In *Proceedings of the 8th European Summer School in Logic, Language and Information (ESSLLI96) Workshop on Robust Parsing*, J. Carroll, ed. Prague, August, 1996, pp. 103-113.

# Using Verb Semantic Role Information to Extend Partial Parses via a Co-reference Mechanism

Robert Gaizauskas and Kevin Humphreys
Department of Computer Science
University of Sheffield
{robertg,kwh}@dcs.shef.ac.uk

**Abstract**

We describe a technique for the robust interpretation of newswire texts which uses semantic role information about verb complements together with a general co-reference mechanism to extend the constituent structure analysis produced by a partial parser. This technique has the advantage that failure to find a spanning parse of an entire sentence does not necessarily preclude correct semantic interpretation of, for example, key subject-verb-object relations. An information extraction system employing this technique has been evaluated in the Sixth Message Understanding Conference (MUC-6), and while the scoring protocols in that exercise do not allow a direct assessment of the technique, we can use them to obtain indirect performance measures which give some indication of how much the technique is contributing to overall system performance.

## 1 Introduction

### 1.1 Terms of Reference

In this paper we describe a technique for the robust interpretation of newswire texts which uses semantic role information about verb complements together with a general co-reference mechanism to extend the interpretation derived from the constituent structure analysis produced by a partial parser. We begin by explaining briefly what we mean by the terms 'parsing' and 'interpretation' and their relation in our approach.

We take *parsing* to be any activity whose goal is the production of a correct and complete syntactic description of a sentence, according to some syntactic theory (we do not try to define 'syntactic', but rely informally on commonly accepted usage). *Partial parsing* is the activity of producing correct, but not necessarily complete, syntactic descriptions, while *full parsing* aims to produce both correct and complete descriptions. *Interpretation* we take to be any activity whose goal is the production of a regularised or canonical representation of the text, sometimes viewed as (part of) the information content or the meaning representation of the text, which can support activities such as question answering, content extraction, summarisation, machine translation, and so on. Like parsing, interpretation may be full or partial, though it is perhaps harder to define *full* interpretation. Note that in these definitions there is no commitment as to what sort of information is used to derive a syntactic description or a meaning representation (so the activities are defined with respect to their outputs, not their inputs).

Our motivation has been the practical one of building a working system to derive interpretations of texts that are useful in information extraction – the task of automatically extracting pre-specified sorts of information from short, natural language texts, typically newswire articles, a task frequently characterised as 'template filling' and exemplified in the Message Understanding Conference (MUC) evaluations (see, e.g., [Adv93], [Adv95]). However, unlike some MUC participants, we do not chose to map surface forms directly into template structures (hence using the template structures as the interpretation language) but rather map surface forms into a logic-based interpretation from which we later fill the templates (so our approach has wider applicability than template filling alone). This mapping is carried out by first performing partial parsing with a context-free grammar augmented with feature-structure information, and then using the resulting partial parses to construct an interpretation in the conventional manner of formal semantics [DWP81] [Can93], based on the principle of compositionality and the rule-to-rule hypothesis. However, since the parse is fragmentary, so too is the resulting interpretation. To overcome this – and this is the key element of the approach described here – we use verb semantic role information to attempt to discover verb arguments that have not been properly linked to the verb by the parser and then link them in to the verbal predicate to produce the final sentence interpretation.

Thus, in our approach partial parsing is essential for interpretation, but full parsing is not (though of course the fuller the parse, the better). However, rather than concentrating directly on making the parsing more robust by increasing grammatical coverage, we have concentrated on making the interpretation robust. As a consequence, after interpretation is complete, and a richer complement structure has possibly been determined, a fuller parse could be constructed from the final interpretation together with the initial partial parse. However, since our goal is not parsing, but interpretation, we do not bother to construct it. Hence, somewhat paradoxically, while robust parsing, in the sense of parsing defined above, is a consequence of our approach, we do not produce any robust parses, only robust interpretations.

### 1.2 An Overview of the Approach

In conventional formal semantic interpretation of parse structures, syntactic structure serves to define function-argument relations in the resulting semantic representation. However, in the absence of a full parse, some other technique is necessary in order to discover the function-argument relations holding between unconnected sub-trees. Our approach is currently limited to discovering verb-complement connections and works as follows.

