The text mapping and inference rule generation problems in text comprehension: Evaluating a memory-based account

Simon Dennis
Walter Kintsch
University of Colorado

Introduction

Words and concepts are obviously useful units for the analysis of language. A number of representational mechanisms have been proposed to capture their meaning, such as feature systems (Katz & Fodor, 1963; E. E. Smith, Shoben, & Rips, 1974) or semantic networks (Collins & Quillian, 1969). There are, however, other levels of analysis that have proven useful in the study of language, both finer grained units (such as syllables or phonemes) and coarser grained ones. A particularly important level of analysis has been the propositional level. The term derives from logic, but has been widely used in linguistics and psychology to designate units consisting of one or more concepts that functions as arguments (with specific semantic roles) and a relational term. In one form or another such units played an important role in the work of linguists like Fillmore (1968), Bierwisch (1969), or van Dijk (1972). Psychologists, too, felt a need for units of this kind, either talking informally about "idea units," or using propositional analysis following the lead of linguists (e.g., Kintsch, 1974). Indeed, the addition of propositional analysis to the toolbox of psychologists played a major role in their efforts to understand discourse comprehension and to model it.

The problem of how to map texts into propositional structures has never been solved adequately, however. What is needed is a formal way of mapping a text into a propositional structure that represents the meaning of the text, not only what is explicitly stated in the text but also the knowledge activated by the text and the inferences involved in constructing a situation model for the text. Twenty years ago, many researchers were confident that such a system could soon be designed. In the meantime, a certain amount of progress could be made by hand coding texts into propositions (e.g., Kintsch, 1974, 1998). For the purpose of constructing experimental materials for discourse studies, hand coding proved to be adequate, as long as the texts were short and ambiguities could be avoided. But it is certainly not a satisfactory solution, and the algorithm to derive propositions and inferences from texts in an automatic and fully principled way has yet to be devised. There has been some partial progress along these lines (e.g., Blaheta & Charniak, 2000; Durbin, Earwood, & Golden, 2000; Gildea & Jurafsky, 2002; Lin, 1998 and see Appendix A), but the problem has not been solved, and is beginning to appear intractable.

Propositions, in the sense used here (e.g., Kintsch, 1998), not the original logical meaning, are the product of the analysis of language as practiced by philosophers and linguists over a long period of time. The concept is clearly defined and well motivated – but it is just one possible product of linguistic analysis, and there may be other ways to arrive at workable multi-word units that function much like propositions. Propositions have proven their usefulness in discourse studies, and there are strong arguments that some such unit at a level higher than the word is needed for the study of language comprehension (see Kintsch, 1998, Section 2.2, for a summary of those arguments), but the failure so far to solve the mapping problem for propositions makes imperative the search for alternatives. Are there higher-order units that are not defined in the same way as propositions, but that could do the same work in discourse models that propositions do? What we need is some way to tell that the words in "Sampras outgunned Agassi" belong together, and to represent the relationship of this unit to other units in complex sentences, such as "Sampras outgunned an injured Agassi in a five-set struggle."

In this paper, we investigate the Syntagmatic Paradigmatic model (SP, Dennis, in press-a; Dennis, in press-b) as a candidate mechanism for solving the text mapping problem. The SP model is a memory-based approach that generates structured knowledge representations automatically from a large text corpus. While these representations have some properties that are similar to propositions they are also different in important respects. One of the objectives of this paper is to examine the extent to which they are capable of supporting and extending the role that propositions play in existing accounts of text comprehension.

The relational representations provided by the SP model function much like propositions, but have some important advantages over propositional representations: texts can be mapped into these units automatically and on a large scale. The model achieves this by shifting from an intentional semantic representation to an extensional representation.

Intentional versus Extensional Semantics

In systems that employ intentional semantics, such as propositional analysis, the meanings of representations are defined by their intended use and have no inherent substructure. For instance, the statement "Sampras outguns Agassi." might be represented propositionally as (OUTGUN, SAMPRASS, AGASSI) where the relation OUTGUN is specified between the two arguments SAMPRAS and AGASSI, which have specific semantic roles:

Sampras: *Winner*
Agassi: *Loser*

However, the names of the roles are completely arbitrary (often they are called *Agent* and *Patient* instead) and carry representational content only by virtue of the inference system in which they are embedded.

Now contrast the above situation with an alternative *extensional* representation of "Sampras outguns Agassi", in which roles are defined by enumerating exemplars, as follows:

Sampras: Kuerten, Hewitt
Agassi: Roddick, Costa

The winner role is represented by the distributed pattern of Kuerten and Hewitt, words that have been chosen because they are the names of people who have filled the "X" slot in a sentence like "X outguns Y" within the experience of the system. Similarly, Roddick and Costa are the names of people that have filled the "Y" slot in such a sentence and form a distributed representation of the loser role. Note the issue is not just a matter of distributed versus symbolic representation. The tensor product representation employed in the STAR model (Halford et al., 1994) of analogical reasoning uses distributed representations of the fillers, but assigns a unique rank to each role and thus forms an intentional scheme. By contrast, the temporal binding mechanism proposed by Hummel & Holyoak (1997) allows for both distributed filler and role vectors and hence could implement extensional semantics.

The use of extensional semantics, of this kind, has a number of advantages. First, defining a mapping from raw sentences to extensional meaning representations is much easier than defining a mapping to intentional representations because it is now only necessary to align sentence exemplars from a corpus with the target sentence. The difficult task of either defining or inducing semantic roles is avoided.

Second, because the role is now represented by a distributed pattern it is possible for a single role vector to simultaneously represent roles at different levels of granularity. The pattern {Kuerten, Hewitt} could be thought of as a proto-agent, an agent, a winner, and a winner of a tennis match simultaneously. The role vectors can be determined from a corpus during processing, and no commitment to an a priori level of role description is necessary.

Third, extensional representations carry content by virtue of the other locations in the experience of the system where those symbols have occurred. That is, the systematic history of the comprehender grounds the representation. For instance, we might expect systematic overlap between the winner role and person-who-is-wealthy role because some subset of {Kuerten, Hewitt} may also have occurred in an utterance such as “X is wealthy”. These contingencies occur as a natural consequence of the causality being described by the corpus. We will call this type of implicit inference, *inference by coincidence*, and as we will see in subsequent sections the performance of the model is in large part due to this emergent property.

Inferences

Inference generation presents even more serious problems to propositional systems than text mapping. The mental representation of a text that comprehenders construct consists not only of the information explicitly expressed by the text, but also of the knowledge activated by the text and the inferences generated that fill in gaps in the text and link prior knowledge to new information. Currently, the process of formulating inference rules relies even more heavily on the intuition of the theorist, and assessing the adequacy of a modeling effort often hinges on the extent to which these intuitions are shared in the research community. Not only do inference rules need to be formal and fully automatic, they should also be an inherent component of the comprehension system, not some sort of add-on. There are important distinctions to be made between the mental representation of the text itself (textbase) and the situation model, which incorporates the comprehender’s goals, interests, and prior knowledge, but a process model of comprehension must account for both.

The dominant questions about inferences in the literature on discourse comprehension have focused on which inferences are made and when they are made, during comprehension or on demand at some later time. Inferences have been classified along the dimensions of automatic vs. controlled and retrieval vs. generation (Kintsch, 1998), and a great deal of discussion has been devoted to the question of when inferences are made (Graesser, Singer, & Trabasso, 1994; McKoon & Ratcliff, 1992). The SP model suggests a different approach to the study of inferences based on inference mechanisms that cuts across these distinctions and complements them. Specifically, the SP model addresses the following inference mechanisms:

1) Inference by coincidence. This is not a process per se. Rather, the inference occurs as an emergent property of the model as the consequence of role-filler overlap between proposition-like traces.
2) Linguistic pattern completion. A person follows a specific linguistic pattern by filling in the values of variables. This is what typically happens in formal logic, but it can also happen in much less formal ways as well, e.g., the rule “I before E except after C” requires the pattern to be instantiated in either inner or outer speech.
3) Inference by relational retrieval (analogical inference). Borrowing from the STAR model of analogical reasoning (Halford et al., 1994), the SP model proposes that analogical inference occurs as a consequence of doing retrieval over structured (proposition-like) units.

In a later section on inference in the SP model, we will give specific examples of how the model implements these forms of inference. For completeness we note two additional inference mechanisms, not considered here:

4) Associative inference, which links causally or temporally, connected propositions by means of constraint satisfaction – e.g., Golden & Rumelhart (1993), Frank, Koppen, Noordman & Vonk (2003).
5) Perceptual simulation, which generates a visual or auditory image of a situation. This process allows the constraints embodied in that system (e.g. spatial or form constraints) to act, forming a coherent image from which the facts of interest can be read off.

Instance-based Models of Cognition

The SP model is a model of memory; with appropriate elaborations that allow it to deal with the complexity of language. The SP model belongs to the class of models that assume that the memory system stores traces that are representative of the instances that an organism has experienced. Instance-based models of memory were introduced in response to data showing that people retain a surprising amount of detail during memory experiments (Hintzman, 1984). For instance, Tulving and Thompson (1973) demonstrated that even when subjects were unable to recognize an item as having appeared on a list, they were nevertheless able to recall it when given appropriate cues. This indicates that the fallibility of human memory can not be taken as direct evidence of what is stored. Rather, factors such interference and cue reinstatement play pivotal roles in performance. Furthermore, Hintzman (1988) showed that people were able to identify not only the frequency with which items appeared in memory, but also the approximate locations of each of the occasions on which they were presented, something that would be impossible unless they retained some information about specific instances. As a consequence of this and a large body of additional data that has been collected in the last two decades many of the most influential models in memory (Shiffrin & Steyvers, 1997), learning (Logan, 1988), decision-making (Dougherty, 1999), phonology (Nakisa & Plunkett, 1998), lexical access (Goldinger, 1998) and categorization (Nosofsky, 1986; J. D. Smith & Minda, 1998) are instance-based.

