A BASIS FOR PRONOMINAL ANAPHORA RESOLUTION USING A MODEL OF WORKING MEMORY AND LONG-TERM MEMORY

Clifford James Thompson

B.Sc., University Of Northern British Columbia, 2001

Thesis Submitted In Partial Fulfillment Of
The Requirements For The Degree Of
Master Of Science
In
Mathematical, Computer, And Physical Sciences
(Computer Science)

The University Of Northern British Columbia

May 2006

Abstract

This thesis presents a new theory of information modelling in natural language processing that attempts to resolve anaphoric references, while also addressing the problem of knowledge complexity. A modular model of semantic representation is introduced that addresses the deficiencies of existing representations, as well as the drawbacks associated with expanding these semantic representations. Rather than using a single semantic representation to model human knowledge and the knowledge within a sentence, the theory proposes a modular, multi-level model which is based around a semantic network. The behaviour of the model uses theories on the nature of working and long-term memory from cognitive psychology. Two methods of artificial neuron activation and decay were implemented – the ACT-R model and the Thompson model. Maximum success rates of 54.10% and 83.61% were achieved for *The Three Brothers* corpus, and maximum success rates of 56.00% and 86.67% were achieved for the *Rumpelstiltskin* corpus.

This thesis is approved
for recommendation to the
Graduate Council.

Thesis Director:

____
Dr. Charles Brown

Thesis Committee:

Dr Charles Brown

Dr. William Owen

Dr. Liang Chen

Table of Contents

List of Figures

List of Tables

Acknowledgements

This thesis would not have been a success without the patience and hard work of many individuals to whom I am greatly indebted. First, and foremost, I must thank my supervising professor, Dr. Charles Brown, for introducing me to the field of Artificial Intelligence and encouraging me to enter graduate studies. Your selfless dedication of time and effort will always be remembered. I must also thank Dr. William Owen for his advice in a field that was completely foreign to me. Your help and direction allowed me to achieve the multi-disciplinary approach that I persevered for. I also would like to thank Dr. Liang Chen for his advice and comments, Dr. Paul Siakaluk for taking the time to be the external examiner for my thesis, Erin Beveridge for the original LaTeX style files, and UNBC for awarding me the Graduate Entrance Scholarship (which made the decision of graduate studies much easier). And finally, I must thank my wife Coral. Without your love, advice, encouragement, and general *kick in the butt*, this thesis would have not have been possible.

CHAPTER ONE

Overview

She had a pretty gift for quotation, which is a serviceable substitute for wit.

- W. Somerset Maugham

1.1 Introduction

It seems ironic that although natural languages are very difficult to model, the languages themselves are quite effective and efficient for communication. If humans use their native language with ease, then why is it so hard for computers to understand natural languages? One of the most obvious answers is that the human brain is so complex. The complexity of human knowledge, and the medium on which it is stored and processed, cannot be understated.

This thesis presents a new theory of memory modelling in natural language processing that attempts to resolve anaphoric references, while also addressing the problem of complexity. Rather than using a single semantic model to represent human knowledge and the knowledge within a sentence, the theory proposes a more general model where multiple semantic representations can be used in a system that models the observed behaviour of working and long-term memory.

1.2 Pronominal Anaphora Resolution

The goal of this thesis was to develop a multi-level model of human memory that is modular and flexible, processes multiple well-known semantic representations such as semantic networks, conceptual graphs and quasi-logical form, and uses these models to resolve anaphoric references. In-particular, this thesis focused on the resolution of simple pronouns such as *he*, *she*, *they*, etc. In the most general of cases, pronoun resolution is quite simply a matter of searching backwards through a corpus of text until the first noun phrase that matches such attributes as *number* and *gender* is found:

Bright and early the next morning, $\boxed{\textit{the shoemaker}}_1$ *rose and went to* $\boxed{\textit{his}}_1$ *work bench. To his amazement, there on the table were* $\boxed{\textit{two shoes}}_2$*, already finished.* $\boxed{\textit{They}}_2$ *were beautifully made, neat and true, and with not a single false stitch.*

The situation can be made slightly more complex by making antecedents separate entities in the context:

For some time that same thing happened, until $\boxed{\boxed{\textit{the good man}}_1 \textit{ and } \boxed{\textit{his wife}}_2}_3$ *were thriving and prosperous. But* $\boxed{\textit{they}}_3$ *were not satisfied to have so much done for* $\boxed{\textit{them}}_3$ *and not know to whom* $\boxed{\textit{they}}_3$ *should be grateful.*

But of course, this is not always the case. The next example, adapted from [Sidner 1983], demonstrates where this method of resolution can break down:

$\boxed{\textit{My neighbours}}_1$ *have* $\boxed{\textit{a monster Harley 1200}}_2$. $\boxed{\textit{They}}_3$ *are really huge but gas efficient bikes.*

In the second sentence, if an individual was to read just the pronoun *they*, their initial preference for the reference may not be *a monster Harley 1200* based on number alone. In this context, a common preference for the pronoun *they* would be *my neighbours.* After reading the remainder of the second sentence, it is apparent that this conclusion was incorrect. Given the additional context, common knowledge concludes that the neighbours are not motorcycles[1].

1.3 Outline of Thesis

Chapter 2 discusses numerous semantic representations that have been introduced over the past few decades. The general domain of use is covered for each semantic representation

[1]That is unless your neighbours actually are motorcycles.

as well as some of the drawbacks for each structure. Chapter 3 will discuss the evolution of theories of short-term and long-term memory as well as current theories in human memory. Several modern psychological models of short-term and long-term memory will be elaborated on, as well as some computational memory models. In Chapter 4, the discussion on semantic representations will move towards a theory on combining semantic representations to overcome their individual deficiencies with the intention of creating a system that is easier to understand and easier to expand. The theory will also model human behaviour more closely.

Chapter 5 will discuss the current state of understanding in pronominal anaphoric reference and anaphora resolution. Several theoretical problems will be introduced and discussed. A number of modern anaphora resolution algorithms will be presented that attempt to solve anaphoric reference issues. The chapter will conclude with an examination of what occurs when humans fail to resolve anaphoric references. In Chapter 6, the combined semantic representation model, memory models, and anaphora resolution algorithms will be integrated into a system that will attempt to solve anaphoric reference problems introduced in Chapter 5. Chapters 7 will cover the methodology for testing the implemented model, the results of testing, and a discussion of those results. Chapter 8 will conclude the thesis by comparing the testing results with the results of other models, and a discussion on how the model presented in this thesis could be improved.

CHAPTER TWO

Declarative Semantic Structures

Oh, and sir, you're wrong. We won't be apart - we just won't be together.

- ARNOLD J. RIMMER *(Holoship)*

Although many types of semantic representations have emerged during the history of natural language processing research, understanding in the domain of semantics is still limited. Some models fall short and are intended for a limited knowledge domain. Others can be expanded but the resulting expansions are often unclear or more difficult to computationally manage.

In this chapter, a number of semantic representation models are examined. As each model is investigated, the shortcomings of each model will be shown. The examination of these shortcomings will lay the initial groundwork for a hypothesis on improving these models. By integrating each semantic model separately into a larger, multi-level system, it is hypothesized that the resulting system would be easier to expand than a system with a single complex semantic model, and would provide a diverse knowledge base from which an anaphora resolution algorithm, or group of algorithms, could draw from. James Allen makes a statement in [Allen 1995] to this vain:

> ...a vigorous debate about knowledge representation is actually the result of each of the debaters focusing on one of the aspects of representation without considering the concerns of the other.

Humans apply much implicit knowledge when understanding an utterance. Information in long-term memory is not considered in many structures, and even if it is, the information is stored only at the discourse level. Ignoring the complexity of a human knowledge base only trivializes the vast learning power of the human mind.

2.1 Semantic Networks

Some of the earliest research with respect to semantic networks can be found in [Quillian 1968] and [Collins and Quillian 1969]. Semantic networks were first introduced as a model of human memory. How semantic networks are realized as models in computation is quite a broad topic. Interpretation varies from graphs with concepts as nodes and the associations between the nodes as links, to more complex graphs such as Sowa's conceptual graphs [Sowa 1984] or conceptual hierarchies [Ma and Isahara 2000] [Chung and Moldovan 1993] . For the purposes of this thesis, semantic networks will be restricted to the first definition, graphs with concepts as nodes and associations as links.

Figure 2.1 represents a semantic network that has a strength associated with each link. The network roughly represents an artificial neural network, which will be discussed further in Chapter 3. Each node[1]represents a single topic or concept. Thus, the relationship between two semantic concepts is based on the strength of the association between the two semantic concepts. The drawback of this model is that it only models a loose relationship between topics. It does not identify what the relationship is. The next example illustrates how drawing the appropriate knowledge from a semantic network would be difficult:

John had a son named Bob. His son is an excellent skier.

In this example, an anaphora resolution algorithm would have a difficult time resolving the possessive pronoun *His* without the father-son relationship being modelled more explicitly.

[Kazuhiro et al 1992] and [Berger et al 2004, Belew 1987] demonstrate examples of semantic networks with weighted links being used in kana-kanji conversion and information retrieval, respectively. In Chapter 4 we will see that semantic networks will not be used to model knowledge directly. Rather, they will be used to connect semantic concepts and their associated semantic representations.

[1]A node does not necessarily represent a single neuron within the human brain. A node could represent a group of neurons.

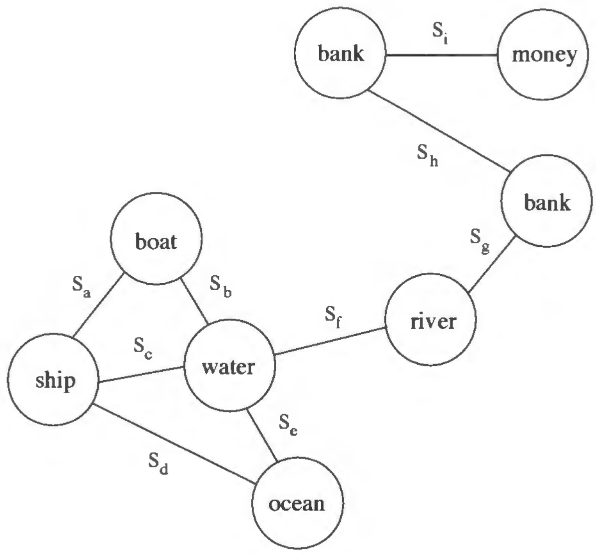


Figure 2.1: Probabilistic Semantic Network

2.2 Conceptual Graphs

Some of the earliest work with respect to conceptual graphs can be attributed to Sowa in [Sowa 1976, Sowa 1979, Sowa 1984]. Conceptual graphs are defined as a directed bipartite graph with two types of nodes. Each node in the graph can be either a *concept* or a *conceptual relation.* Concepts can be concrete (such as cat), or they can be abstract (such as sadness). Conceptual relations can have an arity of $n \geq 1$. Figure 2.2 illustrates conceptual relations with various arities.

Conceptual graphs are not limited to the simple relations shown in Figure 2.2. They can also model simple sentences, as seen in Figure 2.3. Since conceptual graphs are used extensively in database systems, relational database theory allows us to perform certain operations to obtain new conceptual graphs, such as *copy*, *restrict*, *join*, and *simplify.* In their basic form, conceptual graphs do not model the strength of relationships. Common knowledge dictates that information stored in long-term memory is not as concrete as the conceptual graph model it to be. Fuzziness with respect to relations is not accounted for. Conceptual graphs, as defined by Sowa, do not model temporal information implied by verb tense and verb aspect. This deficiency is apparent in Figure 2.4 adapted from

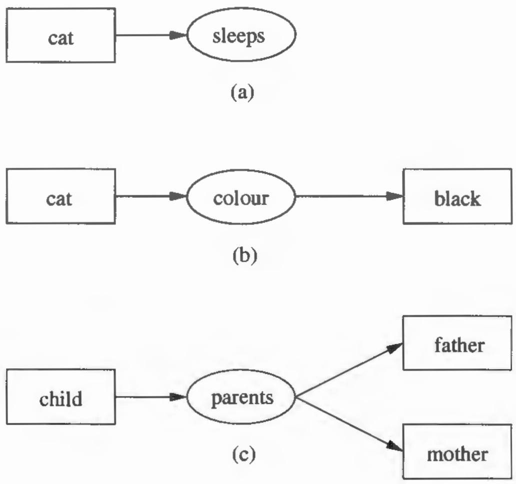


Figure 2.2: Conceptual Relations of Arity (a) 1 (b) 2 and (c) 3

[Sowa 2000]

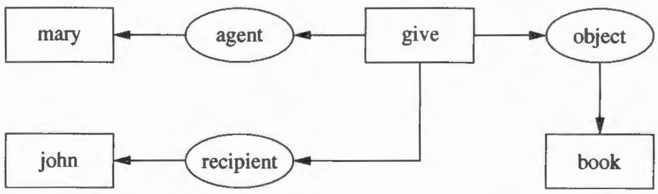


Figure 2.3: Conceptual Graph for *Mary gave John the book.*

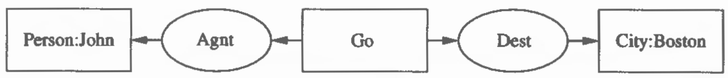


Figure 2.4: Conceptual Graph for *John is going to Boston*

It is apparent that the tense and aspect have been lost for the verb phrase *is going.* Tense and aspect could be reflected by adding another relation, as shown in Figure 2.5.

How would we model one event occurring before another event? Another relation would seem to be a possible answer. This process of adding relations to reflect previous missed information could lead to quite a rat's nest. It becomes difficult to separate verbs and nouns from other semantic information without adding more relations, and it potentially becomes a model that is more difficult to expand and maintain.

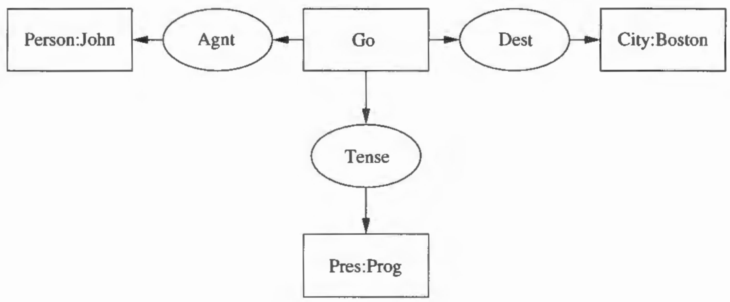


Figure 2.5: Another Conceptual Graph for *John is going to Boston*

2.3 Conceptual Hierarchies

Conceptual hierarchies are a very common structure in object oriented programming. They allow programmers to show how certain objects inherit the properties of another object. Figure 2.6 illustrates an example of how classification and sub-classification have been observed by biology throughout the world. Subclasses inherit attributes from their superclass as well as adding their own attributes. Research in cognitive categorization, such as [Kay 1971, Rosch et al 1976], suggests that the human mind stores and groups information based on taxonomy in long-term memory. One of the main advantages of conceptual hierarchies is that they are very efficient at storing information [Ma and Isahara 2000]. Information common to many concepts is only stored once.

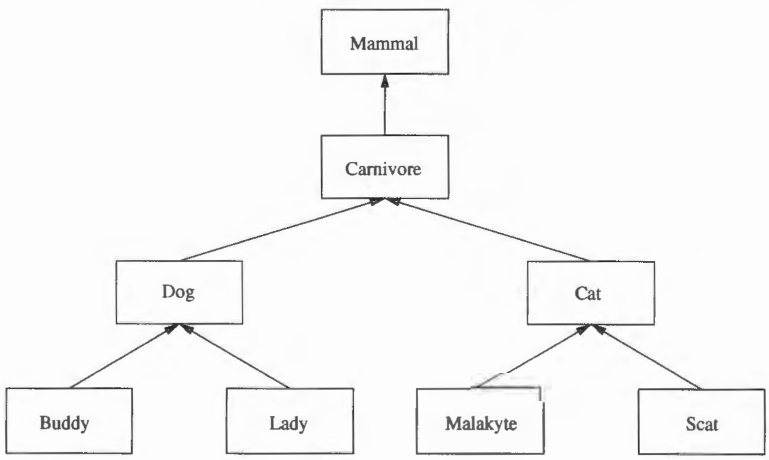


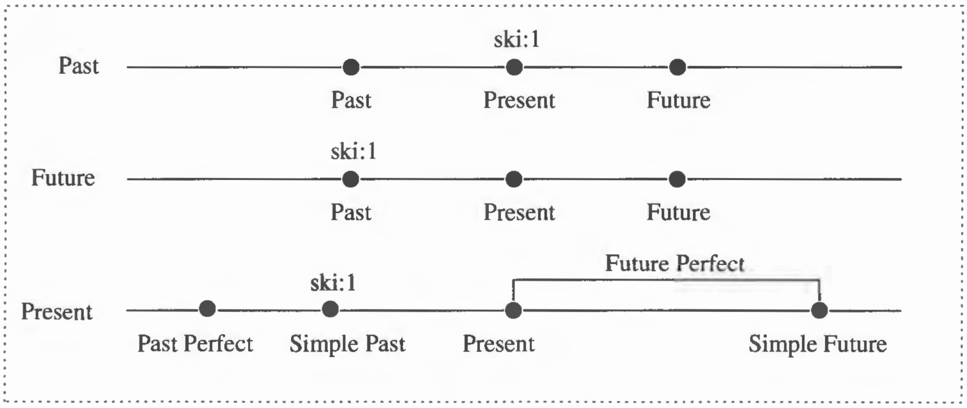
Figure 2.6: Conceptual Hierarchy

One the main disadvantages of conceptual hierarchies is their intended use: modelling concepts and the inheritance of characteristics between concepts. They are not intended to model entire sentences or any kind of temporal information.