Complete bottom-up chart parsing is carried out using a grammar expressing only syntactic constraints. Then, from a preferred partial parse (sequence of sub-trees), a first-order predicate-argument representation is constructed in a conventional compositional semantic fashion. The result is a set of first order terms in which all tensed verbs are interpreted as referring to unique events and all noun phrases are interpreted as referring to unique objects. Relations between these events and objects may or may not have been identified from the syntactic constraints. The representation is added to a 'world model' which contains a specification of the relations required by particular event types. For example the event type *chase*, corresponding to the transitive verb, requires a logical subject and a logical object relation. If these relations were not identified during parsing, a new entity is hypothesised in the world model with the semantic constraints that the event type requires. For example, an entity hypothesised as the logical subject of a *chase* event might have the constraint that it is animate. A general coreference mechanism is then applied to the world model to attempt to unify entities with compatible semantic constraints, and in this way relations between events and objects which are missing from the input may be established and used to extend the semantic interpretations.

The remainder of paper recapitulates this overview in more detail. The approach is implemented in the Large Scale Information Extraction (LaSIE) system [1] [2] and we describe those aspects of the system that are relevant to the approach. Section 2 describes the parser and grammar used for partial parsing and the initial interpretations that are derived from the partial parse. Section 3 gives an overview of the 'discourse interpretation' module which serves the overall function of integrating the semantic representations of successive sentences in the text into a single representation of the whole text: it is this module that, amongst other things, carries out the task of extending the partial interpretations built by the parser. Section 4 gives more detail on the extension algorithm and section 5 reports the results of a preliminary, indirect evaluation of the approach. Section 6 contains a concluding discussion of the approach and describes possible future work.

## 2 Partial Parsing

The LaSIE parser is a simple bottom-up Prolog chart parser which uses unification-style, feature-based, context-free grammars. It is derived from the one described in [GM89]. The input to the parser is a sequence of lexical and multi-word chart edges, which are constructed by part-of-speech tagging (using the Brill tagger [Bri94]) and morphologically analysing a tokenised input stream, and then doing finite state pattern recognition against stored lists of proper names. As a result,

[1] LaSIE, as entered in MUC-6, is described in [GWH+95]. It has since been re-engineered to integrate it within the General Architecture for Text Engineering (GATE) [CGW95] wherein it is known as VIE. GATE/VIE is freely available for research purposes – see `http://www.dcs.shef.ac.uk/research/nlp/gate.html`.

[2] While this paper focusses on robust interpretation within LaSIE, this technique originated within the Portable Extendable Traffic Information Collator (POETIC) system [EGC+95], a system designed to extract information about road traffic incidents from police logs.

the only lexical information available as input to the parser is that derivable from the tagset (a slightly enriched version of the PTB tagset [MSM93]) and from the morphological analysis. No conventional lexical lookup takes place at this point. In particular, there is no subcategorisation information and there is no lexical semantic information.

Parsing takes place in two passes, each using a separate grammar. In the first pass a special named entity grammar is used, the sole purpose of which is to identify noun phrases relevant to the MUC-6 named entity task (places, persons, organisations, dates, and monetary and other numerical expressions). These constituents are then treated as unanalyzable during the second pass which uses a more general 'sentence' grammar.

The general grammar was derived from the Penn TreeBank-II (PTB-II) ([MSM93], [MKM+95]), which contains a large, skeletally parsed corpus of *Wall Street Journal* (WSJ) articles, and was therefore particularly suitable as a potential source for a robust grammar for processing American English financial newswire texts (the MUC-6 text genre). If a number of simplifying assumptions are made (see [GWH+95] for details), a context-free grammar can be extracted from the PTB-II WSJ corpus. However, doing so in a straightforward fashion leads to an unmanageably large grammar of approximately 17,500 rules. Of these rules, only a small number account for the majority of rule occurrences in the corpus. A pragmatic approach to obtaining a more manageable grammar, therefore, is to reduce the size of the grammar by removing the least frequently occurring rules [3]. Given the speed of our parser and the extension mechanism for partial interpretations, we opted for a base 112 rule grammar which comprises the rules accounting for 70% of the rule occurrences for each nonlexical category which is reachable in the reduced grammar. This grammar was then extended manually (to about 140 rules) to account for anomalies discovered during testing.