In our case, since the SP model is concerned with sentence comprehension and memory, the assumption is that sentence memory consists of a large sample of the sentences a person has encountered. While instance-based models have not to date played a significant role in psychological models of sentence processing they have become popular in the computational linguistics literature (Daelesmans, 1999). Thus, an instance model of sentence comprehension like the SP model cannot be rejected a priori as irrelevant psychologically but needs to be evaluated, like any model, by the success and range of the predictions it generates.

In this paper, we first sketch the SP model, and then describe how it addresses the text mapping and inference generation problems.

The Syntagmatic Paradigmatic Model

The SP model has been designed as a model of verbal cognition. It has been used to account for a number of phenomena including long term grammatical dependencies and systematicity (Dennis, in press-a), the extraction of statistical lexical information (syntactic, semantic and associative) from corpora (Dennis, 2003b), sentence priming (Harrington & Dennis, 2003), verbal categorization and property judgment tasks (Dennis, in press-a), serial recall (Dennis, 2003a), and relational extraction and inference (Dennis, in press-a, in press-b). In this section, we give a brief overview of the SP model. More complete descriptions, including the mathematical foundations, are provided by Dennis (in press-a; in press-b).

In the SP model, sentence processing is characterized as the retrieval of associative constraints from sequential and relational long-term memory and the resolution of these constraints in working memory. Sequential long-term memory contains the sentences from a corpus. Relational long-term memory contains the extensional representations of the same sentences (see Figure 1).

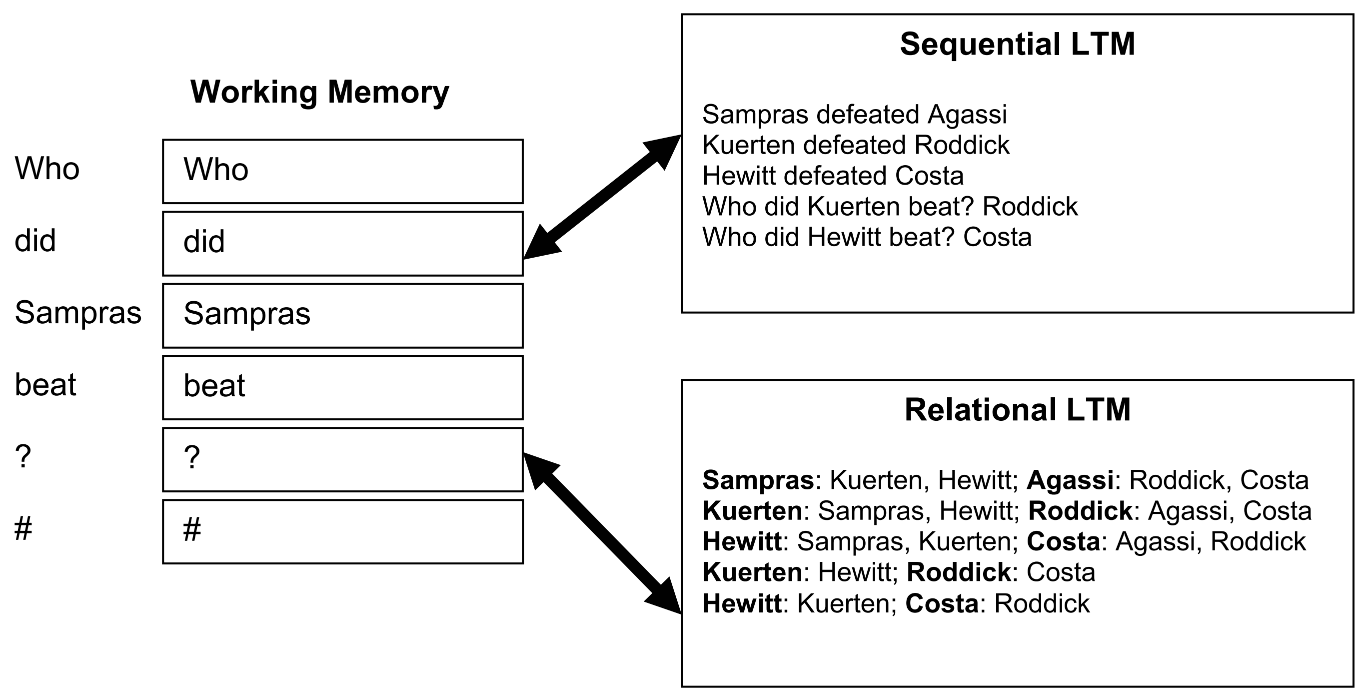

Figure 1: The Syntagmatic Paradigmatic (SP) architecture. The “#” symbol indicates an empty slot. Ultimately, it will contain the answer to the question.

Creating an interpretation of a sentence/utterance involves the following steps:

Sequential Retrieval: The current sequence of input words is used to probe sequential memory for traces containing similar sequences of words. In the example, traces four and five; "Who did Kuerten beat? Roddick" and "Who did Hewitt beat? Costa"; are the closest matches to the target sentence "Who did Sampras beat? #" and are assigned high probabilities (see Figure 2).

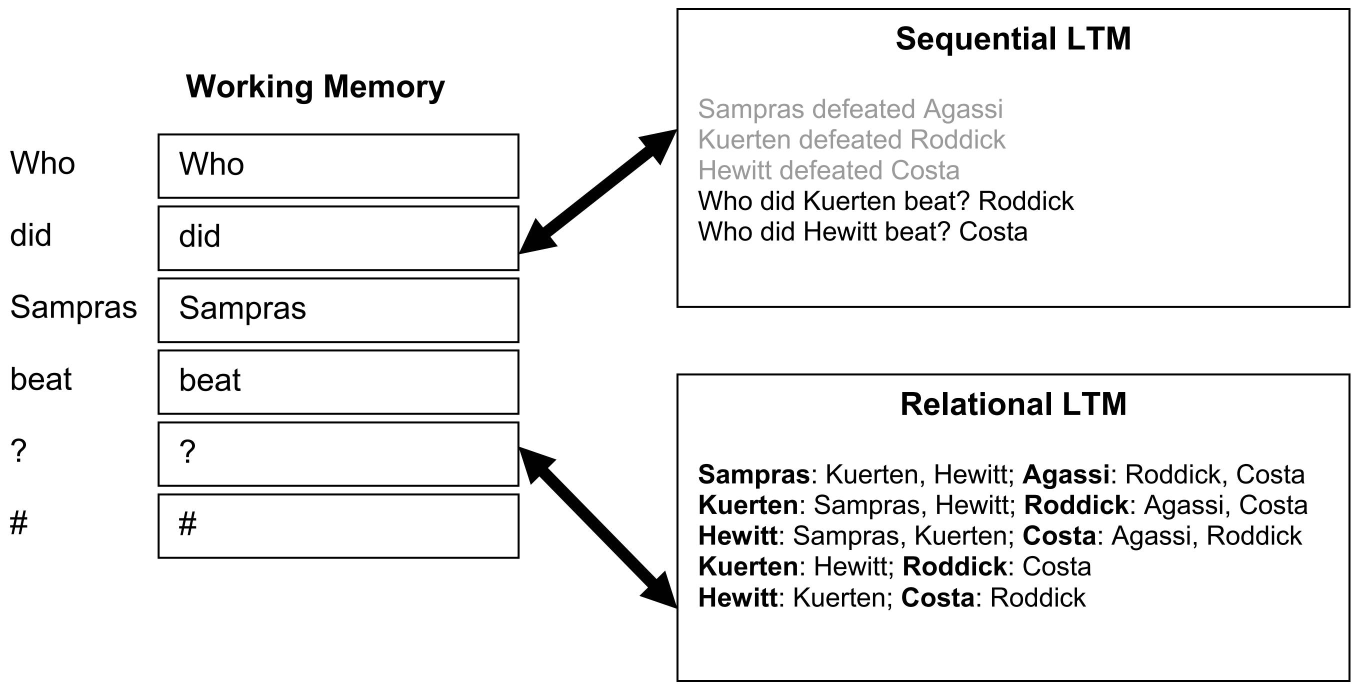

Figure 2: Sequential Retrieval: The traces "Who did Kuerten beat? Roddick" and "Who did Hewitt beat? Costa" are most similar to the input sentence "Who did Sampras beat? #" and are retrieved from sequential long term memory.

Sequential Resolution: The retrieved sequences are then aligned with the target sentence to determine the appropriate set of substitutions for each word (see Figure 3). Note that the slot adjacent to the "#" symbol contains the pattern {Costa, Roddick}. This pattern represents the role that the answer to the question must fill (i.e. the answer is the loser).

