Chapter 21
Computational Discourse

Gracie: *Oh yeah... and then Mr. and Mrs. Jones were having matrimonial trouble, and my brother was hired to watch Mrs. Jones.*
George: *Well, I imagine she was a very attractive woman.*
Gracie: *She was, and my brother watched her day and night for six months.*
George: *Well, what happened?*
Gracie: *She finally got a divorce.*
George: *Mrs. Jones?*
Gracie: *No, my brother's wife.*

George Burns and Gracie Allen in *The Salesgirl*

Orson Welles' movie *Citizen Kane* was groundbreaking in many ways, perhaps most notably in its structure. The story of the life of fictional media magnate Charles Foster Kane, the movie does not proceed in chronological order through Kane's life. Instead, the film begins with Kane's death, (famously murmuring *"Rosebud"*), and is structured around flashbacks to his life inserted among scenes of a reporter investigating his death. The novel idea that the structure of a movie does not have to linearly follow the structure of the real timeline made apparent for 20th century cinematography the infinite possibilities and impact of different kinds of coherent narrative structures.

But coherent structure is not just a fact about movies, or works of art. Up to this point of the book, we have focused primarily on language phenomena that operate at the word or sentence level. But just like movies, language does not normally consist of isolated, unrelated sentences, but instead of collocated, structured, **coherent** groups of sentences. We refer to such a coherent structured group of sentences as a **discourse**.

Discourse

Monologue

The chapter you are now reading is an example of a discourse. It is in fact a discourse of a particular sort: a **monologue**. Monologues are characterized by a *speaker* (a term which will be used to include writers, as it is here), and a *hearer* (which, analogously, includes readers). The communication flows in only one direction in a monologue, that is, from the speaker to the hearer.

Dialogue

After reading this chapter, you may have a conversation with a friend about it, which would consist of a much freer interchange. Such a discourse is called a **dialogue**, specifically a **human-human dialogue**. In this case, each participant periodically takes turns being a speaker and hearer. Unlike a typical monologue, dialogues generally consist of many different types of communicative acts: asking questions, giving answers, making corrections, and so forth.

HCI

You may also, for some purposes, such as booking an airline or train trip, have a conversation with a computer **conversational agent**. This use of **human-computer dialogue** for *human-computer interaction*, or **HCI** has properties that distinguish it from normal human-human dialogue, in part due to the present-day limitations on the ability of computer systems to participate in free, unconstrained conversation.

While many discourse processing problems are common to these three forms of discourse, they differ in enough respects that different techniques have often been used to process them. This chapter focuses on techniques commonly applied to the interpretation of monologues; techniques for conversational agents and other dialogues will be described in Ch. 24.

Language is rife with phenomena that operate at the discourse level. Consider the discourse shown in example (21.1).

(21.1) The Tin Woodman went to the Emerald City to see the Wizard of Oz and ask for a heart. After he asked for it, the Woodman waited for the Wizard's response.

What do pronouns such as *he* and *it* denote? No doubt the reader had little trouble figuring out that *he* denotes the Tin Woodman and not the Wizard of Oz, and that *it* denotes the heart and not the Emerald City. Furthermore, it is clear to the reader that *the Wizard* is the same entity as *the Wizard of Oz*, and *the Woodman* is the same as *the Tin Woodman*.

But doing this disambiguation automatically is a difficult task. This goal of deciding what pronouns and other noun phrases refer to is called **coreference resolution**. Coreference resolution is important for **information extraction**, **summarization**, and for **conversational agents**. In fact, it turns out that just about any conceivable language processing application requires methods for determining the denotations of pronouns and related expressions.

There are other important discourse structures beside the relationships between pronouns and other nouns. Consider the task of **summarizing** the following news passage:

(21.2) First Union Corp is continuing to wrestle with severe problems. According to industry insiders at Paine Webber, their president, John R. Georgius, is planning to announce his retirement tomorrow.

We might want to extract a summary like the following:

(21.3) First Union President John R. Georgius is planning to announce his retirement tomorrow.

In order to build such a summary, we need to know that the second sentence is the more important of the two, and that the first sentence is subordinate to it, just giving background information. Relationships of this sort between sentences in a discourse are called **coherence relations**, and determining the coherence structures between discourse sentences is an important discourse task.

Since **coherence** is also a property of a good text, automatically detecting coherence relations is also useful for tasks that measure text quality, like **automatic essay grading**. In automatic essay grading, short student essays are assigned a grade by measuring the internal coherence of the essay as well as comparing its content to source material and hand-labeled high-quality essays. Coherence is also used to evaluate the output quality of natural language generation systems.