Semantic rules were assigned by hand to each of the grammar rules in both grammars. Predicate-argument structures (first order logical terms) are constructed entirely using Prolog term unification during parsing. The predicates in the predicate-argument representation are derived from the appropriate lexical morphological roots, and tense and number features are translated directly into the semantic representation where appropriate. All NPs and VPs lead to the introduction of a unique instance constant in the semantics which serves as an identifier for the object or event referred to in the text – e.g. *company* will map to something like `company(e22)` in the semantics and *hired* to `hire(e34), time(e34,past)`. Where complement structure has been recognised in the parser this is recorded in the semantic 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).

When parsing for a sentence is complete, the resultant chart is analyzed to extract the 'best parse'. Our algorithm for this is as follows: identify the set of syntactic categories for which useful self-contained semantics can be assigned – in our case S, NP, VP, and PP. Extract the set of shortest sequences of maximally spanning, non-overlapping edges of these categories. In the event of this set containing more than one member, pick one arbitrarily and designate it the 'best parse'. From the 'best parse' the associated semantics are extracted to be passed on to the discourse interpreter. Thus, the output of the parser is a set of predicate-argument structures, though in principle the full parse forest produced by the chart parser is available.

# 3 The Discourse Interpreter

## 3.1 The World Model

The discourse interpretation stage of LaSIE relies on an underlying 'world model', a declarative knowledge base that both contains the semantic role information that is used to extend the partial interpretations constructed by the parser 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 [Gai95] which allows straightforward definition of cross-classification hierarchies, the as-

[3] Charniak and ?? [Cha96] have recently described an approach in which they parse with all the grammar rules extracted from the PTB. We would certainly like to experiment with larger grammars, but do not necessarily expect to there to be any simple relation between grammar size and performance at some task, such as the MUC-6 template filling task.

sociation of arbitrary attributes with classes or individuals, and the inheritance of these attributes by individuals.

The *world model* consists of an *ontology* plus an associated *attribute knowledge base*. 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 these tasks contained only 80 concept nodes though, as described below, new nodes may be created dynamically during processing. The manual development of the ontology for the MUC domain was not therefore a major task. Much of the initial ontology was derived directly from the MUC task specifications, ensuring that distinctions required in the template slots were directly reflected in the ontology.

Associated with each node in the ontology is an attribute-value structure. Attributes are simple `attribute:value` pairs where the value may either be fixed, as in the attribute `animate:yes` which is associated with the `person` node, or where the value may be 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 set of attribute-value structures associated with the whole ontology is referred to as the *attribute knowledge base*.

The higher levels of the ontology for the MUC-6 management succession extraction task are illustrated in figure 1, along with some very simple attribute-value structures.

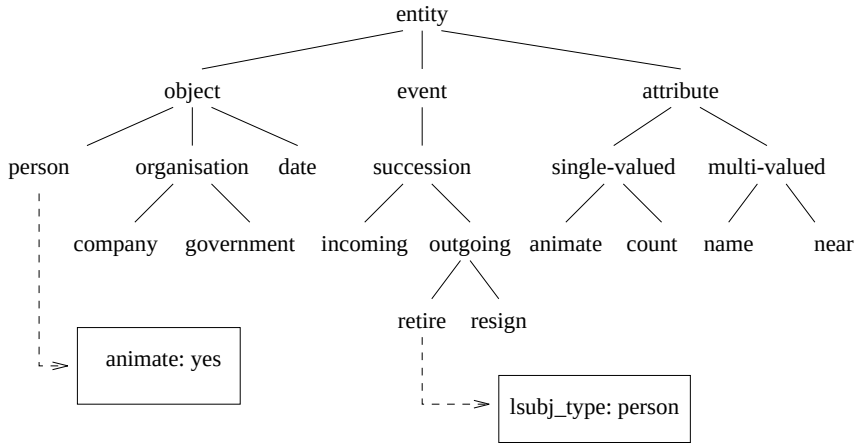


Figure 1: A Fragment of the LaSIE World Model and Associated Attribute Knowledge Base

## 3.2 The Discourse Model

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 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 semantic representation produced by the parser for a single sentence is processed by adding its instances, together with their attributes, to the discourse model which has been constructed so far for the text. Instances 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. `company(e1)`). If, however, the class specified in the input does not exist in the hierarchy (say, `penguin(e23)`) , a new class node (`penguin`) is created dynamically (`event` instances in the input are distinguished from `object` instances by the presence of event-like attributes, i.e. `time`, `lsubj` or `lobj`, thus allowing a crude, high level categorisation of unknown classes). Attributes – binary predicates in which the first argument is always an instance identifier

– are added to the attribute-value structure associated with instance identifiers occurring within them.