2.4 Event Timeline Models

The Bull Framework

In [Culce-Murcia and Larsen-Freeman 1999], Celce-Murcia and Larsen-Freeman adapted the The Bull Framework [Bull 1960] (originally created for Spanish) for teaching ESL students English. The Bull Framework proposes four axes of time: past, present future, and hypothetical. The first three axes contain a point of reference in the centre and the times occurring before and after the time of reference to the left and right, respectively. Figure 2.7 demonstrates an example using the verb *ski.* The fourth axis, hypothetical, is used to model hypothetical events, for example, events created using constructions like *if..then.*



"I skied in the Whistler–Blackcomb backcountry."

Figure 2.7: Timeline Model

Since the Bull Framework was only intended for explaining verb tenses, it does not suitably model many aspects of natural languages such as nouns, adjectives, and adverbs.

The Reichenbach Theory

In [Allen 1995], Allen presents Reichenbach's theory on timeline representation from [Reichenbach 1947]. Reichenbach theorized that each verb embeds information about three points in time: *time of speech* (S), *time of the event/state*(E), and *time of reference*(R). In the simple aspect[2], the *time of the event/state* and the *time of reference* are always equivalent. This equivalency does not exist for the perfect and posterior aspects[3]. Figure 2.8 gives a table outlining the various timelines for different tense and aspect combinations. Table 2.1 shows some example sentences and their tense/aspect. Reichenbach methodology is very similar to that of Bull's in that they both attempt to model a type of *temporal ordering*, which is implied by varying tense and aspect combinations in sentences. Naturally, Reichenbach's theory does not attempt to account for the nature of nouns, adjectives, and adverbs.

Tense	Example Sentence
Simple Present	Jack sings
Simple Past	Jack sang
Simple Future	Jack will sing
Perfect Present	Jack has sung
Perfect Past	Jack had sung
Perfect Future	Jack will have sung
Posterior Present	Jack is going to sing
Posterior Past	Jack was going to sing
Posterior Future	Jack will be going sing

Table 2.1: Example Sentences Taken from [Allen 1995]

2.5 Thematic Roles

Thematic roles are linguistic entities (embodied in the form of noun phrases) that satisfy certain semantic constraints implied by the main verb phrase of a sentence. The idea of

[2]It must be noted that the *progressive aspect* and *perfect progressive aspect* are missing from Allen's listings.

[3]Allen refers to *simple* as being a tense, not an aspect.

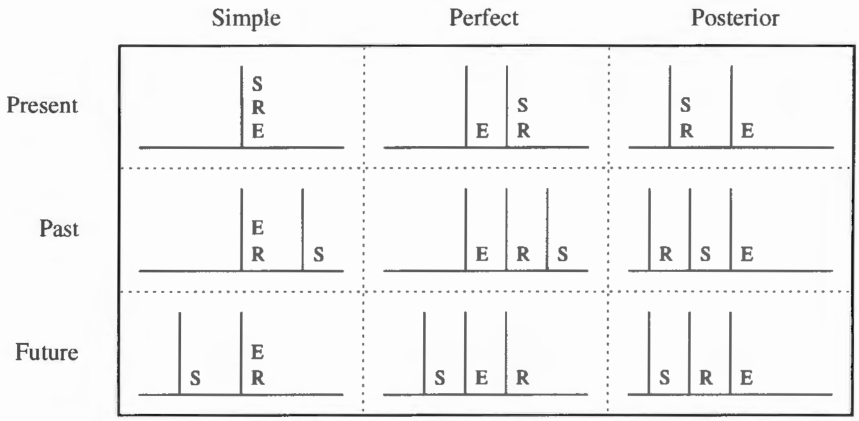


Figure 2.8: Reichenbach Timeline Model

thematic roles draws a close parallel to the morphological *case* systems found in languages such as German and Latin, but expands on the case system by adding a much larger number of *cases*.

Verb phrases require that these thematic roles are present before a sentence can make sense semantically. Altmann demonstrated in [Altmann 1999] that even if all thematic roles are met for a verb phrase, if the antecedent of a thematic role is not plausible, the sentence will not make sense. Figure 2.9 gives an example of (a) an implausible antecedent to a thematic role, and (b) a plausible antecedent to a thematic role. A major drawback of the Thematic Role model is that it only considers concepts at the sentence level. It does not attempt to address how concepts can be inter-related throughout are large body of text.

The structure of English allows thematic roles to be located at different syntactic positions within a sentence. The result is a sentence with a different syntactic structure and more emphasis can be placed on certain roles. Although the syntactic structure is different, when constructed properly, the new sentence should describe the same event. Figure 2.10 gives an example. It is apparent that the antecedents of thematic roles can be extracted from a sentence based on their semantic contribution to that sentence, rather than the syntactic contribution. Table 2.2 outlines some of the thematic roles proposed by Sowa in [Sowa 2000].

(a) *A young toddler was running around his playroom. It was empty except for some chairs in one corner and some pet cats in the other. He chased a chair that he had run into before.*

(b) *A young toddler was running around his playroom. It was empty except for some chairs in one corner and some pet cats in the other. He bumped a chair that he had run into before.*

Figure 2.9: [Altmann 1999] Thematic Implausibility:

(a) *Bart threw a chicken at the house.*

(b) *A chicken was thrown at the house by Bart.*

Figure 2.10: Sentences Implying The Same Event

2.6 Logical Form and Quasi Logical Form

The Core Language Engine was developed at the *Stanford Research Institute* and *The Center for the Study of Language and Information* at Stanford University. The methods of anaphora resolution in the Core Language Engine [Alshawi et al 1989] are heavily motivated by its internal semantic representation, *logical form* and *quasi-logical form.* Quasi-logical form is based on *first order logic*, which has been used widely in the fields of philosophy and linguistics. The structure of logical forms is motivated by the desire to use and extend *first order logic*, which is well suited for modelling quantifier scoping and anaphora.

This section will discuss the Core Language Engine's fully scoped logical form, and a logical form where scoping rules are relaxed for reference resolution, quasi-logical form. Although logical form and quasi-logical form were designed to handle many types of reference phenomena such as *unscoped quantifiers*, *unscoped descriptions*, and *unresolved relations*, the phenomenon that will be focused on is *unresolved reference.* Resolution in

Agent
Beneficiary
Completion
Destination
Duration
Effector
Experiencer
Instrument
Location
Matter
Medium
Origin
Path
Patient
Point In Time
Recipient
Result
Start
Theme

Table 2.2: Thematic Roles

the Core Language Engine uses a set of reference resolution rules that propose possible logical forms that can transform a quasi-logical form statement into a logical form statement. Fully resolved logical forms must conform to the following set of properties:

- should be expressions in a disambiguated language.
- should be suitable for representing the *meanings* of natural language expressions.
- should provide a suitable medium for the representation of knowledge expressed in natural language, and they should be a suitable vehicle for reasoning.

Figure 2.11 lists some of the grammar rules used in the Core Language Engine to model the logical form language, and Figure 2.12 shows an example of logical form for the sentence *Every doctor visited Mary.*

Not all references can be resolved immediately using logical form. Sentences such as (1) *Most doctors read every article*, and (2) *the bishops arrived* contain references where the scope of quantification is not exactly clear. In sentence (1), does each doctor in

$$
\begin{array}{rcl}
\langle lf\_formula\rangle & \rightarrow & \mathrm{quant}(\langle quantifier\rangle, \langle variable\rangle, \langle restriction\rangle, \langle body\rangle) \\
\langle lf\_formula & \rightarrow & [\langle functor\rangle, \langle argument_1\rangle, \langle argument_2\rangle, \cdots, \langle argument_n\rangle] \\
\langle functor\rangle & \rightarrow & \langle atom\rangle \\
\langle quantifier\rangle & \rightarrow & \mathrm{forall}|\mathrm{exists}|\cdots \\
\langle restriction\rangle & \rightarrow & \langle lf\_formula\rangle \\
\langle body\rangle & \rightarrow & \langle lf\_formula\rangle \\
\langle argument\rangle & \rightarrow & \langle lf\_formula\rangle
\end{array}
$$

Figure 2.11: Logical Form Rules from the Core Language Engine [Alshawi et al 1989]

```
quant(forall,D,[doctor1,D],
        [past,
           quant(exists,E,[event,E],
                    [visit,E,D,mary1])])
```

Figure 2.12: Logical Form for *Every doctor visited Mary*

the *most doctors* set read all articles or does the *most doctors* set collectively read all articles. This question can also be considered for sentence (2). Is the arrival of each bishop from the *the bishops* a separate event (the distributive reading), or does a single arrival event encompass all the bishops (the collective reading). The quasi-logical form language extends the grammar of the logical form language to include rules that handle *unscoped quantifiers*, as in sentence (1), *under-specified relations*, as in sentence (2), as well as many other reference phenomena.

The logical form semantic representation is very well suited for modelling reference information where quantification is of key importance, and the discourse contains utterances where the exact scope of a quantifier is not clear. One of the biggest advantages to using logical form is that it allows us to use a large body of knowledge relating to first order logic. Although the Core Language Engine models some verb-structures[4]using logical form and quasi-logical form, it does not attempt to perform any temporal ordering on the verb structures (as in the case of the timeline models from Section 2.4). Another slight deficiency in logical form is how it handles "fuzzy" quantifiers such as *some*, *many*, and *few*. In the Core Language Engine, the threshold defining the boundaries of these quantifiers is given a definite value. Common sense dictates that certain quantifiers, such as *some*, are fuzzy, and subject to contextual factors and personal preference.

2.7 Conclusion

This chapter has examined a number of semantic representations that are used in the fields of natural language processing and knowledge management, such as semantic networks, conceptual hierarchies, logical form and quasi-logical form. The benefits and potential drawbacks were outlined for each representation. Chapter 3 will examine models of human knowledge from the perspective of cognitive science. Chapter 4 will present a new model of semantic representation that attempts to address the shortcomings mentioned in this chapter by combining the representations into a modular multi-level system. This modular system is easier to maintain and allows multiple anaphora resolution algorithms to operate on a corpus simultaneously.

[4] Using first-order logic to model a verb phrase, intuitively, does not seem the most natural method to model that knowledge.

CHAPTER THREE

Working & Long-Term Memory

Although there is still great debate about the exact nature and structure of working memory and long-term memory, the majority of cognitive psychologists agree that the two forms of memory are distinct in their behaviour and capacity [Logie 1996]. The human mind does not have infinite time and infinite working memory capacity. If a natural language processing system is to more closely model how humans process language, it would make sense for that system to be constrained by the limitations and behaviour of human memory. It is the intent of this thesis to stimulate more interest in memory model approaches.

This chapter will outline some of the various views on working memory and long-term memory. It will briefly discuss the evolution of theories of short-term and long-term memory as well as current theories in human memory. Several modern psychological models of short-term and long-term memory will be elaborated on, as well as some memory models with a computational approach.

3.1 Working Memory Models

Some of the earliest work with respect to working memory can be found in the works of William James (1905) [Richardson 1996]. James described working memory, then termed primary memory, as being limited in capacity and volatile in nature. Primary memory was considered to be a distinct system from long-term memory (called secondary memory at the time). Information was retained in primary memory by rehearsal. Rehearsal was also used to move information to and from secondary memory. Within this model, primary memory did not control the flow or manipulation of the information, it only provided a medium of storage.

Richardson continues by describing that during the 1960's, the theory of working memory, then termed short-term memory, was extended to include a control mechanism. This mechanism was responsible for the flow of information as well as processing. Instead of being used only for the storage of information, the space in short-term memory was shared with the processing of the control mechanism. Thus, in this model, there was a trade-off in working memory between processing power (in the control mechanism) and storage capacity.

The work of Baddeley on working memory is some of the most prominent. Gathercole and Baddeley outline in [Gathercole and Baddeley 1993] the structure and behaviour of Baddeley and Hitch's working memory model. Figure 3.1 gives a visual representation of their model. Gathercole and Baddeley state that the *central executive* is the most important component of the model. The central executive is responsible for controlling the flow of data within working memory, the retrieval of data from long-term memory and other memory systems, and the processing and storage of data. Baddeley expanded the model to include an *episodic buffer* in [Baddeley 2000]. In addition to the central executive, an additional two *slave systems* are also included in the working memory model, the *phonological loop* and the *visuo-spatial sketch-pad.* The *phonological loop* is responsible for verbal information while the *visuo-spatial sketch-pad* handles visuo-spatial information. Baddeley and Hitch used dual-task experiments in [Baddeley and Hitch 1974] to justify the separation of the two slave systems. They discovered that when a subject performed a verbal and a visual task concurrently, the individual could perform the tasks as efficiently as if the task were performed serially. When the number of tasks for a single slave system was increased to two tasks, the subject could not perform the tasks as efficiently as performing them one at a time.

Limits of Working Memory

The limits of working memory are as intensely debated as the structure of working memory. Early theories, such as Miller [Miller 1956], place specific limits on working memory. In

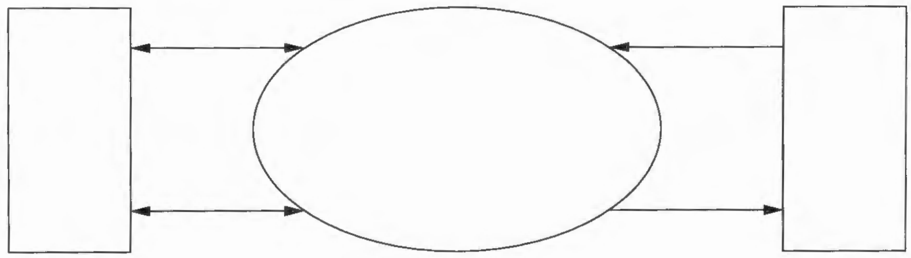


Figure 3.1: Baddeley and Hitch's Working Memory Model

Miller's view, working memory was seen as a short-term storage without any processing ability. More recent work, such as that found in [Baddeley 1986, Haarmann et al 2003], assumes that working memory is a multi-component system with *processing capacity* being inversely proportional to *storage capacity.* In [Baddeley 1990], Baddeley hypothesized that the span of working memory could partially be the result of the *refresh-rate* of items within the current memory span :

> If we assume that memory fades, then the memory span will be determined by the number of items that can be refreshed before they fade away. That number, of course, will depend on how rapidly the trace fades and on how long it takes to articulate each item and hence refresh each memory trace.

3.2 Long-Term Memory Models and Associativity

Federmeier and Kutas mention in [Federmeier and Kutas 1999] that although long-term memory is an integral component of sentence processing, the exact nature of how working-memory interacts with long-term memory information is still largely unknown. Just as there exists a debate as to the exact nature of working memory, differences in opinion also exist on how pieces of information are associated within long-term memory. Federmeier and Kutas outline two hypotheses, the *independent association hypothesis* and the *associative symmetry hypothesis.* The independent association hypothesis states that associations in memory are not bidirectional. That is, given that the recall of item **A** triggers the recall of **B**, **A** $\rightarrow$ **B**, does not necessarily imply that the converse, **B** $\rightarrow$ **A**, is true. In the associative symmetry hypothesis, given **A** $\rightarrow$ **B** implies that **B** $\rightarrow$ **A**. Using the independent association and associative symmetry hypotheses in the context of reading, when a word corresponding to a semantic concept is accessed through the reading of a sentence, the activation levels of neighbouring concepts may also increase.

3.3 ACT-R

The ACT-R theory of human cognition is rooted in ACT-E theory and ACT* theory which were introduced by John Anderson in [Anderson 1976, Anderson 1983], respectively. ACT-R models the interaction between two types of knowledge: procedural knowledge and declarative. Procedural knowledge involves rules that define human cognitive behaviour. Anderson formally calls these rules *productions* in [Anderson et al 2001]. Declarative knowledge encompasses factual information that defines behaviour of cognition, defined by Anderson as *chunks*. Examples of declarative knowledge are *the sky is blue* or *snow is white*. One of the major factors that influences cognitive performance in the ACT-R system is the granularity at which processing occurs. Production rules take at least 50ms and at most 500ms to fire.

Procedural Long-Term Memory

ACT-R is a goal-oriented system that uses productions to define the cognitive behaviour that acts upon declarative memory. Productions define actions such as retrieving information to be processed, as well as actions that define the manipulation of the retrieved information.

Declarative Long-Term Memory

In addition to the procedural memory, the ACT-R system also models declarative knowledge, that is, knowledge that is defined as being not factual and does not control the behaviour of cognition. Declarative memory is composed of *chunks* that are differentiated using a unique identifier. Each chuck has a *type* and may contain multiple *slots*, with each slot linking to additional *chunks*. A good example can be taken from [Group 2004]. Using the sentence *the dog chased the cat*, we can derive the following chunk of declarative memory:

```
Action023:
    isa chase
    agent dog
    object cat
```

In this example, the type of the outer chunk is *isa chase*, and the two slots of the chunk are filled with the chunks *agent dog* and *object cat.*

Declarative Memory Activation

In the ACT-R system, the retrieval of declarative chunks in memory is governed by the speed at which they can be accessed. In [Anderson and Matessa 1997], Anderson defines activation equations that predict the power law of learning and the power law of forgetting. The activation level, A_i, of a declarative memory chunk[1]is define as follows:

$$A_i = B_i + \sum_j A_j S_{ji} \tag{3.1}$$

where B_i is the base level activation of the chunk i, A_j is the activation of a chunk j within the current focus of attention, and S_{ji} is the strength of the association between chunk j and chunk i. The base level activation, B_i, models the recency and frequency of activation of the chunk i, and thus has a factor of decay associated with it. B_i is defined by ACT-R as:

[1]Anderson notes that the definition of *chunks* should not be confused with the definition from [Miller 1956]

$$B_i = \ln\left(\sum_{k=1}^{n} t_k^{-\omega}\right) \tag{3.2}$$

where t_k is the time since the kth[2]use of the chunk i, and ω is the activation decay. In [Anderson and Matessa 1997, Anderson et al 1998],the value of ω is fixed at 0.5. In order for a chunk of declarative memory to be retieved and brought into the current focus, the threshold of activation, σ, must be met.

3.4 Computational Models of Memory

A survey of memory would not be complete without examining those models created from a computational perspective. In particular, this section will examine the models of memory presented by Schank in [Schank 1986] and Hunt in [Hunt 1973]. Although the types of memory described by Schank and Hunt may have much information overlap with previously described models, it is still relevant to examine computational models along with psychological approaches. To some respect, the idea of memory modelling has been largely ignored in the field of computational linguistics.