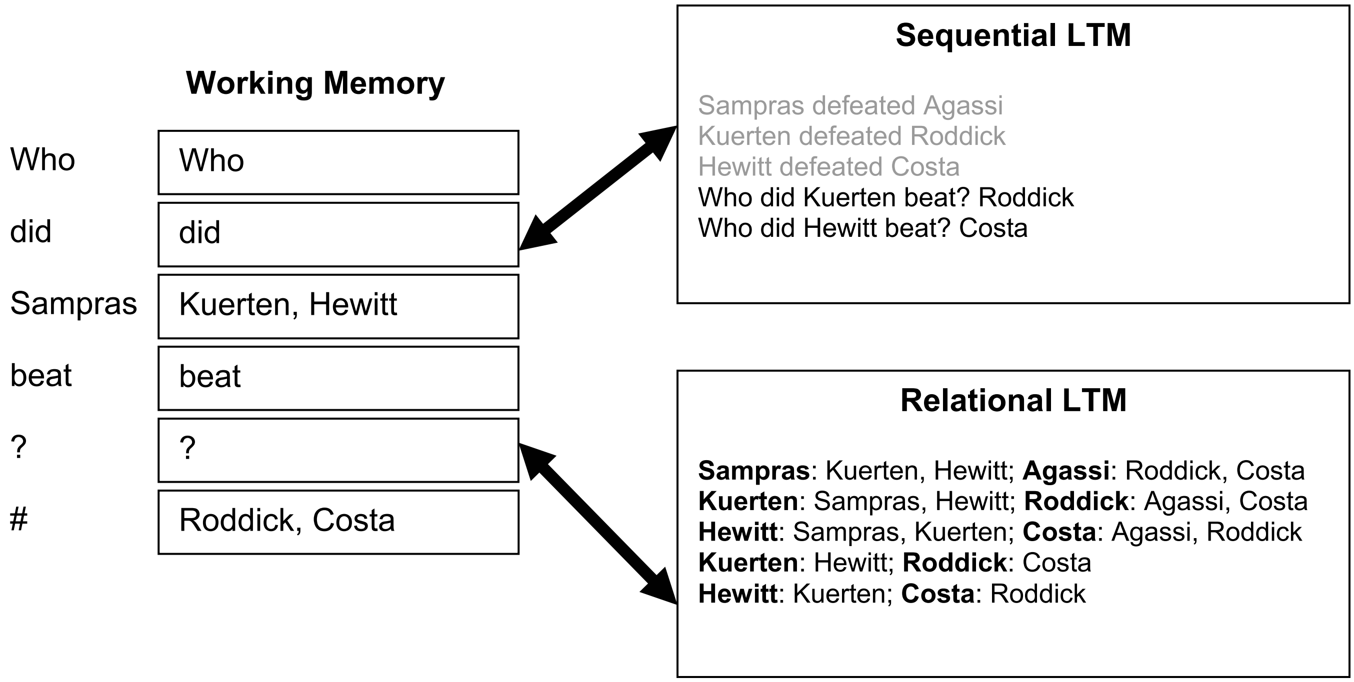

Figure 3: Sequential Resolution: Kuerten and Hewitt align with Sampras and Roddick and Costa align with the answer slot ("#").

Relational Retrieval: The bindings of input words to their corresponding role vectors (the relational representation of the target sentence) are then used to probe relational long-term memory. In this case, trace one is favoured as it involves similar role filler bindings. That is, it contains a binding of Sampras onto the {Kuerten, Hewitt} pattern and it also contains the {Roddick, Costa} pattern. Despite the fact that "Sampras defeated Agassi" has a different surface form than "Who did Sampras beat ? #" it contains similar relational information and consequently has a high retrieval probability.

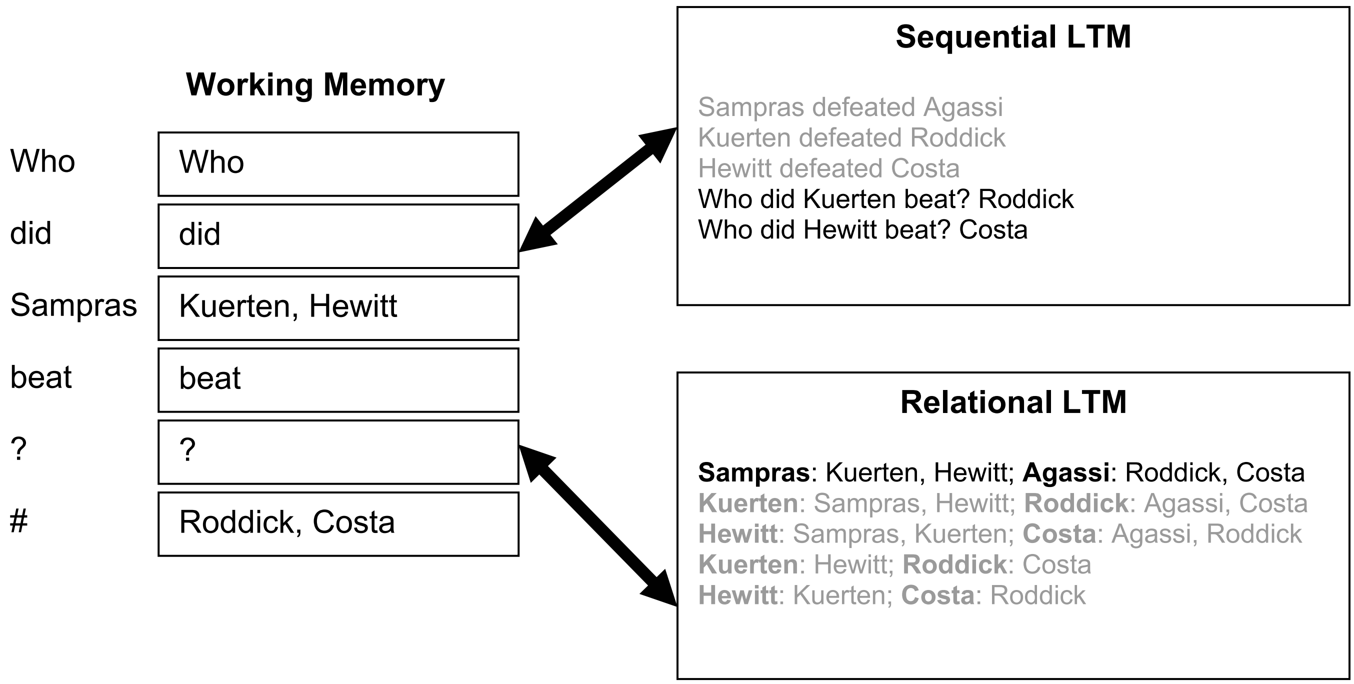

Figure 4: Relational Retrieval: The first relational trace is retrieved as it contains similar role filler bindings.

Relational Resolution: Finally, the paradigmatic associations in the retrieved relational traces are used to update working memory. In the relational trace for “Sampras defeated Agassi”, “Agassi” is bound to the {Roddick, Costa} pattern. Consequently, there is a strong probability that “Agassi” should align with the “#” symbol which as a consequence of sequential retrieval is also aligned with the {Roddick, Costa} pattern. Note that the model has now answered the question - it was Agassi who was beaten by Sampras (see Figure 5).

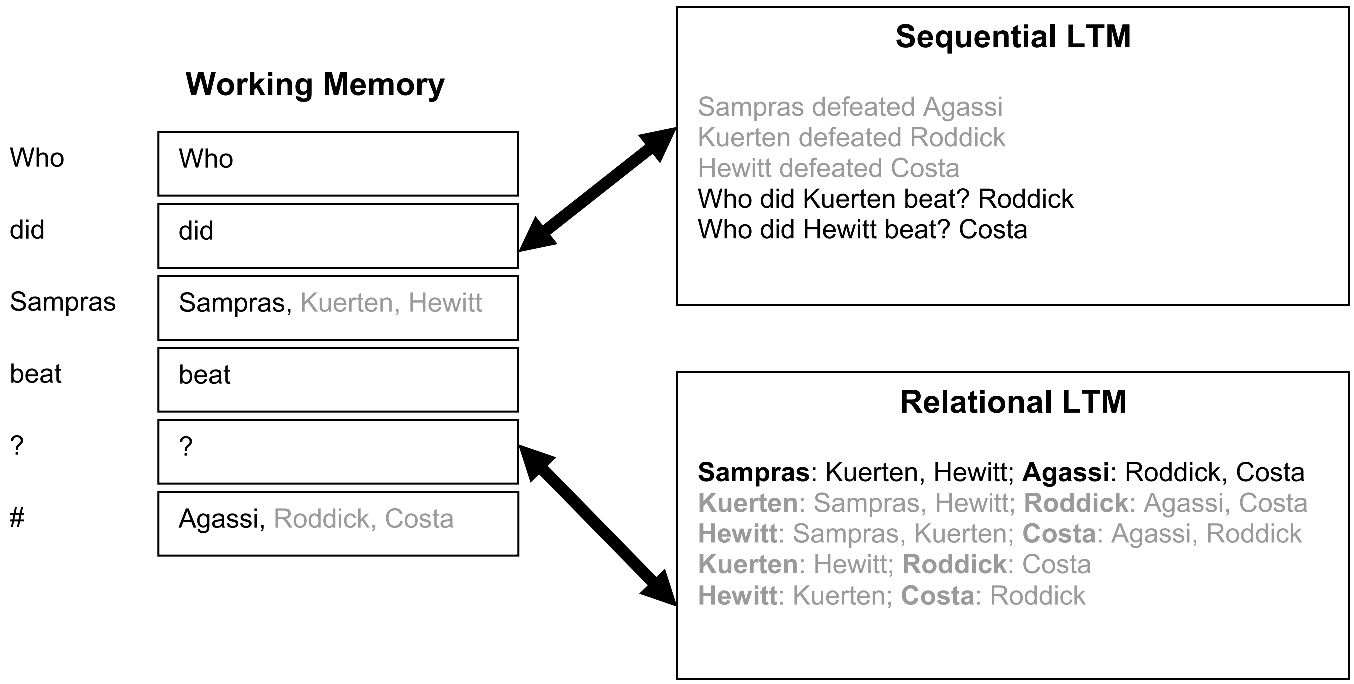

Figure 5: Relational Resolution: Agassi aligns with the answer slot as it is bound to the {Roddick, Costa} pattern in the retrieved relational trace.