On each addition, the model is checked for any inheritable `presupposition` attributes, the values of which are used to add (or remove) further information in the model. One use of this mechanism is to permit missing semantic class information for instances to be derived from type restrictions on attribute arguments. For instance, a `presupposition` attribute associated with the node in the ontology corresponding to the `proper_name` attribute, records that this attribute holds only of entities of type `object`. When attempting to add say `proper_name(e3,Jones)` to the model, then in the absence of any more specific information about the type of `e3`, such as that `e3` is a `person`, `e3` will be added as an `object`. That is, the default semantic type of named entities is `object`, as opposed to, say, `event`.

Figure 2 illustrates how instances are added to the world model, specialising it to convey the information supplied in a specific text. The resulting discourse model corresponds to the text *Mr. Jones will retire.* The `animate` and `lsubj_type` attributes are assumed to have been inherited from the `person` and `retire` nodes respectively.

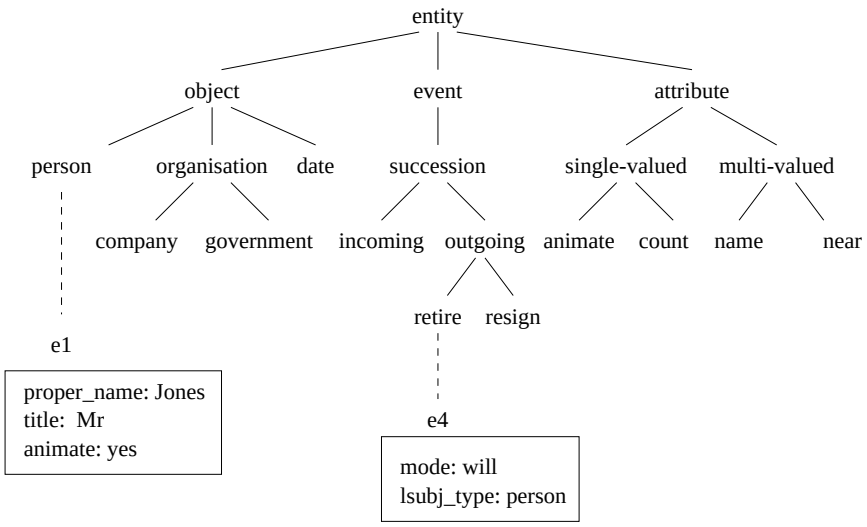


Figure 2: A Fragment of the LaSIE Discourse Model

It should be noted that the current system completely ignores the problem of word sense ambiguity. As described in the previous section, predicates in the semantic representation are derived directly from the morphological roots of words, and in the discourse model 'concept' nodes are identified simply by word names. Thus, for example, references in a text to a river bank and to a financial bank will lead to two instances of the same node `bank` being added to the discourse model. Clearly our approach is naive. Nevertheless, it has not led to serious problems in the information extraction application (perhaps because of the 'one sense per discourse' phenomenon [GCY92]). However, a more refined solution would be to treat nodes in the ontology as word senses and to introduce a procedure for mapping the predicates introduced during semantic interpretation to the appropriate word sense (i.e. to import the problem of word sense disambiguation).

## 3.3 The Coreference Algorithm

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

1. new instances with `proper_name` properties are compared with all existing instances with `proper_name` properties, i.e. named entity coreferences can range over the whole text;

2. all new instances are compared with each other (intrasentential coreference resolution);
3. new instances introduced by pronouns are compared with existing instances from the current paragraph, i.e. pronoun coreferences are intra-paragraph only;
4. all other new instances are compared with existing instances 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 instances' classes lie on the same branch in the ontology (type-compatibility). If not, then the instances are not considered further for coreference. If they are on the same branch then the attributes of the instances are compared to ensure there are no conflicts (attribute-compatibilty). Certain attributes, such as `animate`, are defined in the ontology as taking single, fixed values for a particular instance and so instances 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 instance's name with the known names of the existing instance.

If no attribute conflicts are found between two instances, 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 instance has been compared with all others in a particular comparison set, it is merged in the world model with the instance with the highest similarity score, if one exists.

Further details, and an evaluation, of this coreference algorithm may be found in [GHar].