Event Memory and Generalized Event Memory

Event memory contains semantic knowledge for particular events experienced in a person's life. Schank states that events can be such things as *going to Dr. Smith's dental office last Tuesday, and getting your tooth pulled* or *forgetting your dental appointment and having them call you up and charge you for it.* As a specific event remains in memory longer, the exact details of the event begin to become less salient, and eventually, the event may become a more generalized event or it may disappear entirely.

Generalized event memory, as the name describes, is a more generalized version of the

[2]In [Anderson and Matessa 1997], the index k is actually j. This was changed to prevent confusion with the index j in the equation for A_i. In addition, d is used as the decay rate, instead of ω and A_j is used for the activation of node j instead of W_j

event memory described earlier. Generalized event memory is modelled as a portion of memory that contains abstract events, i.e. events that have occurred numerous times and thus have a *template* associated with them. As events from event memory are brought into short term memory the associated generalized event is also brought in to aid in cognitive processing. One of the results from this behaviour is that events from event memory will become less and less salient and the more generalized event will only remain.

A Distributed Memory Model

In [Hunt 1973], Hunt describes a model of memory that builds upon the basic memory model containing only short-term memory and long-term memory. As shown in Figure 3.2, Hunt adds an *intermediate term* memory structure that resides between short-term and long-term memory, and *buffer memory* which is analogous to *sensory memory* in other literature. Intermediate-term memory stores information about the current situation or *episode*, and thus intermediate-term memory is volatile like short-term memory. The buffer memory is the most volatile of the structures, storing stimulus from sensory input, such as auditory input, for only brief periods of time. In contrast to Baddeley's model of working memory where information flow within working-memory is controlled by the *central executive*, Hunt's model places control within the respective memory substructures, Hunt notes that in his distributed memory model each memory component is *likely* to be associated with varying anatomical areas of the human brain, but does not provide evidence.

3.5 Neural Network Models of Associative Memory

Neural Networks, or more correctly *artificial neural networks*, attempt to model the behaviour of neurons within the human brain. Some of the earliest work on the modelling of artificial neurons can be attributed to McCulloch and Pitts in [McCulloch and Pitts 1943] and Hebb in [Hebb 1949]. Since the early work of McCulloch and Pitts, the field of artificial neural networks has developed into a mature field with large amounts of research in

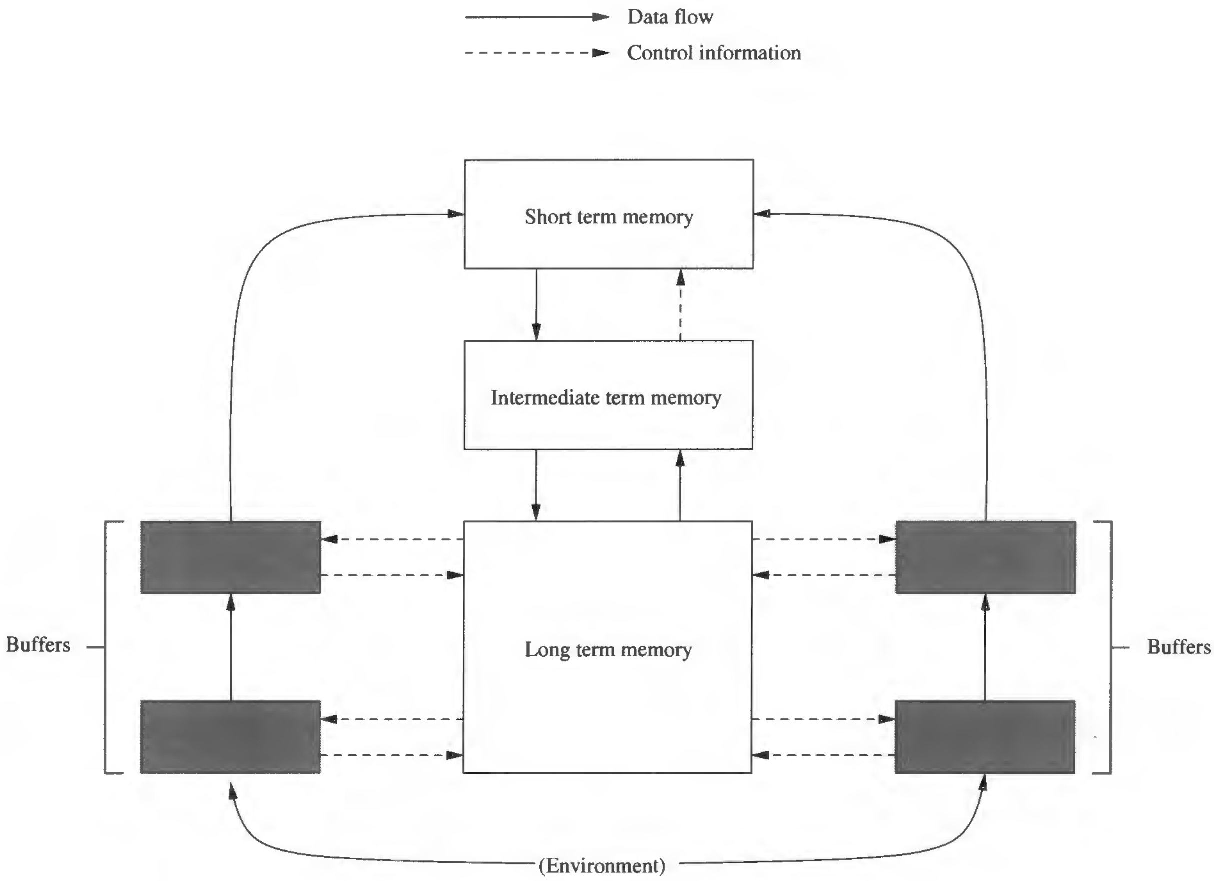


Figure 3.2: Hunt's Distributed Memory Model

areas such as *network topology* and *neuron structure.*

Artificial neurons are modelled after biological neurons. Axons send out signals to another neuron's dendrite. If the sum of the signals received by a neuron is greater than some threshold, then the neuron *fires*, sending signals along its axons to other neurons. Figure 3.3, adapted from [Russel and Norvig 1995], is an example of a typical biological neuron. Artificial neurons attempt to model neurons at the level of a single biological neuron. An artificial neuron does not attempt to model the actual physical chemical reactions occurring, rather, they model neurons at more of a *cause and effect* level.

A typical artificial neuron can be found in Figure 3.4. The input values for neuron i, A_j, can be thought of as the *dendrites.* The output value, A_i, is an axon. The strength of the link between node i and some node $j \neq i$ is modelled using the S_{ji} term. The bias

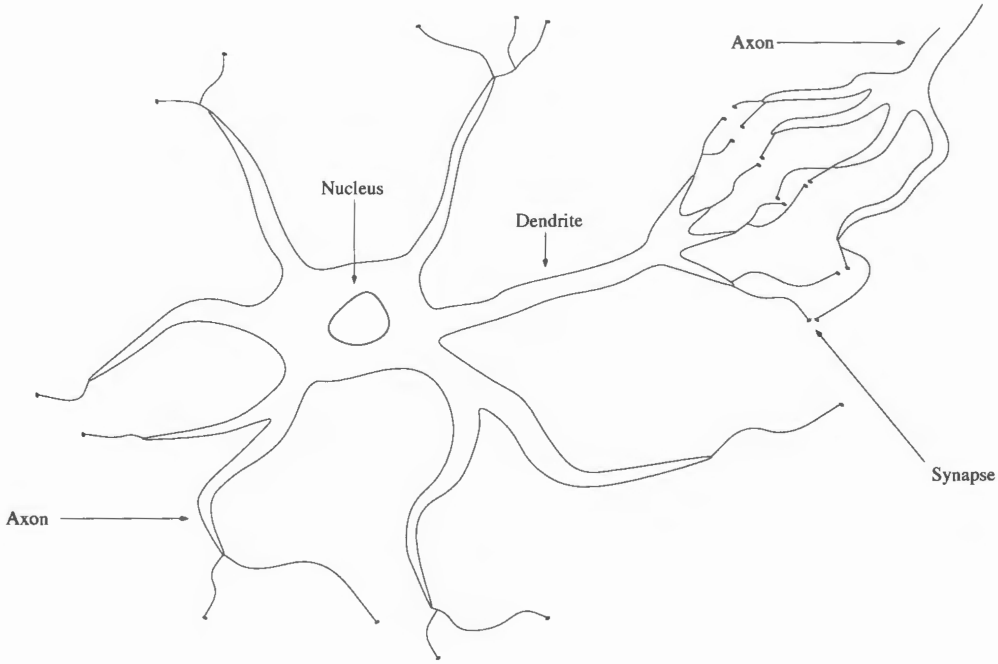


Figure 3.3: Biological Neuron

value, B_i, is often used to influence base-level activation, allowing A_i to be non-zero. The *activation function* of a neuron takes into consideration the input values, A_j, and their associated link strengths, S_{ji}, and generates a result f. Typically, the neuron activation function f for a node i is defined as follows:

$$f = B_i + \sum_{j=0}^{k} A_j S_{ji} \tag{3.3}$$

The final portion of an artificial neuron is the *threshold unit.* If the total signal received by the neuron is greater than some threshold, the neuron will *fire.* The actual value of the threshold is defined by a *threshold function*, $g(f)$. The resulting output from the neuron, is represented as A_i. In general, g can be any single-variable function, but the Equations 3.4-3.7 are some of the more commonly used.

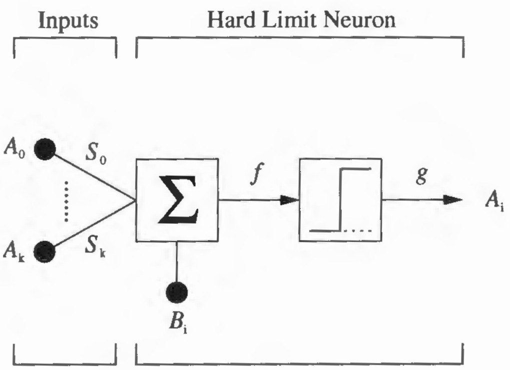


Figure 3.4: Multi-Input Neuron

$$hardlim(n) = \begin{cases} 1 & \text{if } n \geq 0 \\ 0 & \text{if } n < 0 \end{cases} \tag{3.4}$$

$$satlins(n) = \begin{cases} 1 & \text{if } n > 1 \\ -1 & \text{if } n < -1 \\ n & \text{if } -1 < n < 1 \end{cases} \tag{3.5}$$

$$logsig(n) = \frac{1}{1+e^{-n}} \tag{3.6}$$

$$tansig(n) = \frac{e^n - e^{-n}}{e^n + e^{-n}} \tag{3.7}$$

Neural Network Topology

A single set of inputs to a neuron is not very useful, or realistic. Of the 10^{11} neurons in the brain, each neuron is connected to 10^4 other neurons. One of the most common configurations of neural network topology is shown in Figure 3.5[3]. This three-layer configuration allows arbitrary functions to be represented, and is the most commonly used in *pattern matching* applications.

[3]Each circle in the diagram represents a multi-input artificial neuron.

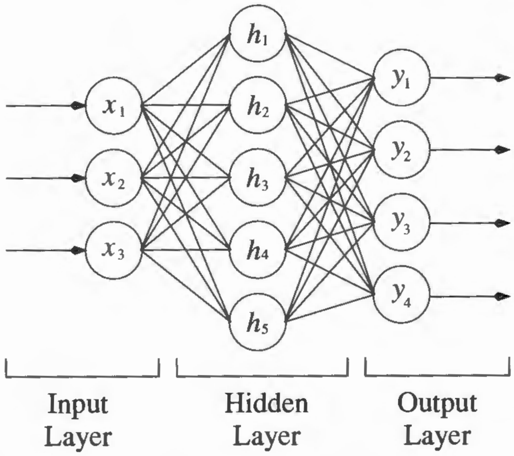


Figure 3.5: Multi-Layered Artificial Neural Network

Hebbian Unsupervised Learning

An artificial neuron by itself does not learn to perform complex tasks or match complex patterns. A learning strategy must also be applied. Two strategies of artificial neuron learning are *supervised* and *unsupervised* learning. This thesis will concentrate on the latter of the two learning strategies. Within the human brain, neurons are connected to many neighbouring neurons. The neuron must be capable of creating implicit associations without direct intervention. Before outlining how an artificial neuron can learn to make associations, it's important that neuron association be properly defined. Hebb's postulate from [Hebb 1949] states:

> When an axon of cell A is near enough to excite a cell B and repeatedly or persistently takes part in firing it, some growth process or metabolic change takes place in one or both cells such that A's efficiency, as one of the cells firing B, is increased.

Using Hebb's observation, the strength of the association between a neuron i and a neuron j can be realized using Equation 3.8 adapted from [Luger and Stubblefield 1998]:

$$S_{ij} = S_{ij} + \alpha A_i A_j \tag{3.8}$$

S_{ij} is the strength of the association between neuron i and neuron j, α is the learning rate of the semantic link, and A_i and A_j are the current activation levels of neuron i and neuron j, respectively.

3.6 Conclusion

This chapter examined numerous theories on the structure and behaviour of human memory. The evolution of working memory and long-term memory was examined from the perspective of cognitive psychology. Modern theories on human memory were also examined. Models of human memory from computation, such as neural networks, event memory, generalized event memory, and the distributed memory model were discussed. Baddeley's work with respect to working memory has demonstrated that there is a trade-off of capacity versus processing power in working memory. The ACT-R model established how associations between concepts can be modelled as well as the activation of those concepts. The discussion on neural network demonstrated how human memory can be modelled by using the behaviour of biological neurons as its basis. In Chapter 6, the work of Baddeley, the ACT-R model of activation, and neural network theory will provide a basis for the behaviour and structure of a memory model. A second model of activation, called Thompson's model, will also be introduced in Chapter 6. The results of testing using the ACT-R and Thompson model will be discussed in Chapter 7.

CHAPTER FOUR

Theory on Combining Semantic Structures

In Chapter 2, numerous semantic structures were outlined, as well as some of their shortcomings in their purest forms. A common thread was that each structure needed to be expanded in order to accommodate additional types of semantic information. The result is a structure that is possibly more complex, without the guarantee that the new structure can adapt to the dynamic nature of human knowledge and language. A structure that works today may not necessarily work tomorrow. From a software engineering perspective, modularizing the semantic structures makes the system easier to understand and easier to expand. As deficiencies are found in the semantic structures, new structures can be incorporated, and deficient structures can be removed.

4.1 Linking Semantic Structures

Intuitively, having completely disjoint semantic representations would not effectively model the nature of human knowledge. Research in human memory has shown that when concepts are activated in memory, related information may also be activated if the link between them is strong enough. Information from one semantic representation must somehow be linked to related information in another semantic representation. Semantic networks, which were introduced in Chapter 2, can be used to model the links between the semantic representations. Figure 4.1 expands on Figure 2.1 by connecting the semantic concept *ship* to various semantic representations such as quasi-logical form, conceptual graphs, conceptual hierarchy, and a timeline model. The information from a semantic representation could also be connected to other concepts nodes within the semantic network. In addition to the bare links, the strength of the links between concept nodes can also be considered by adding weights to each link. In theory, the more often the concept node in the semantic network is activated, the stronger the link will be to other concept nodes.

This is similar to the behaviour of synapses originally theorized by Hebb in [Hebb 1949]. In addition, activation levels for the concept nodes can be modelled using activation level models from artificial neural networks and ACT-R theory. The topology of the resulting network differs from the traditional artificial neural network topology in that there is no distinct input, hidden, or output layers. Rather, the concept nodes are connected in a non-specific fashion. The resulting network is commonly called a *localist network* in other literature. Single concepts are represented as a single node within the network. This representation differs from networks like artificial neural networks, where a single concept is represented as activations across a set of nodes, and nodes can represent more than one concept.

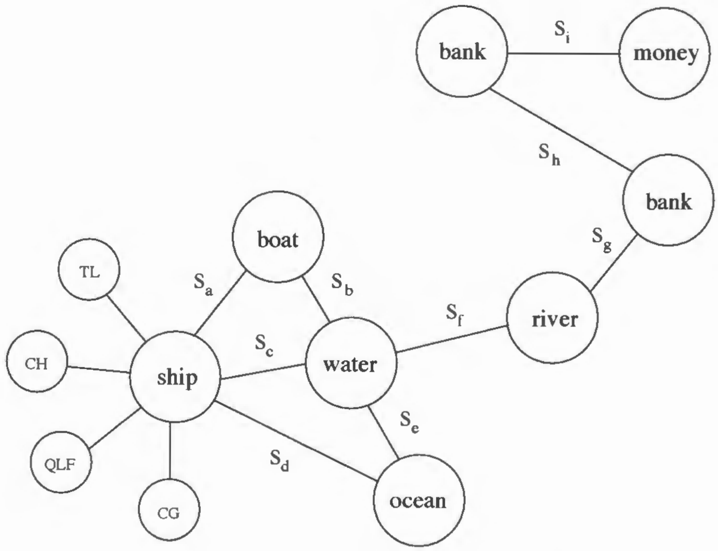


Figure 4.1: Semantic Network Linking Semantic Representations

Implicit Semantic Links

It is plausible that knowledge which is represented using one semantic representation is often intertwined with the knowledge within another. Figure 4.2 shows a good example of how multiple semantic representations collectively model the sentence *Once there was a shoemaker who worked hard and was very honest.* Notice that certain information such as *shoemaker:1* and *work:1* occur in more than one structure. We would like to some how connect this information together, and a semantic network would be a good method to achieve these connections. Concepts in the semantic network would be automatically linked due to the information within their semantic representations.