That completes the description of the basic model. An outstanding question, however, is how one decides how similar two strings of words are and how they should align during sequential retrieval and resolution. Fortunately, there is a significant literature on this problem known as String Edit Theory (SET). In the next section, we give a brief outline of String Edit Theory.

Introduction to String Edit Theory

String Edit Theory (SET) was popularized in a book by (Sankoff & Kruskal, 1983) entitled, “Time warps, string edits and macromolecules” and has been developed in both the fields of computer science and molecular biology (Allison, Wallace, & Yee, 1992; Levenshtein, 1965; Needleman & Wunsch, 1970; Sellers, 1974). As the name suggests, the purpose of string edit theory is to describe how one string, which could be composed of words, letters, amino acids etc., can be edited to form a second string. That is, what components must be inserted, deleted or changed to turn one string into another. As indicated above, in the SP model, SET is used to decide which sentences from a corpus are most like the target sentence and which tokens within these sentences should align.

As an example, suppose we are trying to align the sentences “Sampras defeated Agassi” and “Kuerten defeated Roddick”. The most obvious alignment is that which maps the two sentences to each other in a one to one fashion:

```
Sampras    defeated   Agassi                                  A1
|          |          |
Kuerten    defeated   Roddick
```

In this alignment, we have three edit operations. There is a **change** of “Sampras” for “Kuerten”, a **match** of “defeated” and a **change** of “Agassi” for “Roddick”. In fact, this alignment can also be expressed as a sequence of edit operations:

<Sampras, Kuerten>
<defeated, defeated>
<Agassi, Roddick>

Using SET, sentences do not have to be of the same length in order to be aligned. If we add “Pete” to the first sentence, we can use a **delete** to describe one way in which the resulting sentences could be aligned:

```
Pete  Sampras    defeated   Agassi                            A2
|     |          |          |
-     Kuerten    defeated   Roddick
```

The “-“ symbol is used to fill the slot left by a deletion (or an insertion) and can be thought of as the empty word. The corresponding edit operation is denoted by <Sampras, ->. While these alignments may be the most obvious ones, there are many other options.

For instance, in aligning “Sampras defeated Agassi” and “Kuerten defeated Roddick”, we could start by deleting “Sampras”:

```
Sampras    defeated   Agassi     -                            A3
|          |          |          |
-          Kuerten    defeated   Roddick
```

Note that “Roddick” is now inserted at the end of the alignment (denoted <-, Roddick>).

Alternatively, we could have deleted “Sampras”, and then inserted “Kuerten” to give:

```
Sampras    -          defeated   Agassi                       A4
|          |          |          |
-          Kuerten    defeated   Roddick
```

There are a total of 63 ways in which “Sampras defeated Agassi” can be aligned with “Kuerten defeated Roddick”, but not all of these alignments are equally likely. Intuitively, alignment A4 seems better than A3 because the word “defeated” is matched. However, this alignment still seems worse than A1 because it requires "Sampras" to be deleted and "Kuerten" to be inserted. A mechanism that produces alignments of sentences should favor those that have many matches and should penalize those that require many insertions and deletions. To capture these intuitions, edit operations are assigned probabilities. Typically, match probabilities are higher than change probabilities which are higher than insertion or deletion probabilities. Assuming conditional independence of the edit operations, the probability of an alignment is the multiplication of the probabilities of the edit operations of which it is comprised. Each alignment is an exclusive hypothesis about how the two strings might be aligned and so the probability that the strings are aligned in one of these ways is the addition of the probabilities of the alignments. Given that there are an exponential number of alignments between strings one may be concerned that any algorithm based on SET would be infeasible. However, there exist efficient dynamic programming algorithms that have O(nm) time and space complexity where n and m are the lengths of the two strings (Needleman & Wunsch, 1970).

SET is a well developed and extensive literature and we have only been able to give the briefest overview. For more information readers are referred to Sankoff & Kruskal (1983) and Allison, Wallace and Yee (1992). In addition, Dennis (in press-a) provides a more complete mathematical treatment in the context of the SP model.

Text Mapping in the SP Model

In order for the relational representations generated by the SP model to play the same role as propositions in text comprehension theory, the basic requirements are that they be capable of encapsulating information at something like the idea unit level and that they be able to bind fillers to something like roles that participate systematically across different sentence instances. For instance, when presented with the sentence "Sampras outguns Agassi", the model must be capable of realizing that this is an idea unit, capturing the fact that Sampras and Agassi are not equivalent within this sentence and furthermore to generalize that realization across utterances with potentially very different structures, such as "Who won the match between Sampras and Agassi?". In this section, we describe a simulation experiment designed to investigate these issues. We start by outlining the domain that was used to test the model.

The Tennis News Domain

There were a number of criteria that were used to select the domain on which to test the model. Firstly, the domain was required to be one for which naturally occurring text was available as it is important that the model be capable of dealing robustly with the variety of sentences which are typically found in real text. Also, in real corpora there are many sentences that do not refer to the facts of interest at all, and the model should be capable of isolating the relevant ones.

Secondly, we wished to test the model's ability to extract relational information from sentences. Many question answering systems employ type heuristics rather than engaging in relational analysis. For instance, they might determine the date of the running of the Melbourne Cup by looking for sentences containing the term Melbourne Cup and returning any date within these sentences regardless of the role this date might fill. While such heuristics are often very successful in practice, there are some questions for which a relational analysis is necessary.

Finally, we were interested in testing the model's ability to take advantage of inference by coincidence, and so chose a domain in which the opportunities for such inferences are abundant.

Sixty nine articles were taken from the Association of Tennis Professionals (ATP) website at http://www.atptennis.com/. The articles were written between September 2002 and December 2002 and ranged in length from 134 to 701 words. In total there were 21212 words in the corpus. The documents were manually divided into sentences and the mean sentence length was 23.7. Table 1 provides an example article.

The tennis domain fulfills of the criteria outlined above. Naturally occurring text is available and there were many non target sentences that the model was required to reject in its search for relevant information. Choosing the winner of a tennis match cannot be solved by appealing to simple type heuristics as relevant source sentences often contain the names of both the winner and the loser so that the correct answer must be selected from items of the same type. Finally, in sports reporting of this kind there are often multiple cues many of which are indirect that allow the disambiguation of key facts – like who the winner of a match was.

Then 377 questions of the form "Who won the match between X and Y? X" were created. Any result that could be deduced from the article text was included. So, for instance, results that required the resolution of an anaphoric reference from other sentences in the same document were retained. Also, the winner was alternated between the first and second name positions so that the model could not simply repeat the name in the first slot in order to answer the question.

Results and Discussion

To test the model, each question was presented with the final answer slot vacant (e.g. "Who won the match between Sampras and Agassi? #"). The SP model was invoked to complete the pattern. During sequential retrieval the model was allowed to retrieve any sentence or question from the entire corpus. During relational retrieval, however, only facts derived from the sentences were allowed as retrieval candidates – that is the factual knowledge embodied by the questions was not permitted to influence the results.

The token with the highest probability in the # slot was assumed to be the answer returned by the model. Figure 6 shows a breakdown of the number of results in each category after sequential resolution and after relational resolution. Following relational processing, on about 67% of occasions the model correctly returned the winner of the match. 26% of the time it incorrectly produced the loser of the match. 5% of the time it responded with a player other than either the winner or loser of the match and on 3% of occasions it committed a type error, responding with a word or punctuation symbol that was not a player's name.

Table 1: Example of articles from the Association of Tennis Professionals (ATP) website.

US OPEN Sep. 08, 2002

Sampras Outguns Agassi in US Open Final

Sampras claims 14th Grand Slam title

Pete Sampras captured his 14th Grand Slam title - and fifth US Open - with a 6-3, 6-4, 5-7, 6-4 victory over Andre Agassi in the final at Flushing Meadows. Sampras, who had gone 33 tournaments without lifting a trophy since last winning Wimbledon in 2000, defeated his long-time rival Agassi for the 20th time in 34 meetings. With the win, Sampras joins Jimmy Connors as the only five-time US Open winners in the Open era, and at 31 years, 28 days, becomes the oldest since Ken Rosewall in 1970 (35 years, 10 months). The victory also marked Sampras' 71st match win at the US Open, which ties Bill Tilden in fourth place on the all-time singles list at the tournament, and also takes Sampras to 12th position in the ATP Champions Race 2002. Agassi moves to second behind Lleyton Hewitt.

En route to his third consecutive US Open final, his eighth in total, Sampras defeated Albert Portas, Kristian Pless, No. 33 seed Greg Rusedski, No. 3 Tommy Haas, No. 11 Andy Roddick and No. 24 Sjeng Schalken. Agassi, who was attempting to claim his third US Open championship (1994, 1999) and his eighth Grand Slam title, was looking for his first victory against Sampras in New York (0-3). Agassi reached his 13th Grand Slam final with victories over Robby Ginepri, Justin Gimelstob, Ramon Delgado, Jan-Michael Gambill, Max Mirnyi and No. 1 Hewitt.

In what was the 16th meeting in a tournament final between the two legends - their fifth Grand Slam championship match and third at Flushing Meadows (1990, 1995) - it was Sampras who began in fine form. Holding serve with ease, Sampras broke Agassi to lead 5-3 in the first set before clinching it with a trademark backhand volley. Sampras then used his momentum to break Agassi's serve in the first game of the second set and did so again to lead 5-2. But Agassi found a glimmer of hope as he found a window of opportunity to get one of the breaks back. It was the first time Agassi had broken Sampras in their last two matches having been unsuccessful at the US Open last year and in Houston earlier in 2002. However, Sampras made no mistake at the second attempt to close out the set, holding to love with his 16th ace of the match.