# 4 The Extension Algorithm

Where complement structure has been 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 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 our grammar lacks the appropriate relative clause rule to generate a spanning parse. A parse consisting of a sequence of 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 predicate-argument representation produced above is passed on to the module that carries out this extension, `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 information indicating it has a logical subject role and that the role 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, if possible, the hypothesised entity with entities in the input. Guided by the knowledge that active verb subjects occur before their verbs in the same sentence, the coreference mechanism 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)
```

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 co-referred with Jones, allowing us to add `lsubj(e4,e2)` to our discourse model. It is worth observing that not only must entities be type-compatible in order to be resolved by the coreference mechanism, but their single-valued attributes must be compatible as well – so in this case had we specified only that the logical subject of `succeed` had the attribute `animate:yes` then assuming that `e5` would inherit `animate:no` from some superordinate class of which `company` is a subclass, this too would serve to block the unification with `e5` (the company) and enforce the unification with `e2` (the person).

Of course it will be objected[4] that any presumption that verbs *require* their complements to be of certain semantic types is far too strong and that this precludes extended, metaphorical, or less common usages (for example, in *The terraced houses stretching beyond Rob's succeeded each other with dismal regularity* the constraints cited above will have precisely the wrong effect). This is undoubtedly true, but to this objection we can supply two responses. First, in limited domains, within specific text genres, and for limited tasks such as information extraction, assumptions of semantic restrictions on verb arguments seem to be warranted (and may even help since we may usefully fail to interpret sentences which are not of interest for the task). Second, mechanisms of greater or lesser sophistication can easily be implemented in the current framework to *prefer*, in the tradition of [Wil75], certain semantic types to others, but to gradually relax restrictions until a best match is found (to say the mechanisms are easy to implement in this framework is not to say either that getting the right mechanism or acquiring the data on which it relies is straightforward).

[4]And has been, strenuously, by Yorick Wilks whom we thank for forcing us to make our views clearer here.

The example discussed in this section illustrates the fundamental idea behind the approach. In LaSIE, for reasons of efficiency, the extension mechanism is currently only applied to the event types defined as being relevant to the MUC-6 scenario template task (management succession events), and in the world model semantic constraints are only specified for the `lsubj`, `lobj` and certain prepositional phrase complementation relations. Currently only noun phrase object complements are considered, but there is no reason why the approach should not be extended to cover clausal object complements too.

As we indicated previously, after the extension process has been carried out it would be possible to build a more complete parse tree, but this is superfluous, since the aim is to derive an accurate interpretation.

# 5 Evaluation

At present there is no mechanism to directly evaluate the extended interpretations. However, of the four MUC-6 tasks, the main information extraction task, filling the scenario template, is principally concerned with a particular set of relationships between entities mentioned in a text. For the identification of *management succession events* the relevant relationships are generally described by verbs, and so accurate and complete semantic interpretations of verbs and their arguments will give most benefit to this task.

The evaluation of the scenario template task consisted of the automatic comparison of templates filled by the LaSIE system with manually filled templates produced by the MUC-6 organisation. The scoring software, also provided by the MUC-6 organisation, produces two main scores: Recall - the number of template slot values correctly filled by the system against the total number of slot values in the manually filled templates, and Precision - the number of slot values correctly filled by the system against the number of slot values filled by the system in total. System tuning therefore aims to maximise each score without penalising the other, and the scoring software produces a composite score, P&R, to reflect the extent to which this was achieved.

The overall scenario template scores for the LaSIE system, using the MUC-6 data, which consisted of 100 Wall Street Journal articles (approx. 40,000 words total), and with a slightly enhanced version of the system than that used in the official MUC-6 evaluation, were:[5]

| Recall | Precision | Combined P&R |
|---|---|---|
| 34% | 71% | 45.65% |