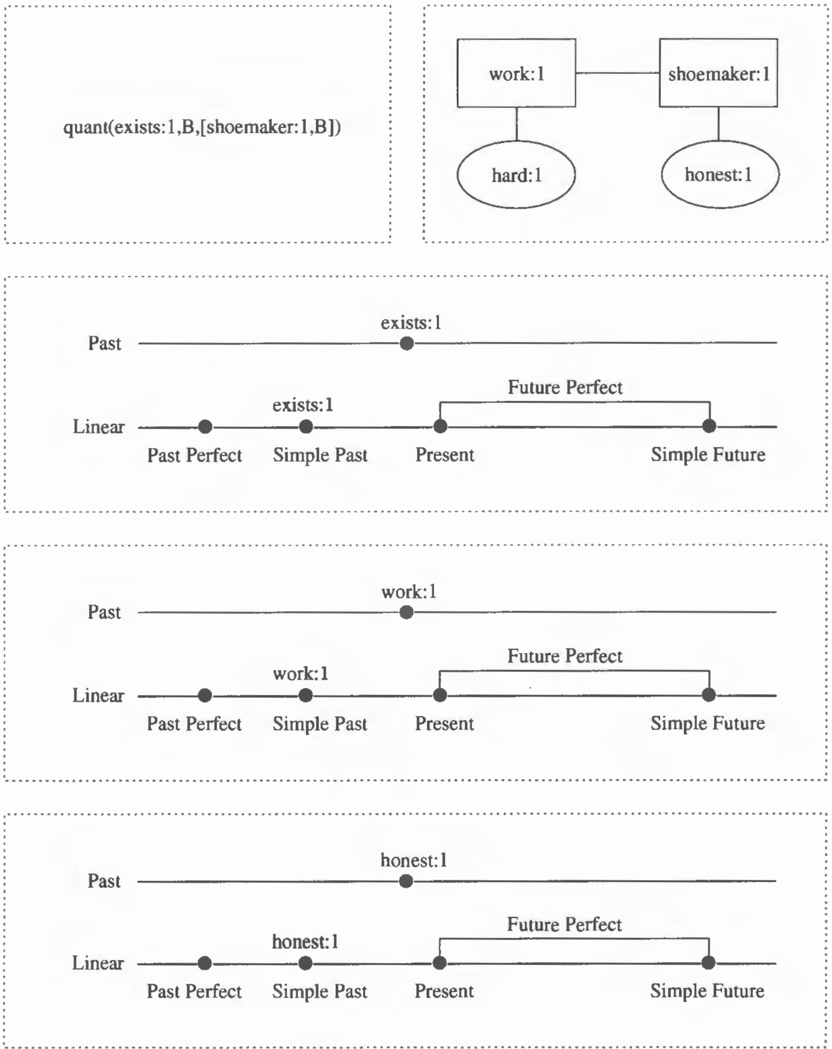


Figure 4.2: Multiple Semantic Representations

4.2 Modularization and The Human Mind

Creating a modularized model of semantic structures is not only a good engineering technique, it is also supported as model of the behaviour of the human mind by such psychological literature as [Foder 1983]. *Modularity of mind* theorizes that parts of the human mind are modular in nature and act autonomously with respect to other modules in the mind. To some degree, the modularity is thought to be genetically determined. Foder's model of human mind requires that modules be specialized in their domain, encapsulate their information from other modules, and have limited outputs to other modules. These requirements are a natural result of the model described in this chapter.

4.3 Conclusion

This chapter has introduced a modular model of semantic representation that addresses the drawbacks of existing semantic representations by connecting them using a semantic network. This model allows new semantic representations to be added and removed from a system without impacting the existing ones. It also addresses the desire to expand a semantic representation to deal with new forms of knowledge. Instead of expanding to a current semantic representation, a new semantic representation can be created and *plugged into* an existing system. This modular separation of semantic representations also permits the development of anaphora resolution algorithms for each semantic representation independently. It is apparent that in order to combine the semantic structures from Chapter 2 using a semantic network, a theory of the exact nature of the network must be considered. The link strength between concepts needs to be modelled, as well as the activation levels for each node. Chapter 3 discussed theories on how link strengths and activation levels are determined in the ACT-R system [Anderson et al 2001] and using traditional artificial neural network theory. Chapter 6 will outline how activation theories will be used in conjunction with a semantic network to define the behaviour of the model proposed in this chapter.

CHAPTER FIVE

Anaphoric Reference and Reference Resolution

Anaphoric reference is a linguistic mechanism with which reference can be made to objects that have been introduced at an earlier point. References are typically made with pronouns or different variations of definite/indefinite articles within a noun phrase. Anaphoric references are also used to reference verb phrase structures. Anaphora, especially in the case of pronouns, often can be resolved by scanning backwards through a corpus of text until the first noun phrase that matches such features as number and gender is found, although Barbara Grosz demonstrated in [Grosz 1977] that this technique can break down.

5.1 Types of Anaphoric Reference

Traditionally, anaphoric reference is observed in the use of pronouns such as *he*, *she*, or *it*. But within recent decades, there have been numerous proposals to extend the definition of anaphoric reference to include other linguistic phenomena such as *verb phrase ellipsis* [Grosz 1977, Hardt 1997, Nash-Webber and Reiter 1977, Ginzburg and Cooper 2001], *presupposition* [Piwek and Krahmer 2000, Geurts 1999], and temporal anaphora [Partee 1984]. In the context of this thesis, the domain of anaphoric references will be restricted to pronominal references of noun phrases.

5.2 Problems in Anaphora Resolution

As stated before, in many cases resolving the antecedent for a pronoun is as simple as searching backwards in a body for the first noun phrase that matches based on such attributes as *gender* or *number*. But in some cases, a more complex model discourse must be modelled in order to resolve a pronoun reference:

John had a son named Bob. His son is an excellent skier.

In this example, a knowledge base about parent-child relationships must be known in order to resolve the reference implied the possessive pronoun *his*. Even trivial references can be more complex by introducing existential quantifiers, as illustrated by Partee in [Partee 1984]:

Every farmer who owns a donkey beats it.

The pronoun *it* does not just reference a single donkey, the pronoun references multiple instances of a donkey. The next example, adapted from [Sidner 1983], demonstrates where this method of resolution can break down:

$\boxed{\textit{My neighbours}}_1$ *have* $\boxed{\textit{a monster Harley 1200}}_2$. $\boxed{\textit{They}}_3$ *are really huge but gas efficient bikes.*

In the second sentence, if an individual was to read just the pronoun *they*, their initial preference for the reference may not be *a monster Harley 1200* based on number alone. In this context, a common preference for the pronoun *they* would be *my neighbours.* After reading the remainder of the second sentence, it is apparent that this conclusion was incorrect. Given the additional context, common knowledge concludes that the neighbours are not motorcycles

5.3 Anaphora Resolution Algorithms

Determining the antecedent of an anaphor is central to the study of anaphoric reference. Over the past few decades much research has involved creating computational methods to resolve these references. Works such as [Sidner 1979, Sidner 1983], [Grosz 1977, Grosz and Sidner 1986], and [Carter 1985, Carter 1990] have concentrated on the study of discourse and the theory of anaphora within a discourse, while [Hobbs 1986, Brown 2003, Mitkov 1998] have focused more specifically on pronominal anaphora resolution. The next few sections will outline Carter, Hobbs, Brown, and Mitkov's approaches to pronominal anaphora resolution.

Hobbs' Approach and Brown's Algorithm

In [Hobbs 1986], Hobbs outlines a simple algorithm for the resolution of pronouns, and although naïve, it provides goods results. The algorithm works by starting at the location of the pronoun and working back through the parse tree in a breadth-first manner until a suitable antecedent match based on gender and plurality is found. When tested on 300 occurrences of references in selected corpora, the algorithm had a success rate of 88.7% in resolving the anaphoric reference. Hobbs notes though that in over half of the cases, there was only one plausible antecedent.

Hobbs analyzed the results further and went on to consider the results for the cases when there was more than one plausible antecedent. Of the 132 cases where an antecedent conflict existed, 98 were resolved by the algorithm, thus a 74.4% success rate. Hobbs goes on to improve the naïve algorithm by adding simple restrictions for resolving pronouns, such as *dates can't move*, *places can't move*, and *large fixed objects can't move.* Without these restrictions, the success of the resolution algorithm was, 81.8%, overall. When the selectional restrictions were used, a 91.7% success rate was achieved.

In [Brown 2003], Brown outlines an algorithm for resolving noun phrase references that is a variation on Hobbs algorithm. Figure 5.1 illustrates the algorithm in pseudo-code[1].

Brown's algorithm has the benefit of not specifying how a reference is resolved when there are multiple antecedents for a single noun phrase, which consequently, allows the implementor to choose how the antecedent can be resolved.

Carter's Approach

In his PhD thesis, [Carter 1985], Carter describes in [Carter 1990] the approach used in the SPAR system . The SPAR system initially starts by resolving semantic and syntactic

[1] In Figure 5.1, NP is an abbreviation for *noun phrase.*

```
IF the NP is a proper name THEN
    ATTEMPT to identify the reference in the knowledge base
    IF no antecedent is found THEN
        CREATE a new reference in the knowledge base
ELSE IF the NP is an indefinite NP THEN
    CREATE a new reference in the knowledge base
ELSE IF the NP is a reflexive pronoun then
    SET the reference to the subject of the clause
ELSE IF the NP is a pronoun THEN
    CHECK NPs that precede for number/gender/person agreement
    check NPs in previous sentences in the same manner
ELSE IF the NP is a definite NP THEN
    CHECK NPs that precede for number/gender/person agreement
    CHECK NPs in previous sentence in the same manner
    IF no is antecedent is found THEN
        CREATE a new reference in the knowledge base
```

Figure 5.1: Brown's Anaphora Resolution Algorithm

issues without concerning itself with potential anaphoric references. Multiple structures can result from this process depending upon the word-sense of the words within a sentence. Given the sentence *He picked up a jack*, Carter theorizes two possible structures. One structure where *jack* is interpreted as a playing card, and the second where it is a tool used to raise an automobile. According to Carter's algorithm, the pronoun *he* is left unbound, and will be dealt with in further stages.

After the initial structures are generated, they are reprocessed and given scores based on factors such as repeated relevant information and its influence on syntactic structure. For example, in Figure 5.2, (b) would be given a higher score than (a) because *a telescope* is used for seeing things, thus it is more highly related to the verb *saw* than the noun phrase *a man*. After assigning scores, the algorithm proceeds to use anaphora resolution rules which Carter describes as being similar to those found in [Sidner 1979].

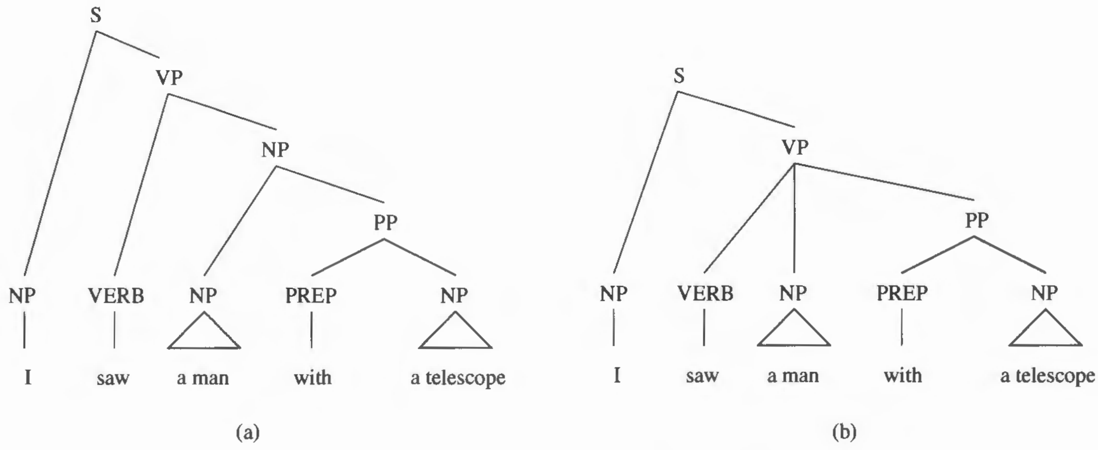


Figure 5.2: Two Parsings of *I saw a man with a telescope*

Mitkov's Work

In [Mitkov 1998], Mitkov outlines an anaphora resolution algorithm that uses scoring factors to determine a plausible antecedent to a reference. The scoring factors are based on the analysis of what Mitkov terms *indicators.* Indicators can be such things as the *definiteness* of the possible antecedent, the *givenness*, *indicating verbs*, *lexical reiteration*, *prepositional position*, and *referential distance.* The domain of possible scores is $\{-1, 0, 1, 2\}$, with varying values being chosen for each *indicator* class. Figure 5.3 outlines Mitkov's algorithm in pseudo-code format. Mitkov claims a success rate of 89.7% with this algorithm.

5.4 Resolution Failure

Many resolution algorithms make the assumption that the antecedent of an anaphora must be resolved. Levine hypothesizes in [Levine et al 2000] that there are conditions under which readers fail to resolve the anaphoric reference, yet are still able to comprehend the text. Levine showed that if the antecedent was salient and distant enough from the point of reference, readers were content with not resolving the reference if it was not disruptive to the comprehension of the text. Although this finding could have a big impact on how

EXAMINE the current sentence and the two preceding sentences
SELECT the noun phrases which agree in gender and number
APPLY the antecedent indicators to each candidate and assign scores
IF two candidates have an equal score THEN
    SELECT candidate with the higher score for immediate reference
IF immediate reference does not hold THEN
    SELECT the candidate with higher score for a collocational pattern
IF collocational pattern suggests a tie or does not hold THEN
    SELECT the candidate with higher score for indicating verbs
IF this indicator does not hold THEN
    SELECT the most recent candidate

Figure 5.3: Mitkov's Anaphora Resolution Algorithm

anaphora resolution algorithms will work in the future, it does not begin to explain what a comprehensible piece of text is.

5.5 Conclusion

This chapter has given an overview of the problem of anaphoric reference with respect to pronouns, as well as algorithms that address reference resolutions. Chapter 6 will describe an anaphora resolution algorithm that uses the theories on working memory and activation, introduced in Chapter 3, to create a list of possible antecedents for a pronoun. The algorithm will then use the idea of feature set scoring from Carter to form a basis for resolving conflicts when multiple plausible antecedents exist. Chapter 7 will outline the results of the algorithm when tested on *The Three Brothers* and the *Rumpelstiltskin* corpora using the ACT-R model and the Thompson model of activation.

CHAPTER SIX

System Modelling

If you don't gosub a program loop, you'll never get a subroutine.

- KRYTEN *(Justice)*

Throughout Chapters 2, 3, and 4, varying theories on semantics representation, the structure of human memory, and its behaviour have been discussed. This chapter will focus on combining parts of these various theories to solve the problems of anaphora resolution outline in Chapter 5. The model of long-term memory and working memory will be outlined as well as the grammar rules and their interaction with the memory models. The algorithm for anaphora resolution will also be discussed. For the purposes of this thesis, the semantic network will be the only semantic representation that is implemented. The implementation of other semantic representations, such quasi-logical form and concept graphs, will be deferred to future work. The implementation and subsequent testing of the semantic network will provide baseline results from which this future work can be compared to.

6.1 Modelling Long-Term Memory

In this thesis, long-term memory will be modelled using a combination of the semantic network theory described in Chapter 2 and the neural network theory described in Chapter 3.

Semantic Networks for Long-Term Memory

Semantic networks are undirected graphs with strengths associated with the links between nodes. Within the context of this thesis, as was described in Chapter 4, semantic networks will not be used directly to store semantic information, rather, they are used to link the other existing semantic structures (i.e. conceptual graphs, classification hierarchies, event timelines, and quasi-logical form). This decision is intended to achieve a behaviour similar to that observed in neural network theory. As the activation levels of a node within the network increase, the activation of neighbouring nodes will also increase. Semantic markers will be used to uniquely identify semantic structures and concepts.

Table 6.1 outlines the Prolog predicates. In the *nn_semNode* predicate, *SemMarker* signifies the semantic marker for the node, *Activation* holds the current activation, and *ActHistory* contains the activation history of the node. In the *nn_semLink* predicate, *SemMarker:1* and *SemMarker:2* identify the two structures or concepts being linked, and *Strength* is, of course, the strength of the link. The *nn_semNode* and *nn_semLink* predicates provide all that is required to build and modify a semantic network.

Predicate	Description
`nn_semNode(+SemMarker,+Bias,+Activation,+ActHistory)`	Semantic Node
`nn_semLink(+SemMarker:1,+SemMarker:2,+Strength)`	Semantic Link

Table 6.1: Semantic Network Predicates

Activation Level Models

The two different models of semantic node activation levels will be used and tested in this thesis: (1) The ACT-R model for activations of declarative memory, and (2) a model derived empirically, called Thompson's model. From the ACT-R model, Anderson's model for the activation level and activation decay for declarative memory will be used, and will be based on Equations 3.1 and 3.2, as described in Chapter 3. Since the activation level of the ACT-R equation is unbound, the *satlins* threshold equation, Equation 3.5, will be applied to bound the resulting activations to the range $-1 \geq A_i \leq 1$.

Thompson's model will be based on Equation 3.3 for the activation function of a semantic node, Equation 3.7 for threshold function, and the following equation for the fading the activation:

$$A_i = \omega A_i \tag{6.1}$$

where ω is the *decay rate* of node i's activation. The base level activation of node i will be set to $B_i = 0.0$. It must be noted that the decay equation for Thompson's model is not applied during node activation, as in the case of the ACT-R model. Rather, decay will occur, due to mental processing in working, which will be described in more detail later in this chapter.

6.2 Working Memory

If the activation level of node i is greater than some activation threshold, σ, node i will be brought from long-term memory into *working memory.* Working memory acts as a repository for concepts that are easily accessible for mental processing.

Working Memory Structure

In this thesis, working memory does not contain the actual structures that represent the currently active concepts, rather, working memory is conceptualized as a list containing *semantic markers.* The semantic markers act as links to the concepts within long-term memory. As semantic concepts are activated, they are placed within the list representing working memory, and as they decay, they are removed from the list.