Discourse structure and coreference are related in deep ways. Notice that in order to perform the summary above, a system must correctly identify *First Union Corp* as the denotation of *their* (as opposed to *Paine Webber*, for instance). Similarly, it turns out that determining the discourse structure can help in determining coreference.

Coherence

Let's conclude this introduction by discussing what it means for a text to be **coherent**. Assume that you have collected an arbitrary set of well-formed and independently interpretable utterances, for instance, by randomly selecting one sentence from each of the previous chapters of this book. Do you have a discourse? Almost certainly not. The reason is that these utterances, when juxtaposed, will not exhibit **coherence**. Consider, for example, the difference between passages (21.4) and (21.5).

Coherence

(21.4) John hid Bill's car keys. He was drunk.

(21.5) ?? John hid Bill's car keys. He likes spinach.

While most people find passage (21.4) to be rather unremarkable, they find passage (21.5) to be odd. Why is this so? Like passage (21.4), the sentences that make up passage (21.5) are well formed and readily interpretable. Something instead seems to be wrong with the fact that the sentences are juxtaposed. The hearer might ask, for instance, what hiding someone's car keys has to do with liking spinach. By asking this, the hearer is questioning the coherence of the passage.

Alternatively, the hearer might try to construct an explanation that makes it coherent, for instance, by conjecturing that perhaps someone offered John spinach in exchange for hiding Bill's car keys. In fact, if we consider a context in which we had known this already, the passage now sounds a lot better! Why is this? This conjecture allows the hearer to identify John's liking spinach as the cause of his hiding Bill's car keys, which would explain how the two sentences are connected. The very fact that hearers try to identify such connections is indicative of the need to establish coherence as part of discourse comprehension.

In passage (21.4), or in our new model of passage (21.5), the second sentence offers the reader an EXPLANATION or CAUSE for the first sentence. These examples show that a coherent discourse must have meaningful connections between its utterances, connections like EXPLANATION that are often called **coherence relations** and will be introduced in Sec. 21.2.

Coherence relation

Let's introduce a second aspect of coherence by considering the following two texts from Grosz et al. (1995a):

(21.6) a. John went to his favorite music store to buy a piano.
b. He had frequented the store for many years.
c. He was excited that he could finally buy a piano.
d. He arrived just as the store was closing for the day.

(21.7) a. John went to his favorite music store to buy a piano.
b. It was a store John had frequented for many years.
c. He was excited that he could finally buy a piano.
d. It was closing just as John arrived.

While these two texts differ only in how the two entities (John and the store) are realized in the sentences, the discourse in (21.6) is intuitively more coherent than the one in (21.7). As Grosz et al. (1995a) point out, this is because the discourse in (21.6) is clearly about one individual, John, describing his actions and feelings. The discourse in (21.7), by contrast, focuses first on John, then the store, then back to John, then to the store again. It lacks the ‘aboutness’ of the first discourse.

These examples show that for a discourse to be coherent it must exhibit certain kinds of relationships with the entities it is about, introducing them and following them in a focused way. This kind of coherence can be called **entity-based coherence**, We will introduce the **Centering** model of entity-based coherence in Sec. 21.6.2.

In the rest of the chapter we’ll study aspects of both discourse structure and discourse entities. We begin in Sec. 21.1 with the simplest kind of discourse structure: simple **discourse segmentation** of a document into a linear sequence of multiparagraph passages. In Section 21.2, we then introduce more fine-grained discourse structure, the **coherence relation**, and give some algorithms for interpreting these relations. Finally, in Section 21.3, we turn to entities, describing methods for interpreting *referring expressions* such as pronouns.

21.1 Discourse Segmentation

The first kind of discourse task we examine is an approximation to the global or high-level structure of a text or discourse. Many genres of text are associated with particular conventional structures. Academic articles might be divided into sections like Abstract, Introduction, Methodology, Results, Conclusion. A newspaper story is often described as having an inverted pyramid structure, in which the opening paragraphs (the **lede**) contains the most important information. Spoken patient reports are dictated by doctors in four sections following the standard SOAP format (Subjective, Objective, Assessment, Plan).