By selectively disabling particular features of the extension mechanism described above, then re-running and re-scoring on the MUC-6 data, the effectiveness of those features may be inferred from the change in the overall score. Although this is a very crude measure of the accuracy of the extended semantic interpretations, it does show considerable effects.

Disabling the extension mechanism altogether, so that the only relational information contributing to the final template slot values is that produced by the parser, the scores are as follows:

| Recall | Precision | Combined P&R |
|---|---|---|
| 9% | 83% | 16.72% |

Only around a quarter of the slot values correctly identified by the full system can therefore be derived from the semantic interpretations of the partial parses alone.

Selectively disabling the extension mechanism for the individual complement relations yields the following results.

No lsubj extensions (i.e. extend only lobj and PP complements):

| Recall | Precision | Combined P&R |
|---|---|---|
| 24% | 79% | 36.65% |

[5]These scores are a few percentage points down on the official MUC-6 scores for LaSIE because they were produced using a local installation of the scoring software running in fully automatic mode. The scorer used for the MUC-6 evaluation permitted some degree of manual intervention to judge certain borderline cases.

No lobj extensions (i.e. extend only lsubj and PP complements):

| Recall | Precision | Combined P&R |
|---|---|---|
| 28% | 72% | 39.96% |

No lsubj or lobj extensions (i.e. extend PP complements only):

| Recall | Precision | Combined P&R |
|---|---|---|
| 18% | 81% | 30.00% |

The effect of the extension mechanism for each relation can be summarised as follows[6]:

| Extension | Change in Recall | Change in Precision |
|---|---|---|
| lsubj | +10% | –9% |
| lobj | +6% | –2% |
| PP comp | +9% | –2% |

While all the extensions benefit the system's recall score, the fact that the precision score of the full system is very close to that with the addition of lobj relations disabled suggests that most of the lobj extensions carried out in the full system are correct. However, disabling the addition of lsubj relations increases precision noticeably, suggesting a relatively higher number of incorrect lsubj extensions are made in the full system.

The scores here only reflect extensions involving verbs which are relevant to the scenario task, but, since LaSIE was tuned to this task, it is these verbs which have their semantic restrictions specified in most detail. From a sample of 5 MUC-6 WSJ articles there were 142 event instances in the semantics produced by parsing. Of these 23 were defined as scenario event instances in the world model, and so could potentially contribute to the information extraction task output. The extension mechanism proposed the addition of 23 new relations involving these events: 6 lsubj, 7 lobj, 10 PP comp., i.e. an average of one extension per verb.

There are currently no detailed figures on the accuracy of the grammar with respect to verb complements, but the lexical coverage of the grammar can be calculated to be around 90%, i.e. semantic information was derived, where appropriate, from 90% of the tokens (including non-lexical tokens) in the input.

# 6 Concluding Remarks

## 6.1 Discussion

The principal strength of our approach is that lack of coverage in the grammar does not lead to a fatal inability to interpret. We still may not have a complete parse or a complete interpretation but we have a much better interpretation than the grammar alone allows.

Of course it may be argued that the same result is obtainable by extending the grammar or lexicon (depending on how lexicalised the approach) with the appropriate rules and subcategorisation frames. This is true, but, aside from the response that of the writing of grammars there is no end, there are further advantages to our approach. The surface order of phrases which identify the entities filling semantic roles need not be specified in our approach, nor need there be constraints about other surface material intervening in the complement structure. By contrast, purely syntactic treatments of complementation tend to prescribe fixed complement structures and attempt to enumerate all possible structures (and recent studies have revealed the difficulty of obtaining broad coverage subcategorisation dictionaries [BC96]). This makes them brittle since absence of the required subcategorisation pattern can lead to failure to produce any interpretation whatsoever. In our approach those complements that can be found are included in an interpretation, while those that cannot be found are simply left out. Indeed, even when complements are missing entirely

[6]No results are reported with the extension of PP complement relations disabled as a single class, because a number of separate prepositional relations are involved. The relative effect of all prepositional relations together was established by comparison with the results of disabling all extensions, whereas the relative effects of the lsubj and lobj relations were established by comparison with the results of PP extension only.

in the text (as in ellipsis or presupposition) our interpretive mechanism does not fail but makes the presupposed entity available for inter-sentential coreference and other inference requirements. However, there is not an either/or here. Better complement attachment through parsing with more accurate subcatgorisation information will lead our system to better performance; the technique we propose enhances partial parsing, it does not replace it.