With Sampras beginning to show signs of fatigue, Agassi began to make an impression with his returns, and despite missing three break point opportunities in the fifth game, the 32-year-old from Las Vegas took the set after breaking for the second time of the match with Sampras serving at 4-5.

The fourth set seemed destined to go Agassi's way, as it was Sampras who had to survive a 20-point, 12-minute game at 1-2 and then saved another break point at 3-4. Having withstood the pressure, Sampras then turned the heat on Agassi to break at 4-4, and, after sending down his 33rd ace of the match gave himself three Championship points, of which he took the second with another backhand volley.

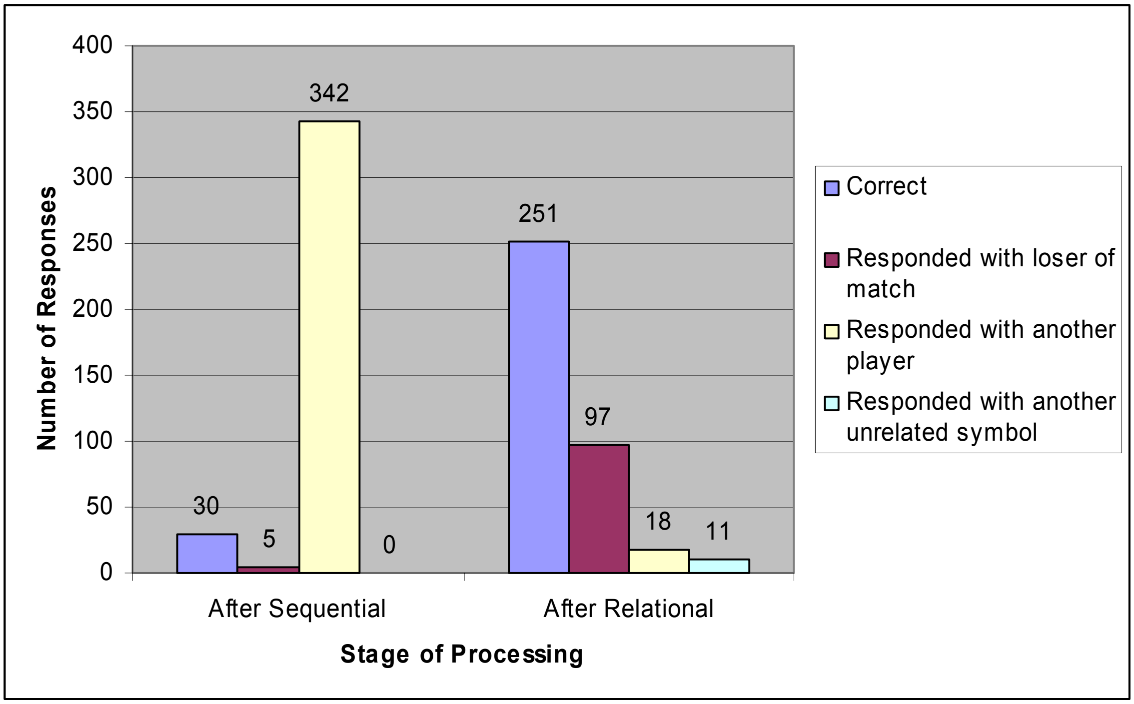

Figure 6: Breakdown of Result Types after Sequential Processing and after Relational Processing.

There are a number of ways in which one might seek to establish an appropriate baseline against which to compare these results. Because the model is developed in a pattern completion framework it is possible for any symbol in the vocabulary to be returned. There were 2522 distinct tokens in the corpus, so nominally the chance rate is less than one percent. However, one might also argue that the chance rate should be related to the number of elements of the appropriate type for a response – that is the number of names of players. There were 142 distinct players' names and so by this analysis baseline would also be below one percent. A further type distinction would be between winners and losers. There were 85 distinct winners which results in a baseline of just over one percent. Note that in any of these cases, the model is performing well above chance.

Note that the SP model is only given a pattern to complete, and so is not only answering the question, but is also extracting the relevant schema within which the question must be answered. In addition, when the SP model is run without relational retrieval or resolution performance drops from 67% to 8% correct (see Figure 6) – so it would seem that relational processing was critical. Given that the questions were not included in relational memory, performance must have been driven by the statistics of the articles rather than by the statistics of the questions.

Issues that compromised performance

In examining the types of errors committed by the model we found a number of recurring types. As mentioned earlier, the use of anaphora is quite common in this corpus. The current model has no mechanism for the resolution of anaphora which undermines both its ability to isolate the sentences containing the appropriate relational information and its ability to select the correct answer token. In addition, a mechanism for isolating appropriate context is necessary. On seven occasions in the current data set there are sets of players for whom the questions are ambiguous without the use of context to isolate the correct match. In addition, inference by coincidence can sometimes induce an incorrect response. For instance, the model induces that Schalken won the match against Pete Sampras in part on the basis of the sentence "Schalken, from the Netherlands, made his best-ever grand slam showing at the US open last month…" However, while having a best-ever showing is indicative of winning, in this case it is misleading as it was in fact Sampras who defeated him in the semifinals. Finally, the models lack of sensitivity to sublexical structure creates difficulties, particularly in deriving relational match when possessives are employed. There are then many avenues by which one could look to improve performance.

Inference in the SP Model

In the introduction, we listed three types of inference for which the SP model is capable of accounting – inference by coincidence, linguistic pattern completion and inference by relational retrieval. In this section, we first show how these mechanisms operate, before demonstrating inference by coincidence in the tennis news domain.

Inference by Coincidence

In production system models of inferencing, common sense knowledge is encoded in the form of if-then rules. For instance, most people know that "if X sold the Y to Z then it is also the case that Z bought the Y from X". Furthermore, it is the case that "Z now has the Y" and "X does not have the Y".

In production system parlance this might be expressed as:

```
if SOLD(X, Z, Y) then
        BOUGHT(Z, X, Y)
        POSSESS(Z, Y)
        NOT POSSESS(X, Y)
```

When attempting to establish whether "John owns a camera" the inferencing system might be asked to verify that POSSESS(John, camera), in which case it would activate the above rule and search memory to determine if John had ever been sold a camera (i.e. SOLD(X, John, camera)). This process, known as backward chaining, has two major disadvantages. Firstly, successful inference relies on all relevant common sense being appropriately coded in production system rules. Secondly, even inferences that people would find trivial can be computationally expensive because there may be many such rules that must be consulted in order to establish the truth value of the predicate. In this example, John may possess a camera because he was given it, or because he found it, or because he stole it etc. All of these possibilities must be considered as new predicates which in turn may rely upon additional rules and so forth.

The SP model provides a mechanism that exploits corpus statistics to avoid the necessity of explicit inferencing of this kind. To illustrate, suppose the model had been exposed to the following corpus:

1. IBM sold the widget to Microsoft .
2. Microsoft bought the widget from IBM .
3. SPSS sold the software to SAS .
4. SAS bought the software from SPSS .
5. Liverpool sold the player to Manchester .
6. Manchester bought the player from Liverpool .

Figure 7 shows the most probable alignments for the sentences “Charlie bought the lemonade from Lucy .” and “Lucy sold the lemonade to Charlie .” As the order of bindings in relational traces is irrelevant these sentences have very similar representations. In both cases, Charlie is bound to {Microsoft, SAS, Manchester}, lemonade is bound to {widget, software, player} and Lucy is bound to {IBM, SPSS, Liverpool}. That is, if we know that Charlie bought the lemonade from Lucy we automatically also know that Lucy sold the lemonade to Charlie without ever explicitly extracting or applying a rule to effect the transformation. Furthermore, if we consider the pattern {Microsoft, SAS, Manchester} as an owner role then the model also automatically “knows” that Charlie owns the lemonade.

Charlie	**bought**	**the**	**lemonade**	**from**	**Lucy**
Microsoft	bought	the	widget	from	IBM
SAS	bought	the	software	from	SPSS
Manchester	bought	the	player	from	Liverpool

Lucy	**sold**	**the**	**lemonade**	**to**	**Charlie**
IBM	sold	the	widget	to	Microsoft
SPSS	sold	the	software	to	SAS
Liverpool	sold	the	player	to	Manchester

Figure 7: Most probable alignments for “Charlie bought the lemonade from Lucy .” and “Lucy sold the lemonade to Charlie .”[1]

By this account, the process that allows us to implicitly form simple inferences and the process that allows us to generalize over different surface forms of a sentence are identical. It is the existence of a critical subset of sequential traces that allows the inference to occur. This type of inference provides a plausible account of how the human cognitive system is able to acquire the wealth of common sense knowledge that it appears to (c.f. Lenat, 1995) despite never being explicitly exposed to this information. In a way reminiscent of Latent Semantic Analysis (Landauer & Dumais, 1997), factual knowledge is accumulated implicitly via the statistics of the corpus. Note however, that unlike LSA the SP model assumes that this knowledge becomes

[1] Note here we just show the most probable alignments for ease of exposition. The model, however, always computes the sum of all possible alignments probability weighted.

available as a consequence of simple retrieval mechanisms operating over a large set of memory traces, rather than proposing comprehensive induction processes.