Lede

Automatically determining all of these types of structures for a large discourse is a difficult and unsolved problem. But some kinds of discourse structure detection algorithms exist. This section introduces one such algorithm, for the simpler problem of **discourse segmentation**; separating a document into a linear sequence of subtopics. Such segmentation algorithms are unable to find sophisticated hierarchical structure. Nonetheless, linear discourse segmentation can be important for **information retrieval**, for example, for automatically segmenting a TV news broadcast or a long news story into a sequence of stories so as to find a relevant story, or for **text summarization** algorithms which need to make sure that different segments of the document are summarized correctly, or for **information extraction** algorithms which tend to extract information from inside a single discourse segment.

Discourse segmentation

In the next two sections we introduce both an unsupervised and a supervised algorithm for discourse segmentation.

21.1.1 Unsupervised Discourse Segmentation

Let’s consider the task of segmenting a text into multi-paragraph units that represent subtopics or passages of the original text. As we suggested above, this task is often called **linear segmentation**, to distinguish it from the task of deriving more sophisticated hierarchical discourse structure. The goal of a segmenter, given raw text, might

Linear segmentation

be to assign subtopic groupings such as the ones defined by Hearst (1997) for the following 21-paragraph science news article called *Stargazers* on the existence of life on earth and other planets (numbers indicate paragraphs):

l-3	Intro - the search for life in space
4–5	The moon's chemical composition
6-8	How early earth-moon proximity shaped the moon
9–12	How the moon helped life evolve on earth
13	Improbability of the earth-moon system
14–16	Binary/trinary star systems make life unlikely
17–18	The low probability of nonbinary/trinary systems
19–20	Properties of earth's sun that facilitate life
21	Summary

Cohesion

Lexical cohesion

An important class of unsupervised algorithms for the linear discourse segmentation task rely on the concept of **cohesion** (Halliday and Hasan, 1976). **Cohesion** is the use of certain linguistic devices to link or tie together textual units. **Lexical cohesion** is cohesion indicated by relations between words in the two units, such as use of an identical word, a synonym, or a hypernym. For example the fact that the words *house*, *shingled*, and *I* occur in both of the two sentences in (21.8ab), is a cue that the two are tied together as a discourse:

(21.8) Before winter **I** built a chimney, and **shingled** the sides of my **house**...
I have thus a tight **shingled** and plastered **house**

In Ex. (21.9), lexical cohesion between the two sentences is indicated by the hypernym relation between *fruit* and the words *pears* and *apples*.

(21.9) Peel, core and slice **the pears and the apples**. Add **the fruit** to the skillet.

There are also non-lexical cohesion relations, such as the use of **anaphora**, shown here between *Woodhouses* and *them* (we will define and discuss anaphora in detail in Sec. 21.6):

(21.10) **The Woodhouses** were first in consequence there. All looked up to **them**.

Cohesion chain

In addition to single examples of lexical cohesion between two words, we can have a **cohesion chain**, in which cohesion is indicated by a whole sequence of related words:

(21.11) Peel, core and slice **the pears and the apples**. Add **the fruit** to the skillet.
When **they** are soft...

Coherence and **cohesion** are often confused; let's review the difference. **Cohesion** refers to the way textual units are tied or linked together. A cohesive relation is like a kind of glue grouping together two units into a single unit. **Coherence** refers to the *meaning* relation between the two units. A coherence relation explains how the meaning of different textual units can combine to jointly build a discourse meaning for the larger unit.

The intuition of the cohesion-based approach to segmentation is that sentences or paragraphs in a subtopic are cohesive with each other, but not with paragraphs in a neighboring subtopic. Thus if we measured the cohesion between every neighboring sentence, we might expect a 'dip' in cohesion at subtopic boundaries.

TextTiling

Let's look at one such cohesion-based approach, the **TextTiling** algorithm (Hearst, 1997). The algorithm has three steps: **tokenization**, **lexical score determination**, and **boundary identification**. In the tokenization stage, each space-delimited word in the input is converted to lower-case, words in a stop list of function words are thrown out, and the remaining words are morphologically stemmed. The stemmed words are grouped into pseudo-sentences of length $w = 20$ (equal-length pseudo-sentences are used rather than real sentences).