Semantic Node Behaviour

The working memory model that was modelled is based on the model described by Baddeley in [Baddeley 1986, Baddeley 1990, Gathercole and Baddeley 1993]. From Baddeley, the theory on working memory capacity was used. The limits of working memory will be based on the contention between storage capacity and processing time. Storage and processing are inversely proportional to each other, and thus, processing will affect how quickly active concepts in working memory will fade. In this thesis, processing will be restricted to the application of grammar rules only. As grammar rules are used (i.e. mental processing), the activation levels of concepts in working memory will fade. This fading effect of grammar rules gives the *storage* versus *processing* behaviour described in Baddeley's model. When the activation level of node i falls below σ, node i will be removed from working memory.

Semantic Link Behaviour

As will also be described in Chapter 7, the semantic links between semantic nodes do not exist when the system is initialized. Semantic links are created between the nodes in working memory after each sentence is parsed and are given an initial link weight, S_{ij} and S_{ji} (a semantic link for each direction). If the links already exists between nodes, new links will not be created.

Although the strengths of links between the nodes are created with the same initial value, they are updated independently after this creation. When a semantic node i is activated into working memory, the new strength of the link to node j is updated. The equation for Hebbian learning, Equation 3.8, will be used for calculating the new semantic link strength:

$$S_{ij} = S_{ij} + \alpha A_i A_j$$

6.3 Prolog Model of Working Memory

The Prolog model of working memory will be a functor, **`wm_workingMemory/1`**, with a single list as an argument. The list will contain the semantic markers of the concepts that are currently active in working memory. The current activation level of a concept will not be contained within the list, rather, the semantic marker will be used to look-up the activation in the semantic network described earlier. For example, given that concepts *rimmer:1*, *lister:1*, and *kryton:1* are active in working memory, the following functor would result:

$$\texttt{wm\_workingMemory}([\texttt{rimmer}:1,\texttt{lister}:1,\texttt{kryton}:1])$$

6.4 English Grammar Rules

One of the major advantages of using Prolog as an implementation language, is that it allows the use of a Definite Clause Grammar to specify grammar rules. Definite Clause Grammars also reduces the amount time required to code grammar rules by eliminating the need to specify mechanism for consuming words from a sentence while parsing. For example, rather that using the rule

$$\texttt{adj}([\texttt{long} \mid \texttt{B}],\texttt{B}).$$

to process the adjective *long*, we can use the Definite Clause Grammar rule

$$\texttt{adj} \rightarrow [\texttt{long}].$$

Definite Clause Grammars are much more elegant because, notationally, they are very similar to context-free grammars.The result being that the source code will be more readable, easier to maintain, and less prone to errors.

The cost associated with using a Prolog grammar rule will be explicitly modelled within the grammar rules. Prolog allows us to add goals to Definite Clause Grammar rules that, when expanded into regular Prolog predicates and clauses, do not consume words from the input string while parsing. As an example, consider the following grammar rule:

$$\texttt{sent}(\texttt{sent}(\texttt{NP},\texttt{VP})) \rightarrow \texttt{np}(\texttt{NP}),\texttt{vp}(\texttt{VP})$$

Adding a processing cost, the resulting rule would be something similar to the following:

$$\texttt{sent}(\texttt{sent}(\texttt{NP},\texttt{VP})) \rightarrow \{\texttt{nn\_fadeNodes}\},\texttt{np}(\texttt{NP}),\texttt{vp}(\texttt{VP})$$

Here, the predicate `nn_fadeNodes` is a predicate that updates the activation levels of concepts in working memory.

The placement of the cost predicate within the grammar rule is important. By placing `nn_fadeNodes` at the front of the rule, the cost is incurred as soon as the rule is used. This placement creates the behaviour that as more backtracking is performed on grammar rules, the more complex the processing. Concepts will fade much more quickly from working memory when backtracking occurs, as opposed to no backtracking. If `nn_fadeNodes` was placed at the end of the grammar rule, the cost would only be incurred after the successful completion of a grammar rule.

6.5 Annotated Parse Tree Model

In this thesis, the modelling of parse trees will be an extension to the model found in [Sterling and Shapiro 1999], where parse trees are stored in an embedded-functor form. For example, given the following Definite Clause Grammar rule

$$\mathtt{sent} \rightarrow \mathtt{np}, \mathtt{vp}.$$

the equivalent Definite Clause Grammar rule with parse trees embedded would be

$$\mathtt{sent(sent(NP, VP))} \rightarrow \mathtt{np(NP), vp(VP)}$$

Sterling and Shapiro's parse tree model will be extended to also include the antecedent for pronoun references. Parse trees of the form

$$\mathtt{noun(N)}$$

would take the following form for pronouns, where "Ant" is bound a plausible antecedent for the pronoun or "null" if no antecedent exists.

$$\mathtt{noun(N, ant(Ant))}$$

Since this extension to the parse tree model still adheres to the syntax of the original model, existing *pretty printers* can be used on the model without any modifications. Figure 6.1 shows two example parse trees with an unresolved and a resolved antecedent.

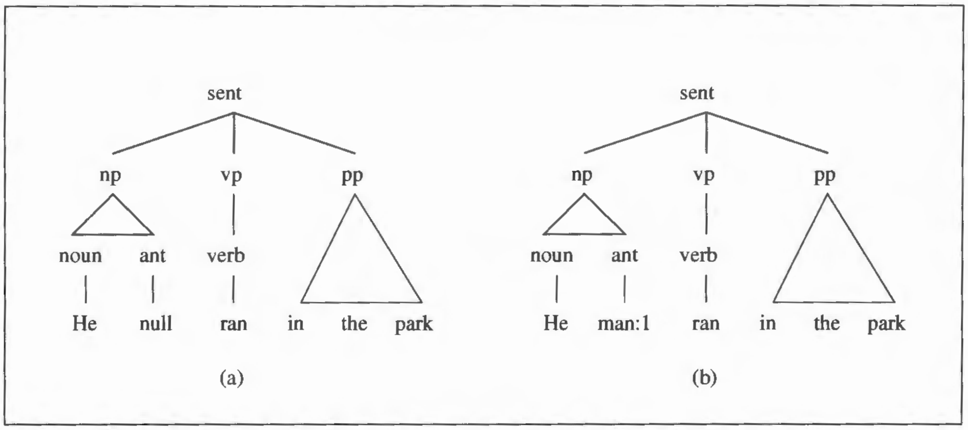


Figure 6.1: Annotated Parse Trees with (a) Unresolved and (b) Resolved Antecedents

6.6 Lexical Feature Sets

Lexical feature sets have been used in various theories of natural language processing such as Generalize Phrase Structure Grammars [Bennet 1995], in the Core Language Engine [Alshawi et al 1989], and the LangEng project [Brown et al 2001]. Features sets allow parsers to restrict parsing based on set of semantic attributes inherent to certain words. For example, in addition to semantic meaning, nouns also have attributes that imply the *gender*, *person perspective*, and *plurality* of the noun. The feature set syntax used in this thesis will be an extension on feature set syntax of the LangEng Project.

Each noun entry will contain two feature sets. The first feature set, the *lexical* feature set, will contain lexical entries such as the noun's *case*, *category*, and so forth, The second feature, called the *semantic* feature set, will contain semantic features such as the *gender*, *number*, and *person* perspective. The following is an example of a noun entry with the above features sets:

$$\mathtt{noun}([\mathtt{case} : \mathtt{nom}, \mathtt{cat} : \mathtt{np}], [\mathtt{gender} : \mathtt{masc}, \mathtt{num} : \mathtt{sing}, \mathtt{person} : 3]) \rightarrow [\mathtt{he}]$$

Feature sets are used by anaphora resolution algorithms to resolve references when multiple antecedents exist. Section 6.7 will discuss how the feature sets are used with an anaphora resolution algorithm. Tables 6.2 and 6.3 illustrate the noun feature sets that were used in this thesis.

Feature	Value Set	Description
sem	prolog atom	semantic marker
cat	$\{np\}$	marks lexical category
case	$\{nom, acc, gen, dat, abl, loc, tmp\}$	morphological case
sound	$\{soft, hard\}$	first consonant sound
type	$\{common, proper, gerund, pronoun, rlfx\_pronoun\}$	noun type

Table 6.2: Lexical Feature Set

Feature	Value Set	Description
gender	$\{masc, femn, neut\}$	gender of a noun
num	$\{sing, plur, mass\}$	plurality of a noun
person	$\{1, 2, 3\}$	person perspective

Table 6.3: Semantic Feature Set

6.7 Anaphora Resolution Algorithm

In Chapter 5, various anaphora resolution algorithms were discussed. A common thread between all the algorithms is that in the absence of multiple antecedents for a reference, the correct antecedent is identified, with the exception of those cases outlined in [Levine et al 2000].

The algorithm that was implemented is combination of the ideas outlined by Brown in [Brown 2003], Carter in [Carter 1985, Carter 1990], and the results from Koh and Clifton in [Koh and Clifton 2002]. Brown's algorithm will provide resolution for references that only have one antecedent. From Carter, the idea of *scoring factors* to influence how conflicts between lexically identical words are resolved was used.

The scoring factor for a semantic concept i will be a combination of the current activation of node i within working memory and relative similarity of the semantic feature set of concept i to the semantic feature set of a pronoun j:

$$Score_i = A_i + F_{ij} \tag{6.2}$$

where F_{ij} a score based upon the relative similarity of the feature sets of concept i and pronoun j. The value of F_{ij} is computed by starting with an initial score of 1.0. Each entry in the semantic feature set of concept i that matches an entry in pronoun j increases F_{ij} by a factor of ϕ, and each entry that does not match decreases F_{ij} by a factor of $\psi = \frac{1}{\phi}$.

The current activation of node i will influence whether it is compared to pronoun j in the anaphora resolution algorithm. The algorithm will ignore concepts with an activation of less than σ. Figure 6.2

```
FIND the semantic concepts currently in working memory
IF (at least one concept was found) THEN
    FOR ( each concept i found) DO
        SET Score_i = 1.0
        FOR (each feature in the feature set of concept i) DO
            COMPARE the feature value to the feature of the pronoun
            IF (the values match) THEN
                SET Score_i = Score_i * φ
            ELSE
                SET Score_i = Score_i * ψ
        ADD the activation level(A_i) of node i to Score_i
    FIND the concept with the highest value for Score_i
    SET the antecedent of the pronoun to that concept
ELSE
    SET the antecedent of the pronoun to NULL
```

Figure 6.2: Pronoun Reference Resolution Algorithm Pseudo-code

6.8 Summary of Chapter

This chapter has examined models for long-term and working memory and how they were realized in Prolog. It has also given a general overview of the format of the grammar rules that are used as well as the *feature sets* and *annotated parse trees* that accompany the rules.

In Chapter 7, the model described in this chapter will be tested against a number of corpora with varying decay rates for the semantic nodes.

CHAPTER SEVEN

System Testing

This chapter will cover the methodology and procedures used for testing the anaphora resolution algorithm described in Chapter 6 using the activation model from ACT-R and the Thompson activation model. Various activation decay rates will be tested for each model. The results will be compared to human-based resolution.

7.1 Overview of Testing Methodology

The anaphora resolution algorithm will be tested against a corpus selected from the Grim Brothers library found at [Ockerbloom 2006]. The selected body of text was slighted modified from their original form to facilitate ease of parsing while retaining the spirit of reference placement. The modified text can be found in Appendix 8.3.

Each corpus will be tested independently and not have influence on the tests of the other two corpora. That is to say, the working memory, and long-term memory will be reset to a default configuration for each test phase. The bodies of text will be tested a number of times each with different decay rates, ω, for the semantic nodes within long-term memory.

7.2 Testing Platform

The testing was performed on a 1.8GHz PowerPC G5 1.25 GB RAM under Mac OS X v10.4.4 using SWI-Prolog v5.4.7.

7.3 Decay Rates

Since the decay rate, ω, affects the activation levels in the ACT-R model differently than those in the Thompson model, different sets of decay rates were chosen. The sets of decay rates were chosen in such a way that they presented a broad spectrum of the behaviour for each model. The value of ω ranged from values that caused short activation times, i.e. fast activation decay, to values that caused low activation, and thus caused the contents of working memory to be quite high.

In the ACT-R model, the values of ω ranged from 0.05 to 0.30. The decay rate of $\omega = 0.5$, which is used in ACT-R, was not chosen because preliminary testing showed that the value caused an extremely high level of decay, which resulted in a large number of pronouns being unresolved. This extreme decay is most likely due to the fact that the ACT-R model may not be 100% compatible with a neural network-type model.

The decay rates in the Thompson model ranged from 9.90999×10^{-1} to 1.0. Although, a decay rate of $\omega = 1.0$ would imply no decay, that is not actually the case. Since the current activation is also based on neighbouring semantic nodes and the weight between the nodes, a certain amount of decay will still occur.

7.4 Default Memory Configuration

Initially, the contents of working memory were empty. This initial state of working memory was represented in Prolog by an empty list as the argument of the *wm_workingMemory* functor:

$$\text{wm\_workingMemory}([])$$

Long-term memory, was represented as a neural network, initially contained nodes for all possible nouns that can be parsed by the system. Each node, i, had an initial activation $A_i = 0.0$, and bias $B_i = 0.0$. The links between nodes did not exist, rather, they were created as describe in Chapter 6.

7.5 Testing Procedure

The ACT-R and Thompson activation models of system were tested against all of the sentences from the *The Three Brothers* corpus in sequence using various rates of decay.

The following items were tabulated during testing:

- results of the anaphora resolution algorithm were tabulated against the expected results outlined in the tagged corpora of Appendix B
- the maximum capacity of working memory across all decay rates for each activation model
- the growth of working memory capacity over the course of parsing the corpus.

7.6 Anaphora Resolution Results

The results of the anaphora resolution algorithm are outlined in Figures 7.1 and 7.2 for the ACT-R model and Figures 7.3 and 7.4 for the Thompson model.

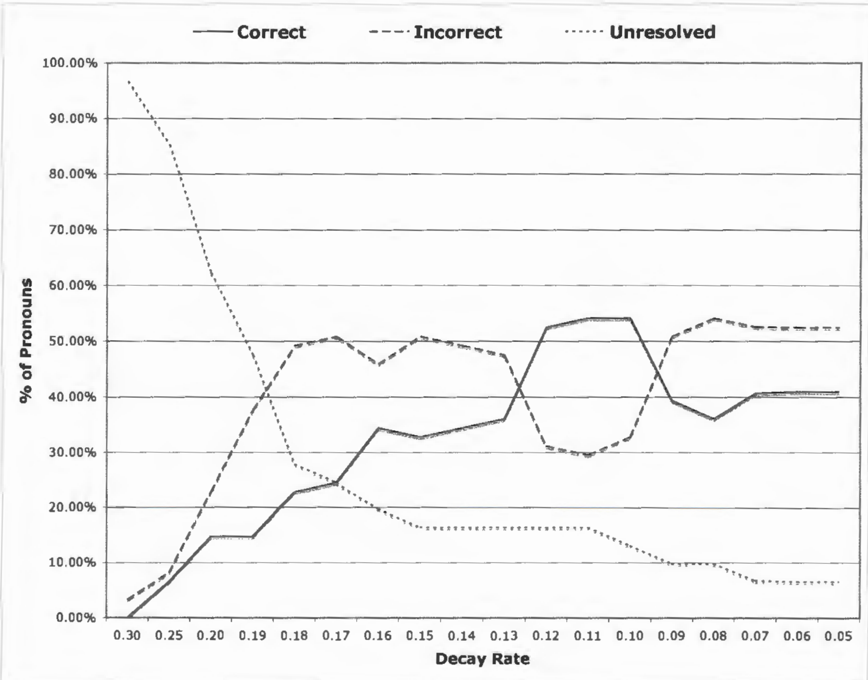


Figure 7.1: The ACT-R Model Resolution Results - 3 Brothers

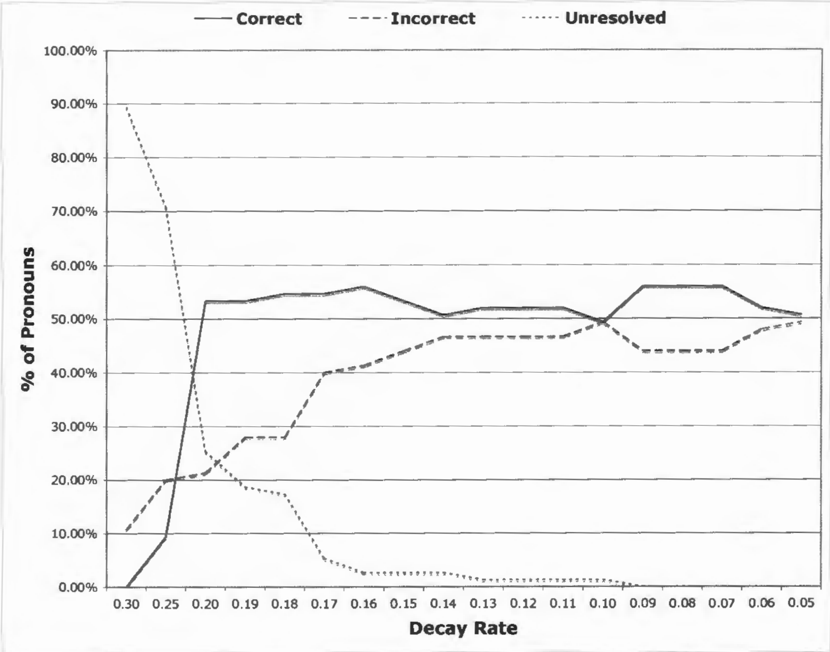


Figure 7.2: The ACT-R Model Resolution Results - Rumpelstiltskin

7.7 Working Memory Capacity Results

In addition to the results of the anaphora resolution algorithm, the behaviour of working memory was also observed for each decay rate. The *working memory max capacity* was the largest number of nouns that were observed to be in working memory at the end of each parsed sentence. Figures 7.5, 7.6, 7.7, and 7.8 outlined the maximal working memory contents for the ACT-R model and the Thompson model, respectively. A comparison of the growth rate of working memory capacity using the optimal decay rate is outlined in Figures 7.9 and 7.10.