Linguistic Pattern Completion (Logical Inference)

While inference by coincidence may be responsible for the majority of automatic inference making that takes place during text comprehension, people are also able to engage in more controlled forms of inference. For instance, people are able to follow rules such as "if X sold the Y to Z then it is also the case that Z bought the Y from X" even in the absence of examples illustrating this principle. To show how this type of inference can be addressed suppose that the SP model had been exposed to the following corpus:

1. if X sold the Y to Z
2. then Z bought the Y from X
3. if Lucy sold the lemonade to Charlie

Note that trace one and two provide the linguistic fragments necessary to define the abstract rule, while trace three is a specific example, which the model will be required to use to infer that "Charlie bought the lemonade from Lucy". Processing now proceeds as in the previous example with Lucy being bound to {X}, lemonade to {Y} and Charlie to {Z}. Then in a second step, trace two is used to infer that Charlie bought the lemonade from Lucy.

Unlike in the previous example, no specific examples are available to make the connection between buying and selling in the way that they did in the previous section. Rather the abstract rule is used in a two stage process. Note, however, that the use of an abstract rule in this way does not lead to the rich set of bindings that were generated in the previous section. For instance, "Lucy" will be bound to {X} using the explicit rule whereas in the previous example "Lucy" was bound to {IBM, SPSS, Liverpool}, all symbols that have a grounded meaning and hence are liable to be used systematically within the systems experience. The symbol "X", on the other hand, may be used to mean many different things in different circumstances and consequently is not a useful symbol when making additional inferences such as inferring possession. So, while explicit reasoning is useful in domains where direct experience is limited, matching explicit rules may interfere with the process of acquiring richly interwoven knowledge structures when direct experience can be made available.

Also, note that implicit inference made the relevant information immediately available in relational memory, whereas explicit inference relies on the application of a rule which must be retrieved from sequential memory, resolved, retrieved from relational memory and resolved before the relational trace is entered into relational memory. This difference may explain why explicit inference takes more time, is more prone to dual task interference and why the process of explicit inference is more likely to be available for report.

In this example, the inference was made by the application of a logical rule. However, some controlled inference may be made not by the application of a rule, but rather by analogy to a separate domain. The next section addresses this kind of analogical inference.

Inference by Relational Retrieval (Analogical Inference)

There is now a well developed literature on analogical reasoning with a long history of well specified computational models (Kokinov & French, to appear). A prerequisite for all of these models is a propositional representation of the relevant facts in the base and target domains. In some models these propositional representations are symbolic (ANALOGY Evans, 1964; SMT Falkenhainer, Forbus, & Gentner, 1989; Gentner, 1983; ACME Holyoak & Thagard, 1989), while other models employ distributed representations (STAR Halford et al., 1994; LISA Hummel & Holyoak, 1997; Wilson, Halford, Gray, & Phillips, 2001). In either case, however, the practice has been to hand code the relevant facts in the appropriate representational format. While there has been considerable work on representation creation mechanisms (COPYCAT French, 1995; Hofstadter, 1995; Mitchell, 1993), the focus in this work has been on allowing different aspects of an analogical mapping problem to be highlighted by dynamically changing the representational scheme rather than on extracting the relevant information from naturalistic corpora (e.g. text corpora). As in the text comprehension domain, the inability to solve the text mapping problem has been an important limitation in assessing the viability of analogical reasoning models and has prevented them from scaling to realistic size knowledge bases. The SP model has the potential to solve this dilemma as it provides a mechanism by which a propositional knowledge base can be extracted automatically from a large corpus.

In addition, however, the basic mechanisms of the SP model implement a form of analogical inference. To illustrate we will use the model to solve a simple proportional analogy such as MAN:HOUSE::DOG:?.

Suppose the model has been exposed to the following corpus:

1. a mother is to a daughter
2. a father is to a son
3. CC a man is to a house

Figure 8 shows the most probable alignment for “a dog is to a #”. The input word “dog” has been aligned with {man} and the empty slot at the end of the sentence is aligned with {house}.

a	**dog**	**is**	**to**	**a**	**#**
a	man	is	to	a	house

a	**kennel**	**is**	**where**	**a**	**dog**	**lives**
a	house	is	where	a	man	lives

Figure 8: Most probably alignments of “a dog is to a #” and “a kennel is where a dog lives”.

Now, the relational trace for “a kennel is where a dog lives” also has a binding of “dog” onto {man} (see Figure 8) and hence is selected in relational retrieval. In the selected trace “kennel” is bound to {house} so during relational retrieval the empty slot is now filled with the correct answer.

While the simple mechanisms proposed in the SP model are clearly insufficient to model the broad spectrum of analogical reasoning results, they may prove useful in describing the sort of analogical inference that happens routinely during the normal course of text comprehension.

In this section, three types of inference that can be implemented by the SP model have been demonstrated. Inference by coincidence, linguistic pattern completion and inference by relational retrieval all occur as a consequence of the basic processing assumptions of the model with minimal requirements on the content of the input corpus. In particular, the distinction between automatic and controlled inference is explained by the fact that automatic inference is not a process at all, but rather a coincidence of relational representation that embodies an inference.

Of the three inference processes outlined here, it is inference by coincidence that one would expect to be ubiquitous. The question then arises to what extent the results of the tennis news simulation study were a consequence of inference by coincidence. In the next section, we address this question.

Demonstration of Inference by Coincidence

To assess the contribution that inference by coincidence made to the performance of the model in the tennis news task, the sentence with maximal retrieval probability for each query was classified into one of three categories.

The literal category contained those sentences where there was an explicit statement of the result – even if it required some interpretation. For example, when processing the question "Who won the match between Ulihrach and Vicente? Ulihrach" the highest probability relational trace was "Vicente bounced by Ulihrach" which literally states the result (even if it is necessary for one to interpret bounced in this context).

The inference category contained those sentences that did not contain a literal statement of the result, but which provided some evidence (not necessarily conclusive) for what the result may have been (see Table 2 for examples). For instance, when processing the question "Who won the match between Portas and Sampras? Sampras" the relational trace with the highest retrieval probability was "Sampras claims 14th Grand Slam title". While this sentence does not explicitly state the result of this match one can infer that if Sampras won the title then it is likely that he won this match. Note that this inference does not always follow as the writer may have made reference to a result from a different tournament, or the question may have come from a different article. However, the fact that Sampras won the title does provide evidence in favor of his having won this match. Unlike a traditional inference system, however, the SP model is making the inference by virtue of the fact that the names of people that appear in statements of the form "X claims --- title" also tend to appear in the winner slot at the end of the questions.

Table 2: Examples of inference by coincidence in the Tennis News domain. Each example shows the question and the sentence that generated the most probable relational trace.

Who won the match between Carlsen and Kiefer ? Carlsen

Kafelnikov now meets Kenneth Carlsen of Denmark in the second round.

Who won the match between Kiefer and Safin ? Safin

Safin , Kafelnikov surge toward hometown showdown

Who won the match between Ljubicic and Kutsenko ? Ljubicic

Sixth seed Davide Sanguinetti of Italy and eighth seed Ivan Ljubicic of Croatia took different paths to their opening-round wins at the president's cup in Tashkent.

Who won the match between Voltchkov and Haas ? Voltchkov

According to Haas, the injury first arose during Wednesday's match against Sargis Sargsian, and became progressively worse during practice and then the match against Voltchkov.

Who won the match between Srichaphan and Lapentti ? Srichaphan

Srichaphan has now won two titles in four finals this year.

Who won the match between Mamiit and Coria ? Coria

Kuerten, Coria withstand heat, set up fiery south American showdown

Finally, the other category included all remaining cases. These included traces in which both players were mentioned but the sentence could not have been used to conclude who the winner may have been. For example, when the question “Who won the match between Acasuso and Pavel? Acasuso” was presented the most probable relational trace was “Pavel and Acasuso to clash in Bucharest semis”. In addition, this category contains sentences which contradict the correct result. For example, the question “Who won the match between Pavel and Srichaphan? Pavel” produced the relational trace “Pavel, now 38-22 on the year, has reached two semifinals in 2002 Chennai l. to Srichaphan and Bucharest l. to Acasuso”. This situation occurs when a player revenges an earlier loss. In addition, the other category was assigned when the sentence was unrelated to the question. For instance, when the model was presented with the question “Who won the match between Meligeni and Etlis? Etlis” it returned “Kiefer quickly overcame Gaston Etlis of Argentina 6-2, 6-4 on Monday to qualify for the main draw of the Kremlin cup.”

Figure 9 shows the number of most probable relational traces in each category.