Now we look at each gap between pseudo-sentences, and compute a **lexical cohesion score** across that gap. The cohesion score is defined as the average similarity of the words in the pseudo-sentences before gap to the pseudo-sentences after the gap. We generally use a block of $k = 10$ pseudo-sentences on each side of the gap. To compute similarity, we create a word vector b from the block before the gap, and a vector a from the block after the gap, where the vectors are of length N (the total number of non-stop words in the document) and the ith element of the word vector is the frequency of the word w_i. Now we can compute similarity by the cosine (= normalized dot product) measure defined in Eq. 20.47 from Ch. 20, rewritten here:

$$\text{sim}_{\text{cosine}}(\vec{b},\vec{a}) = \frac{\vec{b}\cdot\vec{a}}{|\vec{b}||\vec{a}|} = \frac{\sum_{i=1}^{N} b_i \times a_i}{\sqrt{\sum_{i=1}^{N} b_i^2}\sqrt{\sum_{i=1}^{N} a_i^2}} \tag{21.12}$$

This similarity score (measuring how similar pseudo-sentences $i-k$ to i are to sentences $i+1$ to $i+k+1$) is computed for each gap i between pseudo-sentences. Let's look at the example in Fig. 21.1, where $k = 2$. Fig. 21.1a shows a schematic view of four pseudo-sentences. Each 20-word pseudo-sentence might have multiple true sentences in it; we've shown each with two true sentences. The figure also indicates the computation of the dot-product between successive pseudosentences. Thus for example in the first pseudo-sentence, consisting of sentences 1 and 2, the word A occurs twice, B once, C twice, and so on. The dot product between the first two pseudosentences is $2 \times 1 + 1 \times 1 + 2 \times 1 + 1 \times 1 + 2 \times 1 = 8$. What is the cosine between these first two, assuming all words not shown have zero count?

Finally, we compute a **depth score** for each gap, measuring the depth of the 'similarity valley' at the gap. The depth score is the distance from the peaks on both sides of the valley to the valley; In Fig. 21.1(b), this would be $(y_{a_1} - y_{a_2}) + (y_{a_3} - y_{a_2})$.

Boundaries are assigned at any valley which is deeper than a cutoff threshold (such as $\bar{s} - \sigma$, i.e. one standard deviation deeper than the mean valley depth).

Instead of using these depth score thresholds, more recent cohesion-based segmenters use **divisive clustering** (Choi, 2000; Choi et al., 2001); see the end of the chapter for more information.

21.1.2 Supervised Discourse Segmentation

We've now seen a method for segmenting discourses when no hand-labeled segment boundaries exist. For some kinds of discourse segmentation tasks, however, it is relatively easy to acquire boundary-labeled training data.

Consider the spoken discourse task of segmentation of broadcast news. In order to do summarization of radio or TV broadcasts, we first need to assign boundaries between news stories. This is a simple discourse segmentation task, and training sets

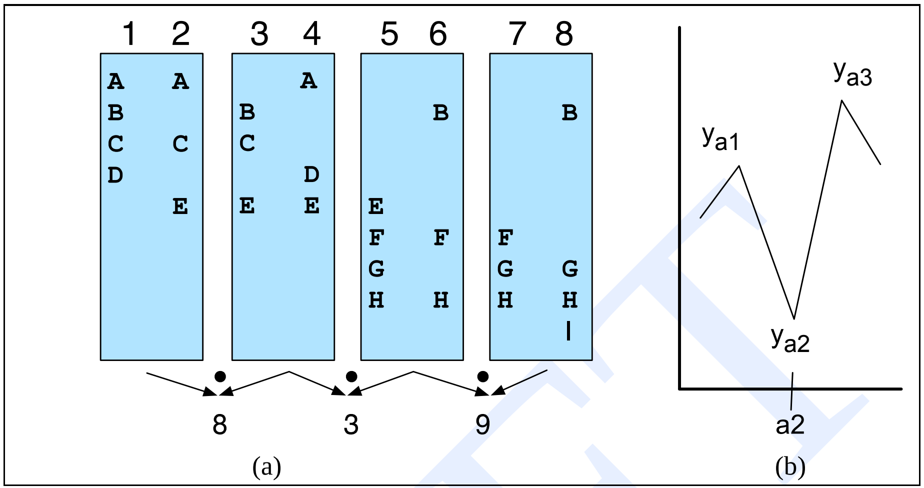


Figure 21.1 The TextTiling algorithm, showing (a) the dot-product computation of similarity between two sentences (1 and 2) and 2 following sentences (3 and 4); capital letters (A, B, C, etc) indicate occurrences of words. (b) shows the computation of the depth score of a valley. After Hearst (1997).

Paragraph segmentation

with hand-labeled news story boundaries exist. Similarly, for speech recognition of monologues like lectures or speeches, we often want to automatically break the text up into paragraphs. For the task of **paragraph segmentation**, it is trivial to find labeled training data from the web (marked with <p>) or other sources.