7.8 Discussion of Results

A baseline comparison between the ACT-R model and the Thompson model can be made by considering the decay rates that give an equivalent, non-zero, number of unresolved pronouns. For *The Three Brothers* corpus, given the decay rate of 0.09, the ACT-R model

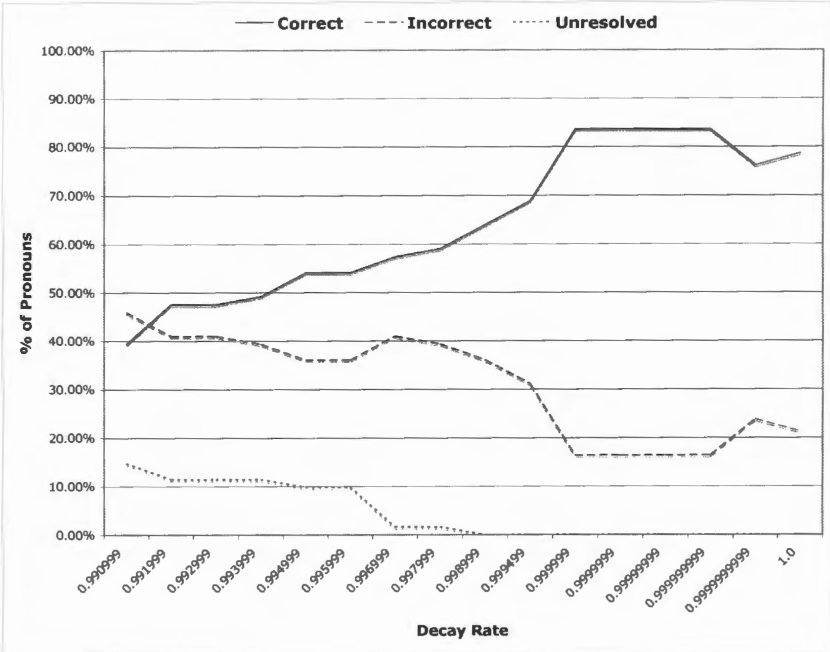


Figure 7.3: The Thompson Model Resolution Results - 3 Brothers

resolved 39.34% of the pronouns correctly. The Thompson model, given the decay rate of 0.995999, was able to resolved 54.10% of the pronouns correctly. Thus, in this comparison, the Thompson model achieved a higher success rate.

When the *Rumpelstiltskin* corpus is considered, the ACT-R model was able to correctly resolve 54.67% of the pronouns, given the decay rate of 0.17. The Thompson model, given the decay rate of 0.995999, was able to correctly resolve 42.67% of the pronouns. So, in this comparison, the ACT-R model achieved a higher success rate than the Thompson model.

When the overall range of results is examined, the Thompson model of activation achieved maximum success rates of 83.61% (The Three Brothers) and 86.67% (Rumpelstiltskin), while the ACT-R model of activation fell short with maximum success rates of 54.10% (The Three Brothers) and 56.00% (Rumpelstiltskin). In general, the Thompson model was able to resolve a higher number of pronouns over a larger range of decay rates. Although the Thompson model achieved a higher overall success rate, a unresolved rate

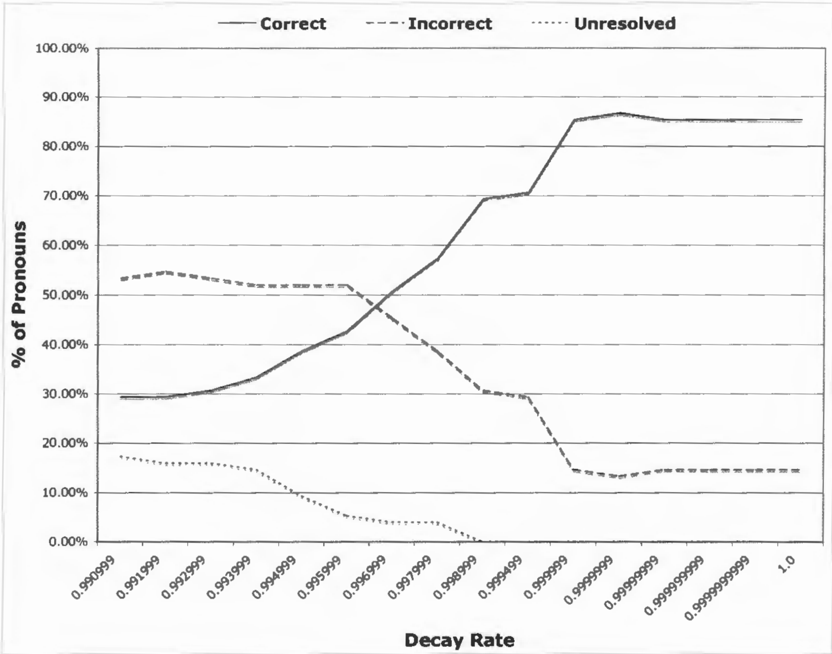


Figure 7.4: The Thompson Model Resolution Results - Rumpelstiltskin

of 0.0% is psychologically implausible based on the findings in [Levine et al 2000].

Examining Figures 7.9 and 7.10, it appears that the ACT-R model of activation had difficulty with concepts begin activated into working memory at the start of each corpus and then problems with getting those concepts out of working memory at the end of the corpus. This difficulty can be attributed to the fact that the ACT-R model of activation is based on the activation history. Concepts appear to move in and out of working memory more fluidly using the Thompson model of activation.

7.9 Classification of Observed Error Types

Throughout the testing of the implemented system, two types, or classes, of errors were observed. The first class of errors involved the incorrect resolution of a pronoun due to lack of information. The following sentence illustrates this error:

The $\boxed{son}_1$ *that builds the best masterpiece will inherit* $\boxed{his}_1$ *house.*

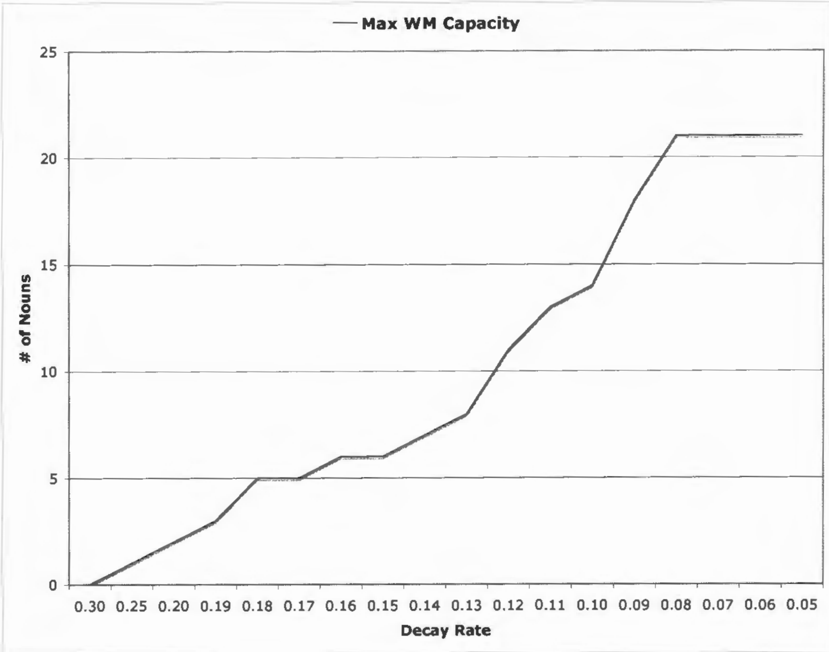


Figure 7.5: The ACT-R Model Working Memory Max Capacity - 3 Brothers

In this example, the system resolved the pronoun *his* with the noun *son*, which was incorrect. The pronoun actually references the noun *father*, which was mentioned earlier in the corpus. This incorrect resolution occurred because the noun *son* was the concept that had the highest antecedent score, and the accompanying noun *house* was not considered by the anaphora resolution algorithm. The second class of errors that was observed involved the incorrect resolution of pronouns that reference events. The next example illustrates this error:

The $\boxed{\textit{father}}_1$ *thought that* $\boxed{\textit{this}}_7$ *was wonderful.*

In this example, the pronoun *this* was incorrectly resolved to reference the noun *horse* from a previous sentence, which was not correct. The pronoun *this* actually references an event from earlier in the corpus. Since resolution of events was not within the scope of this thesis, occurrences of this type of error were not included as results.

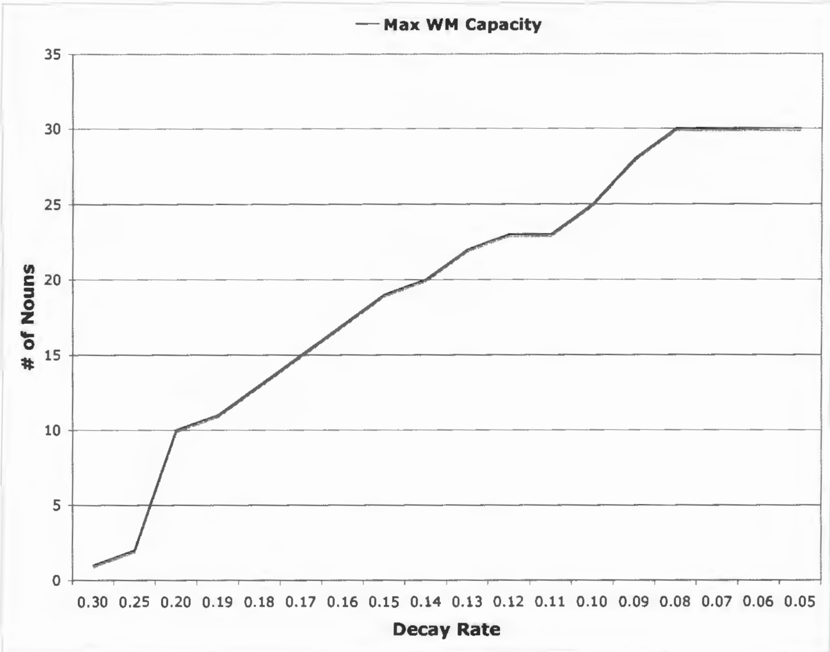


Figure 7.6: The ACT-R Model Working Memory Max Capacity - Rumpelstiltskin

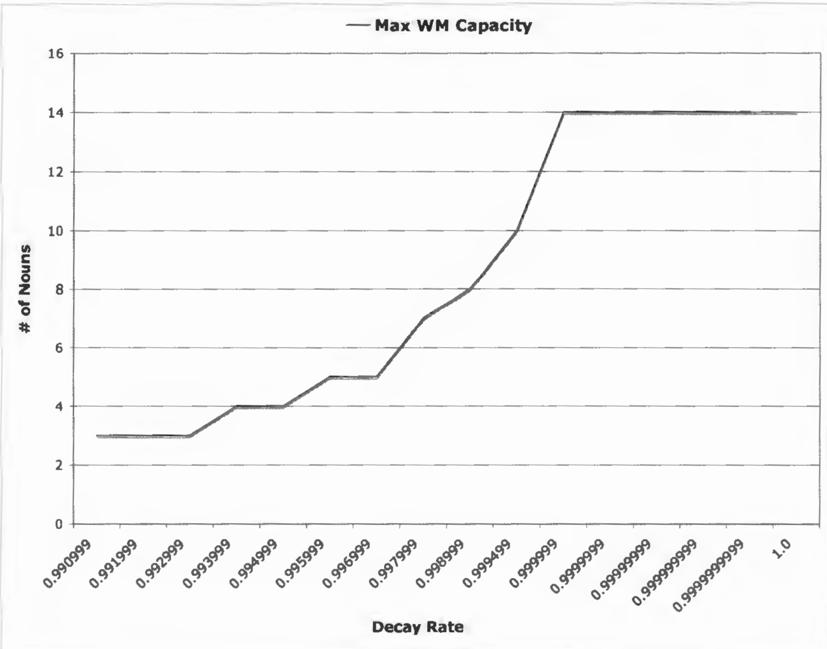


Figure 7.7: The Thompson Model Working Memory Max Capacity - 3 Brothers

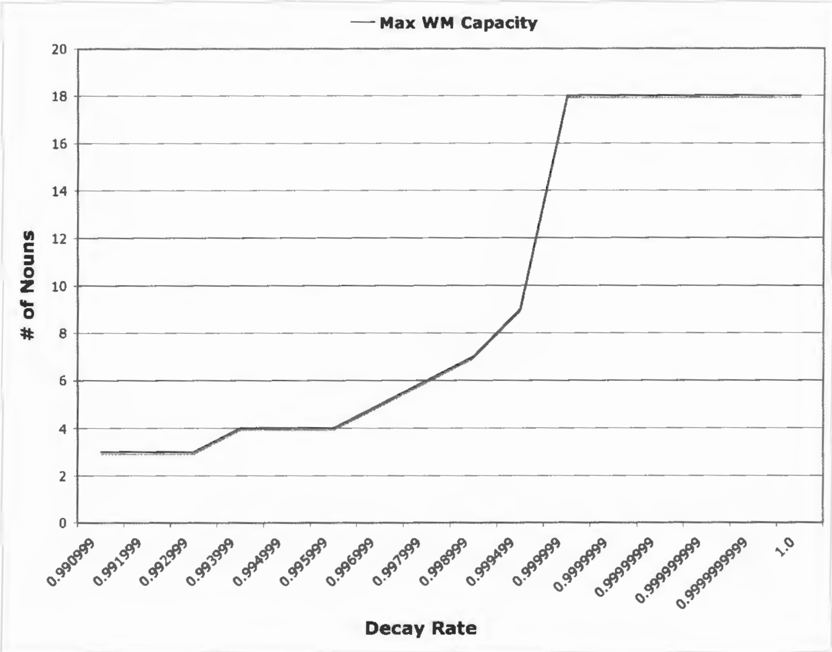


Figure 7.8: The Thompson Model Working Memory Max Capacity - Rumpelstiltskin

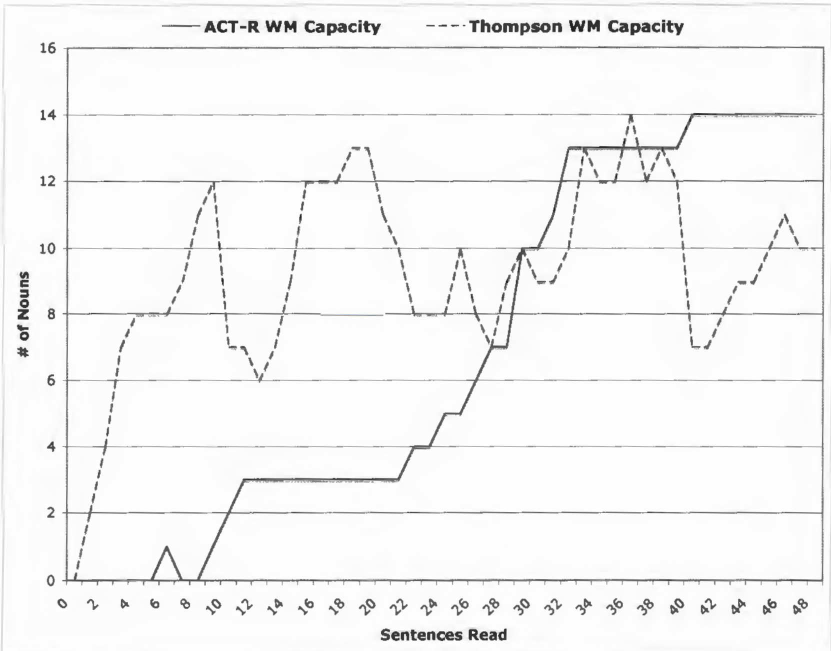


Figure 7.9: Working Memory Contents Comparison - 3 Brothers

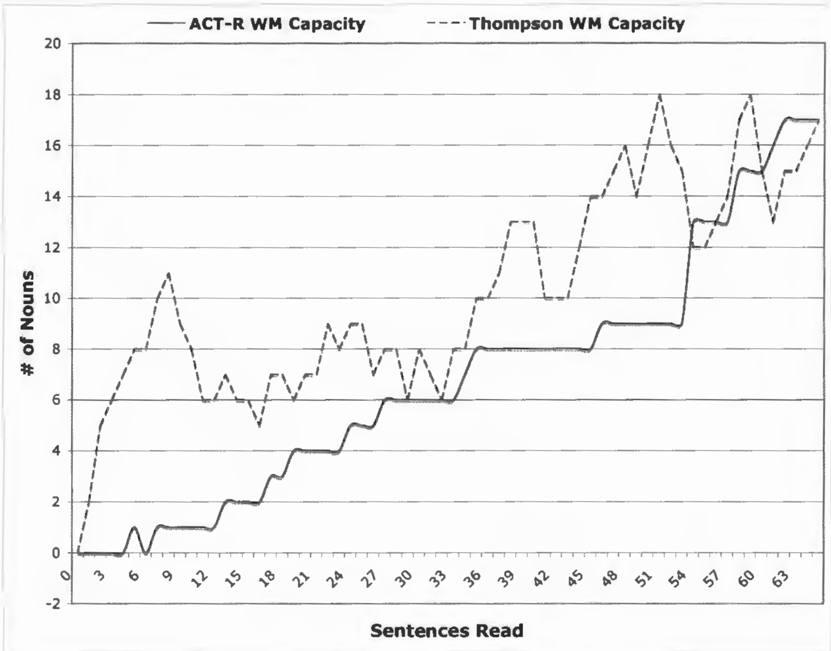


Figure 7.10: Working Memory Contents Comparison - Rumpelstiltskin

CHAPTER EIGHT

Conclusions and Future Work

It will be happened; It shall be going to be happening; It will be was an event that could will have been taken place in the future.

- ARNOLD J. RIMMER *(Future Echos)*

8.1 Comparison with Related Work

Anaphora Resolution

Two of the biggest difficulties with making comparisons between anaphora resolution algorithms are the forthcomingness of authors to publish the results of their algorithms, and obtaining the corpora used for testing their algorithms. Over time, many corpora become difficult to obtain, thus making direct comparisons difficult. Early work such as [Sidner 1979], [Hirst 1981], and [Grosz and Sidner 1986], although popular in the fields of anaphora resolution and discourse analysis, fail to provide comprehensive results for their models. Table 8.1 shows the results obtained in this thesis compared to the work of other authors[1].