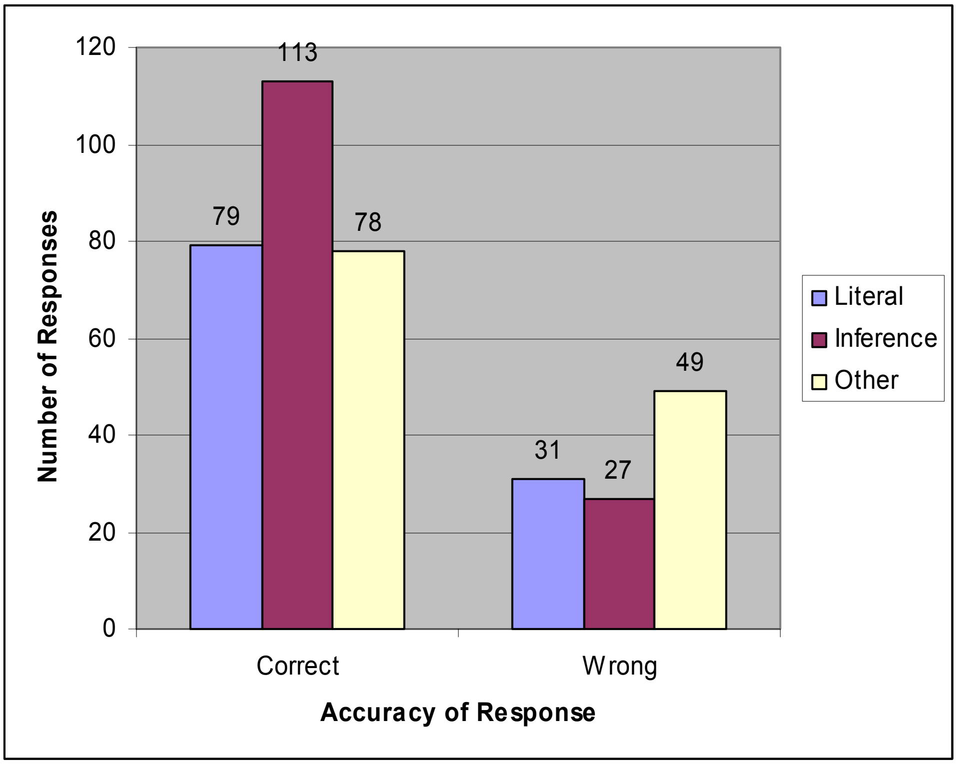

Figure 9: Breakdown of responses based on the accuracy of the response and the type of the most probable relational trace according to the model. Literal refers to traces in which the answer was stated explicitly. Inference refers to traces in which the answer was not stated, but from which it could be inferred. Other refers to traces from which the answer was not derivable. Note that these statistics are for the most probably trace only. The model, however, accumulates information from multiple traces, so that it is still possible for it to answer correctly even if the most probable trace does not contain the relevant information.

To get an indication of the contribution that inference by coincidence is making to correct responding, consider those correct responses which can be attributed to either literal or inference traces. On 59% of occasions the model was inferring the answer rather than relying on literal retrieval. Given that in each case a literal statement of the results existed in the corpus it is significant that inference by coincidence seems to be playing such a crucial role in the performance of the model.

Conclusions

The purpose of this paper was to assess the ability of the Syntagmatic Paradigmatic model to address the text mapping and inference generation problems in text comprehension. Critical was the extent to which the relational representations generated by the SP model are capable of playing the role that propositions have previously done.

On three important criteria they seem to be adequate. Firstly, they are capable of encapsulating something equivalent to an idea unit. Secondly, they are able to maintain filler – role bindings.

Finally, they are capable of doing this in a systematic way across sentence instances with different surface structures. Unlike propositions, however, relational representations can be generated directly from text and do not require a grammar or semantic roles to be specified in advance. They are deduced in an unsupervised fashion from the corpus.

In addition, the SP model offers a new inference mechanism. Inference by coincidence provides an explanation of how people are able to acquire a large database of inference rules without being exposed to them explicitly or even seeing antecedents and consequences in a contingent relationship. Furthermore, unlike previous notions of inference, inference by coincidence is not a process. Rather it is an emergent property that occurs as a direct consequence of retrieval. Inference by coincidence rules have no explicit instantiation and exist only as potentialities until they are used.

Despite the success of the model in the demonstrations outlined in this paper there remain a number of issues to be resolved. Firstly, the model has no real sense of a constituent. In the tennis news domain as outlined above, the answer was always a single word. In general, however, it will be necessary to accommodate multiword fillers and to do so in a way that respects linguistic structure.

Secondly, while inference by coincidence seems to be robust it has yet to be demonstrated that the types of sentence necessary to support inference by linguistic pattern completion and inference by relational retrieval occur in real corpora. Indeed it may well be the case that written text is unsuitable and that it will be necessary to turn to child directed speech corpora to find the relevant linguistic fragments. While the model does not currently address the sort of inference induced by contingent events, the fact that it is based on an episodic memory model suggests that this deficiency can be overcome. What is likely to be more difficult is addressing inference by perceptual simulation, as this would seem to require the inclusion of an adequate model of the perceptual processing apparatus.

References

Allison, L., Wallace, C. S., & Yee, C. N. (1992). Finite-State Models in the Alignment of Macromolecules. *Journal of Molecular Evolution, 35*(1), 77-89.

Bierwisch, M. (1969). On certain problems of semantic representation. *Foundations of language, 5*, 153-184.

Blaheta, D., & Charniak, E. (2000). *Assigning function tags to parsed text.* Paper presented at the Proceedings of the 1st Annual Meeting the North American Chapter of the ACL (NAACL), Seattle, Washington.

Collins, M. (1999). *Head-driven statistical models for natural language parsing.* University of Pennsylvania, Philadelphia.

Collins, M., & Quillian, M. R. (1969). Retrieval from semantic memory. *Journal of Verbal Learning and Verbal Behavior, 8*, 240-247.

Daelesmans, W. (1999). Introduction to the special issue on memory-based language processing. *Journal of experimental and theoretical artificial intelligence, 11*, 369-390.

Dennis, S. (2003a). *An alignment-based account of serial recall.* Paper presented at the Twenty Fifth Conference of the Cognitive Science Society.

Dennis, S. (2003b). *A comparison of statistical models for the extraction of lexical information from text corpora.* Paper presented at the Twenty Fifth Conference of the Cognitive Science Society.

Dennis, S. (in press-a). A memory-based theory of verbal cognition. *Cognitive Science*.

Dennis, S. (in press-b). An unsupervised method for the extraction of propositional information from text. *Proceedings of the National Academy of Sciences*.

Dougherty, M. R. P. (1999). MINERVA-DM: A memory processes model for judgements of likelihood. *Psychological Review, 106*(1), 180-209.

Durbin, M. A., Earwood, J., & Golden, R. M. (2000). *Hidden Markov Models for coding story recall data.* Paper presented at the Proceedings of the 22nd Annual Cognitive Science Society Conference.

Evans, T. (1964). A heuristic program to solve geometric analogy problems. In M. Fischler & O. Firschein (Eds.), *Readings in computer vision*: Morgan Kaufman.

Falkenhainer, B., Forbus, K. D., & Gentner, D. (1989). The structure-mapping engine: Algorithm and examples. *Artificial intelligence, 41*, 1-63.

Fillmore, C. J. (1968). The case for case. In E. Black & R. T. Harms (Eds.), *Universals of linguistic theory*. New York: Holt, Reinhardt and Winston.

Fillmore, C. J. (1971). Some problems for case grammar. In R. J. O'Brien (Ed.), *22nd round table. linguistics: deelopments of the sixties - viewpoints of the seventies* (Vol. 24, pp. 35-56). Washington D. C.: Georgetown University Press.

Fillmore, C. J., Wooters, C., & Baker, C. F. (2001). *Building a large lexical databank which provides deep semantics.* Paper presented at the Proceedings of the Pacific Asian Conference on Language, Information and Computation, Hong Kong.

Frank, S. L., Koppen, M., Noordman, L. G. M., & Vonk, W. (2003). Modeling knowledge-based inferences in story comprehension. *Cognitive Science, 27*(6), 807-950.

French, R. M. (1995). *The subtlety of sameness: A theory and computer model of analogy-making*. Cambridge, MA: MIT Press.

Gentner, D. (1983). Structure-Mapping - a Theoretical Framework for Analogy. *Cognitive Science, 7*(2), 155-170.

Gildea, D., & Jurafsky, D. (2002). Automatic Labeling of semantic roles. *Computational Linguistics, 28*(3), 245-288.
Golden, R. M., & Rumelhart, D. E. (1993). A parallel distributed processing model of story comprehension and recall. *Discourse processes, 16*, 203-207.
Goldinger, S. D. (1998). Echoes of echoes? An episodic theory of lexical access. *Psychological Review, 105*(2), 251-279.
Graesser, A. C., Singer, M., & Trabasso, T. (1994). Constructing inferences during narrative text comprehension. *Psychological Review, 101*, 375-395.
Halford, G., Wilson, W., Guo, K., Gayler, R., Wiles, J., & Stewart, J. (1994). Connectionist implications for processing capacity limitations in analogies. In K. J. Holyoak & J. Barnden (Eds.), *Analogical connections* (Vol. 2, pp. 363-415). Norwood: Ablex.
Harrington, M., & Dennis, S. (2003). *Structural priming in sentence comprehension.* Paper presented at the Twenty Fifth Conference of the Cognitive Science Society.
Hintzman, D. L. (1984). Minerva-2 - a Simulation-Model of Human-Memory. *Behavior Research Methods Instruments & Computers, 16*(2), 96-101.
Hintzman, D. L. (1988). Judgments of Frequency and Recognition Memory in a Multiple-Trace Memory Model. *Psychological Review, 95*(4), 528-551.
Hofstadter, D. (1995). *Fluid concepts and creative analogies: Computer models of fundamental mechanisms of thought*. New York: Basic Books.
Holyoak, K. J., & Thagard, P. (1989). Analogical Mapping by Constraint Satisfaction. *Cognitive Science, 13*(3), 295-355.
Hummel, J., & Holyoak, K. J. (1997). Distributed representations of structure: A Theory of Analogical Access and Mapping. *Psychological Review, 104*, 427-466.
Katz, J. J., & Fodor, J. A. (1963). The structure of semantic theory. *Language, 39*, 170-210.
Kingsbury, P., Palmer, M., & Marcus, M. (2002). *Adding semantic annotation to the Penn TreeBank.* Paper presented at the Proceedings of the Human Language Technology Conference, San Diego, CA.
Kintsch, W. (1974). *The representation of meaning in memory*. New York: Wiley.
Kintsch, W. (1998). *Comprehension: a paradigm for cognition*: Cambridge University Press.
Kokinov, B., & French, R. M. (to appear). Computational models of analogy-making. In *Macmillan Encyclopedia of the Cognitive Sceinces*.
Landauer, T. K., & Dumais, S. T. (1997). A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review, 104*, 211-240.
Lenat, D. B. (1995). CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM, 38*(1).
Levenshtein, V. I. (1965). Binary codes capable of correcting deletions, insertions and reversals. *Dokl. Akad. Nauk. SSSR, 163*, 845-848.
Lin, D. (1993). *Princinciple-based parsing without overgeneralization.* Paper presented at the Proceedings of ACL-93, Columbus, OH.
Lin, D. (1994). *Principar - an efficient, broad-coverage, principle-based parser.* Paper presented at the Proceedings of COLING-94, Kyoto, Japan.
Lin, D. (1998). *Dependency-based evaluation of MINIPAR.* Paper presented at the Workshop on the Evaluation of Parsing Systems, Granada, Spain.
Lin, D., & Pantel, P. (2001). Discovery of inference rules for question answering. *Natural Language Engineering, 7*(4), 343-360.

