer Science, University of Sheffield.

Gale, W. A., Church, K. W. and Yarowsky, D. (1992) One sense per discourse. *Proceedings of the DARPA Speech and Natural Language Workshop*. Harriman, New York.

Grishman, R. and Sterling, J. (1993) Description of the Proteus system as used for MUC-5. *Proceedings of the Fifth Message Understanding Conference (MUC-5)*, pp. 181–194. Morgan Kaufmann.

Grishman, R. and Sundheim, B. (1996) Message understanding conference – 6: A brief history. *Proceedings of the 16th International Conference on Computational Linguistics*, Copenhagen, Denmark pp. 466–471.

Grishman, R. (1995) The NYU system for MUC-6 or where's the syntax. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 167–176. Morgan Kaufmann.

Hobbs, J. R., Appelt, D., Tyson, M., Bear, J. and Israel, D. (1992) Description of the FASTUS system as used for MUC-4. *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pp. 268–275. Morgan Kaufmann.

Hobbs, J. R. (1991) Description of the TACITUS system as used for MUC-3. *Proceedings of the Third Message Understanding Conference (MUC-3)*, pp. 200–206. Morgan Kaufmann.

Jacobs, P., Krupka, G. R. and Rau, L. (1995) Description of the SHOGUN System as used for MUC-5. *Proceedings of the Fifth Message Understanding Conference (MUC-5)*, pp. 109–120. Morgan Kaufmann.

Krupka, G. R., Jacobs, P., Rau, L., Childs, L. and Sider, I. (1992) GE NLTOOLSET: Description of the System as used for MUC-4. *Proceedings of the Fourth Message Understanding Conference (MUC-4)*, pp. 177–185. Morgan Kaufmann.

Krupka, G. R. (1995) Description of the SRA System as used for MUC-6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 221–236. Morgan Kaufmann.

Lin, D. (1995) Description of the PIE System as used for MUC-6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 113–126. Morgan Kaufmann.

Miller, G. A. (ed.). (1990) WordNet: An on-line lexical database. *International Journal of Lexicography*, **3**(4): 235–312.

Morgan, R., Garigliano, R., Callaghan, P., Poria, S., Smith, M., Urbanowicz, A., Collingham, R., Costantino, M. and Cooper, C. (1995) Description of the LOLITA System as used for MUC-6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 71–86. Morgan Kaufmann.

Pietrosanti, E. and Graziadio, B. (1997) Artificial Intelligence and Legal Text Management: Tools and Techniques for Intelligent Document Processing and Retrieval. *Natural Language Processing: Extracting Information for Business Needs*, pp. 277–291. Unicom Seminars Ltd., UK.

Poulos, P. (1996) Deriving XI models from WordNet. *Master's thesis*, Department of Computer Science, University of Sheffield.

Sowa, J. F. (ed.). (1991) *Principles of Semantic Networks: Explorations in the Representation of Knowledge*. Morgan Kaufmann.

Weischedel, R. (1995) Description of the PLUM System as used for MUC-6. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pp. 55–60. Morgan Kaufmann.