Capacity of Working Memory

Figures 7.9 and 7.10 illustrated the results of the dynamic capacity of working memory when an optimal[2] decay rate was used. If a single concept is considered to be a *chunk*, the maximum capacity of the working memory model in this thesis appears to be much higher

[1] Mitkov, Callway, Hobbs, and Ferrández all give results for pronominal references. It is unknown whether these results included pronominal references to events or just nouns.

[2] Optimal decay rate was defined as the value that gave the highest anaphora resolution results.

Algorithm	Accuracy	Corpus
[Mitkov 1998]	89.70%	Minolta Photocopier Manual and StyleWriter User's Guide
[Hobbs 1986]	91.70%	Early Civilization in China, Wheels, and Newsweek
[Callaway and Lester 2002]	97.80%	Little Red Riding Hood
[Ferrández et al 1998]	83.00%	TTU CCITT Handbook
ACT-R	55.05%	The Three Brothers and Rumpelstiltskin
Thompson	85.14%	The Three Brothers and Rumpelstiltskin

Table 8.1: Algorithm Comparison

than the capacity proposed in [Miller 1956], for short-term memory. These differences are possibly an artifact of the differences between what is considered a *chunk* in the human mind and what is considered a *chunk* in the model presented in this thesis.

8.2 Future Work

Inclusion of Additional Semantic Structures

Linking multiple semantic structures in the manner described in Chapter 4 could potentially increase the level of accuracy of anaphora resolution by providing additional contextual information. An excellent example is from *The Three Brothers*, where both the ACT-R model of activation and the Thompson model failed to resolve the antecedent for the sentence *The son that builds the best masterpiece will inherit his house.* Both models resolved the possessive pronoun *his* with the noun *son*, since *son* had the highest activation within working memory and highest feature set scoring. Unfortunately, this resolution is incorrect. The pronoun *his* should resolve to the noun *father* described in an earlier sentence. If an anaphora algorithm had information relating to the father-son relationship available, and information relating to the fact that the father owned a house, the algorithm could use this information to give a higher score to *father*, and resolve the antecedent correctly. Semantic representations, such as conceptual graphs, are effective in modelling this type of information. This modularization of modelling the behaviour of the human mind is supported by psychological literature.

Noun Instances in Working Memory and Long-Term Memory

The models of long-term memory and working memory presented in this thesis only interact with single, generalized instances of nouns. For example, within the model of long-term memory, there exists only a single occurrence of the noun *man*. The existence of only one occurrence of a noun is problematic when defining the correctness of an anaphora resolution algorithm when a corpus contains multiple *men*. A possible solution to this problem is the introduction of an additional memory model that is a hybrid of working memory and long-term memory. Event memory, which was introduced in Chapter 3, is a model that could be adapted to handle nouns in addition to handling events. Event memory could contain instances of concepts that are generated as a corpus is read. The instance of a concept would gradually decay until only the most general concept exists. Long-term memory would act as a repository for generalized concepts, analogous to generalize event memory. The hierarchy of generalized concepts and instances of concepts could be realized by using the conceptual hierarchies introduced in Chapter 2.

Chart Parsing

Chart parsing is a technique for bottom up parsing that avoids parsing the same structure more than once. Parsed sub-phrases are stored in a database called a *chart*, which is consulted when any type of backtracking occurs. A chart parser can be used to increase the speed of parsing while also creating a parser that is more tolerant to ungrammatical sentences [Allen 1995], [Russel and Norvig 1995], [Brown 2000], and [Thompson 2001a]. Potentially, a complete parse of a sentence would not be required for anaphora resolution. Parsing could be limited to just the *noun phrase* and *verb phrase* levels, and anaphora resolution could proceed from there.

8.3 Conclusion

This thesis demonstrated the viability of using a modular, two-level memory model to perform anaphora resolution. Two models of human memory were used in conjunction with an anaphora resolution algorithm to solve the problem of pronominal references. Two models of concept activation and decay were implemented and subsequently tested on corpora of text with varying decay rates. The two-level memory model and anaphora resolution algorithm achieved resolution accuracy rates of up to 54.10%(ACT-R) and 83.61%(Thompson) for a modified version of *The Three Brothers* corpus, and 56.00%(ACT-R) and 86.67%(Thompson) for a modified version of the *Rumpelstiltskin* corpus. Although the results fall a bit short of the results from other works(with the exception of [Ferrández et al 1998]), these results are only a baseline for additional work. The model is intended to be an expansive model of human memory. It is theorized that adding additional semantic representations, and anaphora resolution algorithms, would increase the accuracy of the two-level memory model.

Bibliography

[Allen 1995] James Allen. 1995. *Natural Language Understanding.* The Benjamin/Cummings Publishing Company, Redwood City, CA, 2nd edition. vii, 5, 11, 61

[Alshawi et al 1989] H. Alshawi, D.M. Carter, J. van Eijck, R.C. Moore, D.B. Moran, F.C.N. Pereira, S.G. Pulman, and A.G. Smith. 1989. Research Programme in Natural Language Processing. Technical report, SRI International, Cambridge, UK. vi, 13, 15, 45

[Altmann 1999] Gerry T.M. Altmann. 1999. Thematic Roles Assignment in Context. *Journal of Memory and Language*, 41(1):124–145, July. vi, 12, 13

[Anderson and Matessa 1997] John R. Anderson and Michael Matessa. 1997. A Production System Theory Of Serial Memory. *Psychological Review*, 104:728–748. 21, 22

[Anderson et al 1998] John R. Anderson, Dan Bothell, Christian Lebiere, and Michael Matessa. 1998. An Integrated Theory of List Memory. *Journal of Memory and Language*, 38:341–380, May. 22

[Anderson et al 2001] John R. Anderson, Raluca Budiu, and Lynne M. Reder. 2001. A Theory of Sentence Memory as Part of A General Theory of Memory. *Journal of Memory and Language*, 45:337–367, October. 20, 32

[Anderson 1976] John R. Anderson. 1976. *Language, Memory, And Thought.* Erlbaum, Hillsdale, NJ. 20

[Anderson 1983] John R. Anderson. 1983. *The Architecture Of Cognition.* Harvard University Press, Cambridge, MA. 20

[Baddeley and Hitch 1974] Alan D. Baddeley and Graham Hitch, 1974. *The psychology of learning and motivation: Advances in research and theory*, volume 8, chapter Working Memory, pages 47–89. Academic Press, New York, NY. 18

[Baddeley 1986] Alan D. Baddeley. 1986. *Working Memory.* Oxford Psychology Series. Oxford University Press, Oxford. 19, 41

[Baddeley 1990] Alan D. Baddeley. 1990. *Human Memory : Theory And Practice.* Allyn Bacon, Boston. 19, 41

[Baddeley 2000] Alan Baddeley. 2000. The episodic buffer: a new component of working memory? *Trends in Cognitive Sciences*, 4(11):417–423, November. 18

[Belew 1987] R. K. Belew. 1987. A connectionist approach to conceptual information retrieval. In *ICAIL '87: Proceedings of the 1st international conference on Artificial intelligence and law*, pages 116–126, New York, NY, USA. ACM Press. 6

[Bennet 1995] Paul Bennet. 1995. *A Course in Generalized Phrase Structure Grammar.* Studies in Computational Linguistics. UCL Press Limited, London, England. 45

[Berger et al 2004] Helmut Berger, Michael Dittenbach, and Dieter Merkl. 2004. An adaptive information retrieval system based on associative networks. In *CRPIT '04: Proceedings of the first Asian-Pacific conference on Conceptual modelling*, pages 27–36, Darlinghurst, Australia, Australia. Australian Computer Society, Inc. 6

[Brown et al 2001] Charles Grant Brown, Lynda Williams, David Kubert, Clifford James Thompson, Neil Hagen, Dan Wolczuk, Cameron Rowe, Kurt Bolko, Jason Hess, Doug Smith, David Fidler, Mike Sluszkiewicz, John Everman, Richard Little, Ho Shum, and Alex Yung. 2001. The Language Engine Project 1996-2001. University of Northern British Columbia, Prince George, BC. 45

[Brown 2000] Charles Grant Brown. 2000. Natural Language Processing Manuscripts. University of Northern British Columbia, Prince George, BC. 61

[Brown 2003] Charles Grant Brown. 2003. Lecture Notes in Computational Linguistics. University of Northern British Columbia, Prince George, BC. 34, 35, 46

[Bull 1960] W. Bull. 1960. *Time, Tense, and the Verb: A Study in Theoretical and Applied Linguistics, with Particular Attention to Spanish.* University of California Press, Berkeley, CA. 10

[Callaway and Lester 2002] Charles B. Callaway and James C. Lester. 2002. Pronominalization in General Discourse and Dialogue. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 88–95. Association for Computational Linguistics, July. 60

[Carter 1985] David Maclean Carter. 1985. A Shallow Approach To Anaphora Resolution. PhD Thesis 88, University Of Cambridge, Camdridge, England. 34, 35, 46

[Carter 1990] David M. Carter. 1990. Control Issues In Anaphora Resolution. *Journel of Semantics*, 7:435–454. 34, 35, 46

[Chung and Moldovan 1993] Sang-Hwa Chung and Dan I. Moldovan. 1993. Modeling Semantic Networks On The Connection Machine. *Journal Of Parallel And Distributed Computing*, 17:1993. 6

[Collins and Quillian 1969] A. Collins and M.R. Quillian. 1969. Retrieval Time from Semantic Memory. *Journal of Verbal Leanring & Verbal Behavior*, 8:240–247. 6

[Culce-Murcia and Larsen-Freeman 1999] Marianne Culce-Murcia and Diane Larsen-Freeman. 1999. *The Grammar Book: An ESL/EFL Teacher's Course.* Heinle & Heinle, Boston, MA, 2nd edition. 10

[Federmeier and Kutas 1999] Kara D. Federmeier and Marta Kutas. 1999. A Rose by Any Other Name: Long-Term Memory Structure and Sentence Processing. *Journal of Memory and Language*, 41:469–495. 19

[Ferrández et al 1998] A. Ferrández, M. Palomar, and L. Moreno. 1998. Anaphor resolution in unrestricted texts with partial parsing. In *Proceedings of the 36th annual meeting on Association for Computational Linguistics*, pages 385–391, Morristown, NJ, USA. Association for Computational Linguistics. 60, 62

[Foder 1983] Jerry A. Foder. 1983. *Modularity of Mind: An Essay on Faculty Psychology.* MIT Press, Cambridge, MA. 32

[Gathercole and Baddeley 1993] Susan E. Gathercole and Alan D. Baddeley. 1993. *Working Memory and Language.* Essays in Cognitive Psychology. Lawrence Erlbaum Associates, Hove, United Kingdom. 18, 41

[Geurts 1999] Paul Geurts. 1999. *Presuppositions and Pronouns.* Current Research in the Semantics/Pragmatics Interface. Elsevier, Amsterdam. 33

[Ginzburg and Cooper 2001] Jonathan Ginzburg and Robin Cooper. 2001. Resolving Ellipsis In Clarification. In *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pages 236–243. 33

[Grosz and Sidner 1986] Barbara J. Grosz and Candice L. Sidner. 1986. Attention, Intentions, and the Structure of Discourse. *Computational Linguistics*, 12(3):175–204. 34, 59

[Grosz 1977] Barbara J. Grosz. 1977. The Representation And Use Of Focus In Dialogue Understanding. Technical report, SRI International. 33, 34

[Group 2004] ACT-R Research Group. 2004. ACT-R Tutorial: Unit 1. http://act-r.psy.cmu.edu/tutorials, January. 21

[Haarmann et al 2003] Henk J. Haarmann, Edddy J. Davelaar, and Marius Usher. 2003. Individual Differences In Semantic Short-Term Memory Capacity and Reading Comprehension. *Journal of Memory and Language*, 48:320–345. 19

[Hardt 1997] Daniel Hardt. 1997. An Empirical Approach to VP Ellipsis. *Computational Linguistics*, 23(4):525–541, December. 33

[Hebb 1949] D.O. Hebb. 1949. *The Organization of Behavior.* Wiley, New York. 23, 27, 30

[Hirst 1981] Graeme Hirst. 1981. Discourse-oriented anaphora resolution in natural language understanding: a review. *Computational Linguistics*, 7(2):85–98. 59

[Hobbs 1986] Jerry R. Hobbs, 1986. *Readings In Natural Language Processing*, chapter Resolving Pronoun References, pages 339–352. Morgan Kaufmann Publishers, Los Altos, CA. 34, 35, 60

[Hunt 1973] Eart Hunt, 1973. *Computer Models Of Thought And Language*, chapter The Memory We Must Have, pages 343–371. Freeman and Company, San Francisco, CA. 22, 23

[Kay 1971] P. Kay. 1971. Taxonomy and semantic contrast. *Language*, 47:866–887. 9

[Kazuhiro et al 1992] Kimura Kazuhiro, Suzuoka Takashi, and Amano Sin-ya. 1992. Association-based natural language processing with neural networks. In *Proceedings of the 30th annual meeting on Association for Computational Linguistics*, pages 224–231, Morristown, NJ, USA. Association for Computational Linguistics. 6

[Koh and Clifton 2002] Sungryong Koh and Charles Clifton. 2002. Resolution of the Antecedent of a Plural Pronoun: Ontological Categories and Predicate Symmetry. *Journal of Memory and Language*, 46(4):830–844, May. 46

[Levine et al 2000] Willian H. Levine, Alexandria E. Guzmán, and Celia M. Klin. 2000. When Anaphor Resolution Fails. *Journal of Memory and Language*, 43(4):594–617, November. 37, 46, 54

[Logie 1996] Robert H. Logie, 1996. *The Seven Ages of Working Memory*, pages 31–65. Counterpoints: Cognition, Memory, and Language. Oxford University Press, New York, NY. 17

[Luger and Stubblefield 1998] George F Luger and Wlliam A Stubblefield, 1998. *Artificial Intelligence*, chapter 14, pages 690–700. Addison Wesley Longman Inc, Reading, MA, 3rd edition. 27

[Ma and Isahara 2000] Qi Ma and Hitoshi Isahara. 2000. Semantic Networks Represented by Adaptive Associative Memories. *Nerocomputing*, 34:207–225, September. 6, 9

[McCulloch and Pitts 1943] W.S. McCulloch and W. Pitts. 1943. A logical calculas of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133. 23

[Miller 1956] G.A. Miller. 1956. The Magical Number Seven, Plus or Minus Two: Some Limits On Our Capacity For Processing Information. *Psychological Review*, 63:81–97. 18, 21, 60

[Mitkov 1998] Ruslan Mitkov. 1998. Robust pronoun resolution with limited knowledge. In *Proceedings of the 36th annual meeting on Association for Computational Linguistics*, pages 869–875, Morristown, NJ, USA. Association for Computational Linguistics. 34, 37, 60

[Nash-Webber and Reiter 1977] Bonnie Nash-Webber and Raymond Reiter. 1977. Anaphora And Logical Form: On Formal Meaning Representations For Natural Language. In *Proceedings of 5th International Joint Conference on Artificial Intelligence*, volume 1, pages 121–131. 33

[Ockerbloom 2006] John Mark Ockerbloom. 2006. Grimm's Fairy Tales. `http://www.cs.cmu.edu/~spok/grimmtmp`. 49

[Partee 1984] Barbara Partee. 1984. Nominal and Temporal Anaphora. *Linguistics and Philosophy*, 7:243–286. 33, 34

[Piwek and Krahmer 2000] Paul Piwek and Emiel Krahmer. 2000. Presuppositions in Context: Constructing Bridges. In Emiel Krahmer and Paul Piwek, editors, *Varieties Of Anaphora*, pages 87–96, Birmingham England. ESSLLI. 33

[Quillian 1968] M.A. Quillian, 1968. *Semantic Information Processing*, chapter Semantic Memory, pages 216–270. MIT Press, Cambridge, MA. 6

[Reichenbach 1947] H. Reichenbach. 1947. *Elements of Symbolic Logic*. Macmillian, New York. 11

[Richardson 1996] John T.E. Richardson, 1996. *Evolving Concepts of Working Memory*, chapter 1, pages 3–30. Counterpoints: Cognition, Memory, and Language. Oxford University Press, New York, NY. 17

[Rosch et al 1976] E. Rosch, C.B. Mervis, W.D. Gray, D.M. Johnson, and P. Boyes-Braem. 1976. Basic objects in natural categories. *Cognitive Psychology*, 8:382–439. 9

[Russel and Norvig 1995] Stuart J. Russel and Peter Norvig. 1995. *Artificial Intelligence: A Modern Approach*. Prentice Hall Series In Artificial Intelligence. Prentice Hall, Upper Saddle River, NJ. 24, 61

[Schank 1986] Roger C. Schank, 1986. *Readings In Natural Language Processing*, chapter Language And Memory, pages 171–191. Morgan Kaufmann Publishers Inc., Los Altos, CA. 22

[Sidner 1979] Candice L. Sidner. 1979. Toward A Computational Theory of Definite Anaphora Comprehension in English. PhD Thesis 537, MIT Artificial Intelligence Laboratory, Cambridge, MA. 34, 36, 59

[Sidner 1983] Candice L. Sidner, 1983. *Focusing in the Comprehension of Definite Anaphora*, chapter 5, pages 267–330. M.I.T. Press, Cambridge, MA. 3, 34

[Sowa 1976] John F. Sowa. 1976. Conceptual Graphs for a Data Base Interface. *IBM Journal of Research and Development*, 20:336–357. 7

[Sowa 1979] John F. Sowa. 1979. Semantics of conceptual graphs. In *Proceedings of the 17th annual meeting on Association for Computational Linguistics*, pages 39–44, Morristown, NJ, USA. Association for Computational Linguistics. 7

[Sowa 1984] John F. Sowa. 1984. *Conceptual Structures: Information Processing in Mind and Machine.* Addison-Wesley, Reading, MA. 6, 7

[Sowa 2000] John F. Sowa. 2000. *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* Brooks/Cole, Pacific Grove. 8, 12

[Sterling and Shapiro 1999] Leon Sterling and Ehud Shapiro. 1999. *The Art of Prolog.* MIT Press, 3rd edition. 44

[Thompson and Hagen 2001] Clifford James Thompson and Neil Hagen. 2001. CPSC 496 Relative Clauses, Quantifiers and Determiners. University of Northern British Columbia, Prince George, BC.