### 6.2 Future Enhancements

Extending the range of verbs for which semantic restrictions are specified is the most obvious extension to the current implementation, and this information could be acquired in a (semi-)automated way from a text corpus. The semantic types of verb arguments found during parsing can be used to refine the semantic restrictions which are used for extending parses. For example, the most specific semantic class which covers all the encountered instances of a particular argument could be used, possibly along with frequency information, as the restriction on that argument. Restrictions gathered in this way would, of course, be quite domain specific, reflecting only certain usages of verbs. The world model in which the semantic types are classified would also need to be specified in sufficient detail for the domain in question.

The techniques described here as applying to verb semantic roles could also be used with noun modifiers. This may not assist the partial parse extension to the same degree as verb semantic roles, but would mean that prepositional phrase attachment ambiguities do not need to be handled by the grammar. Any syntactically problematic prepositional phrase could simply be left unattached in the semantics produced from the grammar, and then semantic restrictions brought to bear in the parse extension mechanism to attempt to determine the appropriate attachment position.

A more detailed evaluation of the effectiveness of the technique would also be beneficial. A manually produced predicate-argument analysis of a corpus could be used to evaluate both the semantic analysis from the grammar and the extension mechanism. Much more direct performance scores could then be produced, rather than the restricted and indirect MUC-6 measures presented here.

## Acknowledgements

This work has been supported by grants from the U.K. Department of Trade and Industry (Ref. YAE/8/5/1002) and the Engineering and Physical Science Research Council (# GR/K25267). The evaluation results reported here would not have been possible without the ARPA sponsorship of MUC-6 and the kind invitation of the MUC-6 organising committee for us to take part. The authors would like to thank Hamish Cunningham and Takahiro Wakao for their extensive contributions to the LaSIE system and Mark Hepple for optimising the chart parser. Our general thanks to members of the Sheffield NLP group who read and commented on a draft of this paper. Finally, thanks to Roger Evans, of the Information Technology Research Institute, Brighton University, with whom the first author discussed earlier versions of these ideas.

## References


[Adv93] Advanced Research Projects Agency. *Proceedings of the Fifth Message Understanding Conference (MUC-5)*. Morgan Kaufmann, 1993.

[Adv95] Advanced Research Projects Agency. *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, 1995.

[BC96] T. Briscoe and J. Carroll. Automatic extraction of subcategorization from corpora. Submitted to Fourth Workshop on Very Large Corpora, 1996.

[Bri94] E. Brill. Some advances in transformation-based part of speech tagging. In *AAAI*, 1994.

[Can93] R. Cann. *Formal Semantics: An Introduction*. Cambridge University Press, 1993.

[CGW95] H. Cunningham, R. Gaizauskas, and Y. Wilks. A General Architecture for Text Engineering (GATE) – a new approach to Language Engineering R & D. Technical Report CS-95-21, Department of Computer Science, University of Sheffield, 1995. Also available as `http://xxx.lanl.gov/cmp-lg/9601009`.

[Cha96] E. Charniak. Tree-bank Grammars. Technical Report CS-96-02, Department of Computer Science, Brown University, 1996.

[DWP81] D.R. Dowty, R.E. Wall, and P.S. Peters. *Introduction to Montague Semantics*. Reidel, Dordrecht, 1981.

[EGC+95] R. Evans, R. Gaizauskas, L. Cahill, J. Walker, J. Richardson, and A. Dixon. Poetic: A system for gathering and disseminating traffic information. *Natural Language Engineering*, 1(4), 1995.

[Gai95] R. Gaizauskas. XI: A knowledge representation language based on cross-classification and inheritance. Technical Report CS-95-24, Department of Computer Science, University of Sheffield, 1995.

[GCY92] W. Gale, K. Church, and D. Yarowsky. One sense per discourse. In *DARPA Speech and Natural Language Workshop*, 1992.

[GHar] R. Gaizauskas and K. Humphreys. Quantitative evaluation of coreference algorithms in an information extraction system. In S. Botley and T. McEnery, editors, *Corpus-based and Computational Approaches to Discourse Anaphora*. UCL Press and CCI, UMIST, Lancaster, U.K., To appear.

[GM89] G. Gazdar and C. Mellish. *Natural Language Processing in Prolog*. Addison-Wesley, Wokingham, 1989.

[GWH+95] R. Gaizauskas, T. Wakao, K. Humphreys, H. Cunningham, and Y. Wilks. University of sheffield: Description of the LaSIE system as used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*. Morgan Kaufmann, 1995.

[MKM+95] M. Marcus, G. Kim, M.A. Marcinkiewicz, R. MacIntyre, A. Bies, M. Ferguson, K.Katz, and B. Schasberger. The penn treebank: Annotating predicate argument structure. Distributed on The Penn Treebank Release 2 CD-ROM by the Linguisitc Data Consortium, 1995.

[MSM93] M.P. Marcus, B. Santorini, and M.A. Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

[Wil75] Y. Wilks. A preferential, pattern-seeking, semantics for natural language. *Artificial Intelligence*, 6:53–74, 1975.