Every kind of classifier has been used for this kind of supervised discourse segmentation. For example, we can use a binary classifier (SVM, decision tree) and make a yes-no boundary decision between any two sentences. We can also use a sequence classifier (HMM, CRF), making it easier to incorporate sequential constraints.

The features in supervised segmentation are generally a superset of those used in unsupervised classification. We can certainly use cohesion features such as word overlap, word cosine, LSA, lexical chains, coreference, and so on.

Discourse marker

Cue word

A key additional feature that is often used for supervised segmentation is the presence of **discourse markers** or **cue words**. A discourse marker is a word or phrase that functions to signal discourse structure. Discourse markers will play an important role throughout this chapter. For the purpose of broadcast news segmentation, important discourse markers might include a phrase like *good evening, I'm* ⟨*PERSON*⟩, which tends to occur at the beginning of broadcasts, or the word *joining*, which tends to occur in the phrase *joining us now is* ⟨*PERSON*⟩, which often occurs at beginnings of specific segments. Similarly, the cue phrase *coming up* often appears at the end of segments (Reynar, 1999; Beeferman et al., 1999).

Discourse markers tend to be very domain-specific. For the task of segmenting newspaper articles from the Wall Street Journal, for example, the word *incorporated* is a useful feature, since Wall Street Journal articles often start by introducing a company with the full name *XYZ Incorporated*, but later using just *XYZ*. For the task of segmenting out real estate ads, Manning (1998) used discourse cue features like *'is the following word a neighborhood name?'*, *'is previous word a phone number?'* and even punctuation cues like *'is the following word capitalized?'*.

It is possible to write hand-written rules or regular expressions to identify discourse markers for a given domain. Such rules often refer to named entities (like the PERSON examples above), and so a named entity tagger must be run as a preprocessor. Automatic methods for finding discourse markers for segmentation also exist. They first encode all possible words or phrases as features to a classifier, and then doing some sort of **feature selection** on the training set to find only the words that are the best indicators of a boundary (Beeferman et al., 1999; Kawahara et al., 2004).

21.1.3 Evaluating Discourse Segmentation

Discourse segmentation is generally evaluated by running the algorithm on a test set in which boundaries have been labeled by humans. The performance of the algorithm is computed by comparing the automatic and human boundary labels using the *WindowDiff* (Pevzner and Hearst, 2002) or P_k (Beeferman et al., 1999) metrics.

We generally don't use precision, recall and F-measure for evaluating segmentation because they are not sensitive to near misses. Using standard F-measure, if our algorithm was off by one sentence in assigning each boundary, it would get as bad a score as an algorithm which assigned boundaries nowhere near the correct locations. Both *WindowDiff* and P_k assign partial credit. We will present WindowDiff, since it is a more recent improvement to P_k.

WindowDiff compares a reference (human labeled) segmentation with a hypothesis segmentation by sliding a probe, a moving window of length k, across the hypothesis segmentation. At each position in the hypothesis string, we compare the number of **reference** boundaries that fall within the probe (r_i) to the number of **hypothesized** boundaries that fall within the probe (h_i). The algorithm penalizes any hypothesis for which $r_i \neq h_i$, i.e. for which $|r_i - h_i| \neq 0$. The window size k is set as half the average segment in the reference string. Fig. 21.2 shows a schematic of the computation.

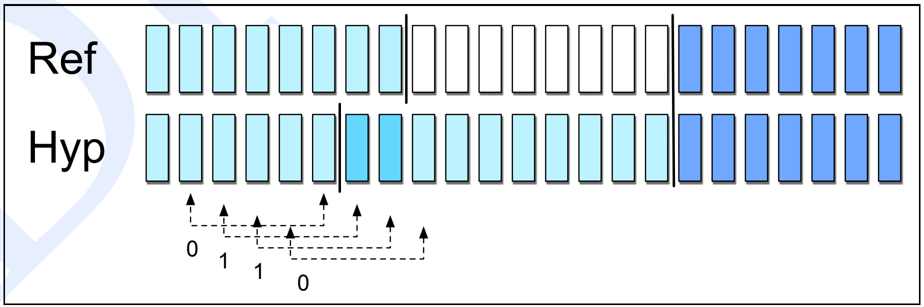


Figure 21.2 The WindowDiff algorithm, showing the moving window sliding over the hypothesis string, and the computation of $|r_i - h_i|$ at four positions. After Pevzner and Hearst (2002).