[Thompson 2001a] Clifford James Thompson. 2001a. CPSC 674 Bottom-Up Parsing: Chart Parsing. University of Northern British Columbia, Prince George, BC. 61

[Thompson 2001b] Clifford James Thompson. 2001b. CPSC 674 Grammars & Syntactic Analysis. University of Northern British Columbia, Prince George, BC.

APPENDIX A

Corpora

The Three Brothers[3]

There was a man who had three sons. He had nothing in the world. Each son wanted the house after his death. The Father loved them. He did not know what he should do. He did not wish to sell the house. It had belonged to his forefathers. He conceived a plan and told his sons that they must learn a trade. The son that builds the best masterpiece will inherit his house.

The sons were content with this. The first son was determined to be a blacksmith. The second son wanted to be a barber. The third son desired to be a fencing master. They set a time when they should come home.

The brothers found skillful masters who taught them their trades. The blacksmith had to shoe horses that belonged to the king. He believed that he would inherit the house. The barber shaved only distinguished people. He believed that his father would give the house to him. The fencing master suffered many blows to his body but he grit his teeth. He thought that he would win the house.

The brothers returned home to their father. They did not know when they would demonstrate their skills to their father. The brothers sat and contemplated what they could do. A hare ran across the field. The barber took his basin and soap. He lathered until the hare drew near. He soaped and saved the hare's whiskers while he was running at his top speed. He did not cut his skin or a hair on his body. The father was delighted.

A nobleman can in his coach and at full speed. The blacksmith ran towards the coach. He took four horseshoes off the horse while it was galloping and put new shoes on him. The father thought that this was wonderful.

[3] Adapted from `http://www.cs.cmu.edu/~spok/grimmtmp/094.txt`

The third son asked to demonstrate his skills. It began to rain an the son drew sword. The sword flourished backwards and forwards above his head. No raindrops fell upon him. The rain fell harder and harder. He flourished his sword and remained dry. His father was amazed at this and gave his house to the third son.

His brothers were satisfied with this. They decided to line together since they loved eachother. The brothers continued their trades and earned a good living. They lived happily until they grew old. One brother became sick and died. The brothers grieved intensely and they became ill and died. They were laid in the same grave because they loved eachother.

Rumpelstiltskin[4]

There was a miller who was poor but had a beautiful daughter. The miller visited the king and told him that his daughter could spin straw into gold. The king said that this was an art and pleased him. He requested that the miller bring his daughter to the palace.

The girl was brought to the king. He took her into a room that was filled with straw. She was given a spinning wheel and a reel. The king demanded that she complete the work by tomorrow or die. He locked the room and left the daughter. The poor daughter sat there and wept. She knew that she could not spin straw into gold.

The door opened and a little man entered the room. He asked the girl why she was crying. She told him that she must spin straw into gold. The little man told her that he could spin the straw for a price. He asked the girl what she could give to him. The daughter offered a necklace to him. The man took the necklace and sat at the spinning wheel. He spun the straw into gold.

The king returned in the morning and saw the gold. He was astonished and delighted. His heart filled with greed. The daughter was taken to a larger room that was filled with straw. He demanded that she complete the work by tomorrow.

The girl sat in the large room and cried. The little man returned and told her that he could spin the straw into gold. He asked her what she could give for the task. The daughter gave a ring to the small man. He grabbed the ring and spun the straw into gold.

The king returned and was amazed by the feat. He demanded that she spin more gold. The daughter was placed in a larger room. The king asked that she complete the task by the morning. He thought that she would be his wife when the task was completed. The manikin returned when the girl was alone. He asked what she would give for the task. She answered that she had nothing. The girl promised to give her first child when she becomes queen. She did not think that this would happen. The little man spun the straw into gold.

[4]Adapted from `http://www.cs.cmu.edu/~spok/grimmtmp/044.txt`

The king returned in the morning. He found what he wished. The king took the girl in marriage and she became queen. She brought a beautiful child into the world. The queen gave no thought to the manikin. He entered her room and asked for her child. She was surprised and offered riches to him. The manikin refused the offer. The queen began to cry and the little man felt pity. He said that she could keep her child but she must guess his name in three days. She sent a messenger across the country. He searched for any name that might exist.

The manikin returned the next day. The queen guessed Casper and Melchior and Balthazar and other names that she knew. He said that she was incorrect. She sent a messenger on the second day. The queen asked for uncommon names. She guessed Shortribs and Sheepshanks and Laceleg but was incorrect.

The messenger found the manikin's house and overheard his name. The queen was delighted. The manikin returned on the final day and ask for a name. She guessed Conrad and Harry. He said that she was incorrect. She guessed Rumpelstiltskin. The manikin became angry and was pulled into the earth.

APPENDIX B

Tagged Corpora

The Three Brothers

There was a $\boxed{\text{man}}_{1}$ who had three $\boxed{\text{sons}}_{2}$. $\boxed{\text{He}}_{1}$ had nothing in the world. Each son wanted the house after $\boxed{\text{his}}_{1}$ death. The $\boxed{\text{father}}_{1}$ loved $\boxed{\text{them}}_{2}$. $\boxed{\text{He}}_{1}$ did not know what $\boxed{\text{he}}_{1}$ should do. $\boxed{\text{He}}_{1}$ did not wish to sell the $\boxed{\text{house}}_{3}$. $\boxed{\text{It}}_{3}$ had belonged to $\boxed{\text{his}}_{1}$ forefathers. $\boxed{\text{He}}_{1}$ conceived a plan and told $\boxed{\text{his}}_{1}$ $\boxed{\text{sons}}_{2}$ that $\boxed{\text{they}}_{2}$ must learn a trade. The $\boxed{\text{son}}_{2abc}$ that builds the best masterpiece will inherit $\boxed{\text{his}}_{1}$ house.

The $\boxed{\text{sons}}_{2}$ were content with this. The first $\boxed{\text{son}}_{2a}$ was determined to be a $\boxed{\text{blacksmith}}_{2a}$. The second $\boxed{\text{son}}_{2b}$ wanted to be a $\boxed{\text{barber}}_{2b}$. The third $\boxed{\text{son}}_{2c}$ desired to be a $\boxed{\text{fencing master}}_{2c}$. $\boxed{\text{They}}_{2}$ set a time when $\boxed{\text{they}}_{2}$ should come home.

The $\boxed{\text{brothers}}_{2}$ found skillful $\boxed{\text{masters}}_{4}$ who taught $\boxed{\text{them}}_{2}$ $\boxed{\text{their}}_{4}$ trades. The $\boxed{\text{blacksmith}}_{2a}$ had to shoe horses that belonged to the king. $\boxed{\text{He}}_{2a}$ believed that $\boxed{\text{he}}_{2a}$ would inherit the house. The $\boxed{\text{barber}}_{2b}$ shaved only distinguished people. $\boxed{\text{He}}_{2b}$ believed that $\boxed{\text{his}}_{2b}$ father would give the house to $\boxed{\text{him}}_{2b}$. The $\boxed{\text{fencing master}}_{2c}$ suffered many blows to $\boxed{\text{his}}_{2c}$ body but $\boxed{\text{he}}_{2c}$ grit $\boxed{\text{his}}_{2c}$ teeth. $\boxed{\text{He}}_{2c}$ thought that $\boxed{\text{he}}_{2c}$ would win the house.

The $\boxed{\text{brothers}}_{2}$ returned home to $\boxed{\text{their}}_{2}$ father. $\boxed{\text{They}}_{2}$ did not know when $\boxed{\text{they}}_{2}$ would demonstrate $\boxed{\text{their}}_{2}$ skills to $\boxed{\text{their}}_{2}$ $\boxed{\text{father}}_{1}$. The $\boxed{\text{brothers}}_{2}$ sat and contemplated what $\boxed{\text{they}}_{2}$ could do. A $\boxed{\text{hare}}_{5}$ ran across the field. The $\boxed{\text{barber}}_{2b}$ took $\boxed{\text{his}}_{2b}$ basin and soap. $\boxed{\text{He}}_{2b}$ lathered until the $\boxed{\text{hare}}_{5}$ drew near. $\boxed{\text{He}}_{2b}$ soaped and saved the $\boxed{\text{hare's}}_{5}$ whiskers while $\boxed{\text{he}}_{2b}$ was running at $\boxed{\text{his}}_{2b}$ top speed. $\boxed{\text{He}}_{2b}$ did not cut $\boxed{\text{his}}_{5}$ skin or a hair on $\boxed{\text{his}}_{5}$ body. The $\boxed{\text{father}}_{1}$ was delighted.

A $\boxed{\text{nobleman}}_{6}$ can in $\boxed{\text{his}}_{6}$ coach and at full speed. The $\boxed{\text{blacksmith}}_{2a}$ ran towards the coach. $\boxed{\text{He}}_{2a}$ took four horseshoes off the $\boxed{\text{horse}}_{7}$ while $\boxed{\text{it}}_{7}$ was galloping and put new shoes on $\boxed{\text{him}}_{7}$. The $\boxed{\text{father}}_{1}$ thought that this was wonderful.

The $\boxed{\text{third son}}_{2c}$ asked to demonstrate $\boxed{\text{his}}_{2c}$ skills. It began to rain and the $\boxed{\text{son}}_{2c}$ drew $\boxed{\text{his}}_{2c}$ sword. The sword flourished backwards and forwards above $\boxed{\text{his}}_{2c}$ head. No raindrops fell upon $\boxed{\text{him}}_{2c}$. The rain fell harder and harder. $\boxed{\text{He}}_{2c}$ flourished $\boxed{\text{his}}_{2c}$ sword and remained dry. $\boxed{\text{His}}_{2c}$ $\boxed{\text{father}}_{1}$ was amazed at this and gave $\boxed{\text{his}}_{1}$ $\boxed{\text{house}}_{3}$ to the $\boxed{\text{third son}}_{2c}$.

$\boxed{\text{His}}_{2c}$ $\boxed{\text{brothers}}_{2}$ were satisfied with this. $\boxed{\text{They}}_{2}$ decided to line together since $\boxed{\text{they}}_{2}$ loved eachother. The $\boxed{\text{brothers}}_{2}$ continued $\boxed{\text{their}}_{2}$ trades and earned a good living. $\boxed{\text{They}}_{2}$ lived happily until $\boxed{\text{they}}_{2}$ grew old. One brother became sick and died. The $\boxed{\text{brothers}}_{2}$ grieved intensely and $\boxed{\text{they}}_{2}$ became ill and died. $\boxed{\text{They}}_{2}$ were laid in the same grave because $\boxed{\text{they}}_{2}$ loved eachother.

Rumpelstiltskin

There was a $\boxed{\text{miller}}_1$ who was poor but had a beautiful $\boxed{\text{daughter}}_2$. The $\boxed{\text{miller}}_1$ visited the $\boxed{\text{king}}_3$ and told $\boxed{\text{him}}_3$ that $\boxed{\text{his}}_1$ $\boxed{\text{daughter}}_2$ could spin straw into gold. The $\boxed{\text{king}}_3$ said that this was an art and pleased $\boxed{\text{him}}_3$. $\boxed{\text{He}}_3$ requested that the $\boxed{\text{miller}}_1$ bring $\boxed{\text{his}}_1$ $\boxed{\text{daughter}}_2$ to the palace.

The $\boxed{\text{girl}}_2$ was brought to the $\boxed{\text{king}}_3$. $\boxed{\text{He}}_3$ took $\boxed{\text{her}}_2$ into a room that was filled with straw. $\boxed{\text{She}}_2$ was given a spinning wheel and a reel. The $\boxed{\text{king}}_3$ demanded that $\boxed{\text{she}}_2$ complete the work by tomorrow or die. $\boxed{\text{He}}_3$ locked the $\boxed{\text{room}}_6$ and left the $\boxed{\text{daughter}}_2$. The poor $\boxed{\text{daughter}}_2$ sat $\boxed{\text{there}}_6$ and wept. $\boxed{\text{She}}_2$ knew that $\boxed{\text{she}}_2$ could not spin straw into gold.

The door opened and a little $\boxed{\text{man}}_4$ entered the room. $\boxed{\text{He}}_4$ asked the $\boxed{\text{girl}}_2$ why $\boxed{\text{she}}_2$ was crying. $\boxed{\text{She}}_2$ told $\boxed{\text{him}}_4$ that $\boxed{\text{she}}_2$ must spin straw into gold. The little $\boxed{\text{man}}_4$ told $\boxed{\text{her}}_2$ that $\boxed{\text{he}}_4$ could spin the straw for a price. $\boxed{\text{He}}_4$ asked the $\boxed{\text{girl}}_2$ what $\boxed{\text{she}}_2$ could give to $\boxed{\text{him}}_4$. The $\boxed{\text{daughter}}_2$ offered a necklace to $\boxed{\text{him}}_4$. The $\boxed{\text{man}}_4$ took the necklace and sat at the spinning wheel. $\boxed{\text{He}}_4$ spun the straw into gold.

The $\boxed{\text{king}}_3$ returned in the morning and saw the gold. $\boxed{\text{He}}_3$ was astonished and delighted. $\boxed{\text{His}}_3$ heart filled with greed. The $\boxed{\text{daughter}}_2$ was taken to a larger room that was filled with straw. $\boxed{\text{He}}_3$ demanded that $\boxed{\text{she}}_2$ complete the work by tomorrow.

The $\boxed{\text{girl}}_2$ sat in the large room and cried. The little $\boxed{\text{man}}_4$ returned and told $\boxed{\text{her}}_2$ that $\boxed{\text{he}}_4$ could spin the straw into gold. $\boxed{\text{He}}_4$ asked $\boxed{\text{her}}_2$ what $\boxed{\text{she}}_2$ could give for the task. The $\boxed{\text{daughter}}_2$ gave a ring to the small $\boxed{\text{man}}_4$. $\boxed{\text{He}}_4$ grabbed the ring and spun the straw into gold.

The $\boxed{\text{king}}_3$ returned and was amazed by the feat. $\boxed{\text{He}}_3$ demanded that $\boxed{\text{she}}_2$ spin more gold. The $\boxed{\text{daughter}}_2$ was placed in the largest room. The $\boxed{\text{king}}_3$ asked that $\boxed{\text{she}}_2$ complete the task by the morning. $\boxed{\text{He}}_3$ thought that $\boxed{\text{she}}_2$ would be $\boxed{\text{his}}_3$ $\boxed{\text{wife}}_2$ when the task was completed. The $\boxed{\text{manikin}}_4$ returned when the $\boxed{\text{girl}}_2$ was alone. $\boxed{\text{He}}_4$ asked what $\boxed{\text{she}}_2$ would give for the task. $\boxed{\text{She}}_2$ answered that $\boxed{\text{she}}_2$ had nothing. The $\boxed{\text{girl}}_2$ promised to give $\boxed{\text{her}}_2$ first child when $\boxed{\text{she}}_2$ becomes $\boxed{\text{queen}}_2$. $\boxed{\text{She}}_2$ did not think that this would happen. The little $\boxed{\text{man}}_4$ spun the straw into gold.

The $\boxed{\text{king}}_3$ returned in the morning. $\boxed{\text{He}}_3$ found what $\boxed{\text{he}}_3$ wished. The $\boxed{\text{king}}_3$ took the $\boxed{\text{girl}}_2$ in marriage and $\boxed{\text{she}}_2$ became $\boxed{\text{queen}}_2$. $\boxed{\text{She}}_2$ brought a beautiful child into the world. The $\boxed{\text{queen}}_2$ gave no thought to the $\boxed{\text{manikin}}_4$. $\boxed{\text{He}}_4$ entered $\boxed{\text{her}}_2$ room and asked for $\boxed{\text{her}}_2$ child. $\boxed{\text{She}}_2$ was surprised and offered riches to $\boxed{\text{him}}_4$. The $\boxed{\text{manikin}}_4$ refused the offer. The $\boxed{\text{queen}}_2$ began to cry and the little $\boxed{\text{man}}_4$ felt pity. $\boxed{\text{He}}_4$ said that $\boxed{\text{she}}_2$ could keep $\boxed{\text{her}}_2$ child but $\boxed{\text{she}}_2$ must guess $\boxed{\text{his}}_4$ name in three days. $\boxed{\text{She}}_2$ sent a $\boxed{\text{messenger}}_5$ across the country. $\boxed{\text{He}}_5$ searched for any name that might exist.

The $\boxed{\text{manikin}}_4$ returned the next day. The $\boxed{\text{queen}}_2$ guessed Casper and Melchior and Balthazar and other names that $\boxed{\text{she}}_2$ knew. $\boxed{\text{He}}_4$ said that $\boxed{\text{she}}_2$ was incorrect. $\boxed{\text{She}}_2$ sent a $\boxed{\text{messenger}}_5$ on the second day. The $\boxed{\text{queen}}_2$ asked for uncommon names. $\boxed{\text{She}}_2$ guessed Shortribs and Sheepshanks and Laceleg but was incorrect.

The $\boxed{\text{messenger}}_5$ found the $\boxed{\text{manikin's}}_4$ house and overheard $\boxed{\text{his}}_4$ name. The $\boxed{\text{queen}}_2$ was delighted. The $\boxed{\text{manikin}}_4$ returned on the final day and ask for a name. $\boxed{\text{She}}_2$ guessed Conrad and Harry. $\boxed{\text{He}}_4$ said that $\boxed{\text{she}}_2$ was incorrect. $\boxed{\text{She}}_2$ guessed Rumpelstiltskin. The $\boxed{\text{manikin}}_4$ became angry and was pulled into the earth.