Logan, G. D. (1988). Towards an instance theory of automatization. *Psychological Review, 95*, 492-527.
McKoon, G., & Ratcliff, R. (1992). Inference during reading. *Psychological Review, 99*, 440-466.
Mitchell, M. (1993). *Analogy-making as perception: A computer model*. Cambridge, MA: MIT Press.
Moldovan, D., Harabagiu, S., Girju, R., Morarescu, P., Lacatusu, F., Novischi, A., et al. (2002). *LCC tools for question answering.* Paper presented at the Eleventh Text Retrieval Conference (TREC 2002).
Nakisa, R. C., & Plunkett, K. (1998). Evolution for rapidly learned representations for speech. *Language and cognitive processes, 13*(2/3), 105-127.
Needleman, S. B., & Wunsch, C. D. (1970). A general method applicable to the search for similarities in teh amino acid sequence of two proteins. *Journal of Molecular Biology, 48*, 443-453.
Nosofsky, R. (1986). Attention, similarity and the idenitification-categorization relationship. *Journal of Experimental psychology: General, 115*, 39-57.
O'Hara, T., & Wiebe, J. (2002). *Classifying preposition semantic roles using class-based lexical associations* (No. NMSU-CS-2002-013): Computer Science Department, New Mexico State University.
Palmer, M., Rosenzweig, J., & Cotton, S. (2001). *Automatic predicate arguemnt analysis of the Penn TreeBank.* Paper presented at the Proceedings of HLT 2001, First International Conference on Human Language Technology Research, San Francisco.
Sankoff, D., & Kruskal, J. B. (1983). *Time warps, string edits and macromolecules: the theory and practise of sequence comparison*: Addison Wesley.
Sellers, P. H. (1974). An algorithm for the distance between two finite sequences. *Journal of Combinatorial Theory, 16*, 253-258.
Shiffrin, R. M., & Steyvers, M. (1997). Model for recognition memory: REM - Retrieving effectively from memory. *Psychonomic Bulletin & Review, 4*(2), 145-166.
Smith, E. E., Shoben, E. J., & Rips, L. J. (1974). Structure and process in semantic memmory: A feature model for semantic decision. *Psychological Review, 81*, 214-241.
Smith, J. D., & Minda, J. P. (1998). Prototypes in the mist: The early epochs of category learning. *Journal of Experimental Psychology: Learning, Memory and Cognition, 24*(6), 1411-1436.
Soubbotin, M. M., & Soubbotin, S. M. (2002). *Use of patterns for detection of answer strings: A systematic approach.* Paper presented at the Eleventh Text Retrieval Conference (TREC 2002).
Stallard, D. (2000). *Talk'n'travel: A conversational system for air travel planning.* Paper presented at the Proceedings of the 6th Applied Natural Language Processing Conference (ANLP'00).
Tulving, E., & Thompson, D. M. (1973). Encoding spcificity and retireval processes in episodic memory. *Psychological Review, 80*, 352-373.
van Dijk, T. A. (1972). *Some aspects of text grammars*. The Hague: Mouton.
Van Valin, R. D. (1993). A synopsis of role and reference grammar. In R. D. Van Valin (Ed.), *Advances in Role and Reference Grammar*. Amsterdam: John Benjamins Publishing Company.

Wilson, W., Halford, G., Gray, B., & Phillips, S. (2001). The STAR-2 model for mapping hierarchially structured analogs. In D. Gentner, K. J. Holyoak & B. Kokinov (Eds.), *The analogical mind*. Cambridge, MA: MIT Press.

Appendix A: Computational Linguistics Methods for Text Mapping and Inference Generation

While computational linguistics has yet to provide a complete solution to either the text mapping or inference rule generation problems a number of techniques have been reported that offer partial solutions. In this literature, being able to identify propositional information is an enabling technology in creating more intelligent question answering and information extraction systems. For instance, in the 2002 Text REtrieval Conference (TREC) competition, the PowerAnswer system (Moldovan et al., 2002), which converts both questions and answers into propositional form and employs an inference engine, achieved a confidence weighted score of 0.856, a substantive improvement over the second placed exactanswer (Soubbotin & Soubbotin, 2002), which relies solely on a surface analysis of the text and received a score of 0.691.

In this section, we briefly discuss two classes of extraction methods that have been investigated in the computational linguistics literature – supervised semantic parsers and dependency parsers.

Supervised Semantic Parsing

The task addressed by supervised semantic parsers is to take a sentence and assign role labels to the relevant constituents for each of the predicates in the sentence (Blaheta & Charniak, 2000; Gildea & Jurafsky, 2002; O'Hara & Wiebe, 2002; Palmer, Rosenzweig, & Cotton, 2001).

For instance, given the sentence:

Sampras outguns Agassi in US Open Final

these systems might produce an intentional annotation such as:

[$_{\text{Winner}}$ Sampras] outguns [$_{\text{Loser}}$ Agassi] [$_{\text{Location}}$ in US Open Final]

This work has been driven, at least in part, by the availability of semantically labeled corpora such as Propbank (Kingsbury, Palmer, & Marcus, 2002) and FrameNet (Fillmore, Wooters, & Baker, 2001) which provide the relevant training data.

Perhaps the most appropriate of these systems for use in text comprehension is that of Gildea and Jurafsky (2002) as they attempt to label all roles available in a sentence (c.f. Blaheta & Charniak, 2000). In their system, a classifier is trained using features such as voice, constituent grammatical type, the path between predicate and the constituent, the lexical head of the constituent and the predicate. To label a new sentence, the system first uses a statistical parser (such as Collins, 1999) to extract the features and then applies the classifier. As a consequence errors can arise both due to parse errors resulting in improperly identified constituents and due to role mislabeling. While the system achieves 82% accuracy in labeling, performance drops to 65% precision and 61% recall when it must also find the constituents.

Whether the system would be sufficient for text comprehension research is an open question. While the precision and recall figures may seem low, some of the difficulty arises because of differences in the parsing formalism between the Collins parser and the identification of constituents in the FrameNet corpus on which it is trained. For instance, in FrameNet relative pronouns are included with their antecedents in constituents. For instance consider the following sentence,

“In its rough state he showed it to [$_{Agt}$ the Professor, who] bent [$_{BPrt}$ his grey beard] [$_{Path}$ over the neat script] and read for some time in silence.”

Notice that “who” has been included in the agent constituent, a consistent property of the FrameNet corpus which the Collins parser does not replicate (Gildea & Jurafsky, 2002).

In addition, the Gildea and Jurafsky system was tested on unrestricted text. Often the texts used in text comprehension research and those produced by subjects in experimental tasks show less grammatical variation and may be easier for the system to label.

One important limitation of supervised semantic parsers is that they rely on the accuracy, coverage and labeling scheme of their training set. In particular, the semantic roles employed by the systems are those defined by the corpus annotators. However, deciding upon a best set of semantic roles has proven extremely difficult. There are a great many schemes that have been proposed ranging in granularity from very broad, such as the two macro-role proposal of Van Valin (1993), through theories that propose nine or ten roles, such as Fillmore (1971), to much more specific schemes that contain domain specific slots such as ORIG_CITY, DEST_CITY or DEPART_TIME that are used in practical dialogue understanding systems (Stallard, 2000). In any given experimental context then, it is necessary to consider how specific the role labels must be in order to demonstrate the phenomena of interest.

Dependency Parsers

Dependency parsers are another mechanism that provides proposition-like information from open sentences. The purpose of a dependency parser is to annotate a sentence with links that specify the dependencies in a sentence. For instance, in the sentence “the large cat sat on the mat” the Minipar dependency parser (Lin, 1993, 1994, 1998) generates the following labeled arcs:

sit:s:cat
cat:det:the
cat:mod:large
sit:subj:cat
sit:mod:on
on:pcomp-n:mat
mat:det:the

which can be visualized as in Figure 14.

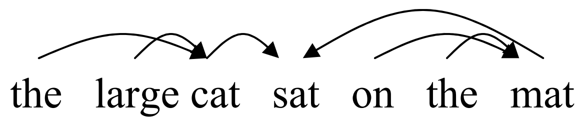

Figure 14: Example of Minipar dependency parse for the sentence “The large cat sat on the mat.”

While dependency parsers do not give role-filler bindings per se they do enumerate relationships and they have the advantage of much better performance than existing semantic parsers. For instance, Lin (1998) reported 79% recall and 89% precision on the SUZANNE corpus.

Furthermore, dependency parsers have been shown to be useful for extracting inference rules from text. Lin and Patel (2001) used Minipar to extract simple inference rules such as if “X solves Y” then “X deals with Y”. While the rules extracted in this case are mainly surface form transformations, given the inference by coincidence results reported in this paper it seems likely that Minipar could be used to extract deeper relationships as well (a research direction that we are pursuing currently).