More formally, if $b(i, j)$ is the number of boundaries between positions i and j in a text, and N is the number of sentences in the text:

(21.13) $$\text{WindowDiff}(ref,hyp) = \frac{1}{N-k}\sum_{i=1}^{N-k}(|b(ref_i,ref_{i+k}) - b(hyp_i,hyp_{i+k})| \neq 0)$$

WindowDiff returns a value between 0 and 1, where 0 indicates that all boundaries are assigned correctly.

21.2 Text Coherence

The previous section showed that cohesive devices, like lexical repetition, can be used to find structure in a discourse. The existence of such devices alone, however, does not satisfy a stronger requirement that a discourse must meet, that of being *coherent*. We briefly introduced coherence in the introduction. In this section we offer more details on what it means for a text to be coherent, and computational mechanisms for determining coherence. We will focus on **coherence relations** and reserve **entity-based coherence** for discussion in Sec. 21.6.2.

Recall from the introduction the difference between passages (21.14) and (21.15).

(21.14) John hid Bill's car keys. He was drunk.

(21.15) ?? John hid Bill's car keys. He likes spinach.

The reason (21.14) is more coherent is that the reader can form a connection between the two utterances, in which the second utterance provides a potential CAUSE or EXPLANATION for the first utterance. This link is harder to form for (21.15). The possible connections between utterances in a discourse can be specified as a set of **coherence relations**. A few such relations, proposed by Hobbs (1979a), are given below. The terms S_0 and S_1 represent the meanings of the two sentences being related.

Coherence relation

Result: Infer that the state or event asserted by S_0 causes or could cause the state or event asserted by S_1.

(21.16) The Tin Woodman was caught in the rain. His joints rusted.

Explanation: Infer that the state or event asserted by S_1 causes or could cause the state or event asserted by S_0.

(21.17) John hid Bill's car keys. He was drunk.

Parallel: Infer $p(a_1,a_2,\ldots)$ from the assertion of S_0 and $p(b_1,b_2,\ldots)$ from the assertion of S_1, where a_i and b_i are similar, for all i.

(21.18) The Scarecrow wanted some brains. The Tin Woodman wanted a heart.

Elaboration: Infer the same proposition P from the assertions of S_0 and S_1.

(21.19) Dorothy was from Kansas. She lived in the midst of the great Kansas prairies.

Occasion: A change of state can be inferred from the assertion of S_0, whose final state can be inferred from S_1, or a change of state can be inferred from the assertion of S_1, whose initial state can be inferred from S_0.

(21.20) Dorothy picked up the oil-can. She oiled the Tin Woodman's joints.

We can also talk about the coherence of an entire discourse, by considering the hierarchical structure between coherence relations. Consider passage (21.21).

(21.21) John went to the bank to deposit his paycheck. (S1)
He then took a train to Bill's car dealership. (S2)
He needed to buy a car. (S3)
The company he works for now isn't near any public transportation. (S4)
He also wanted to talk to Bill about their softball league. (S5)

Intuitively, the structure of passage (21.21) is not linear. The discourse seems to be primarily about the sequence of events described in sentences S1 and S2, whereas sentences S3 and S5 are related most directly to S2, and S4 is related most directly to S3. The coherence relationships between these sentences result in the discourse structure shown in Fig. 21.3.

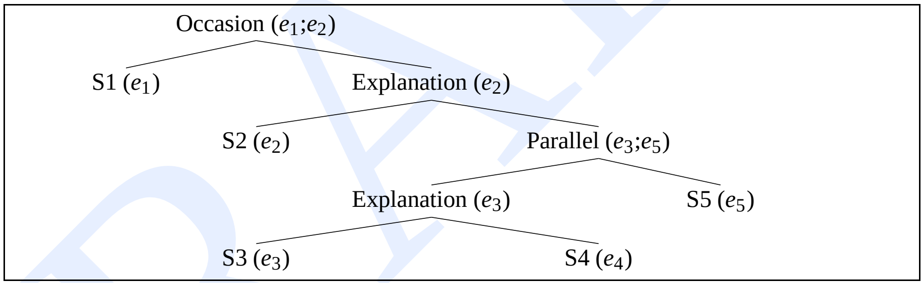


Figure 21.3 The discourse structure of passage (21.21).

Discourse segment

Each node in the tree represents a group of locally coherent clauses or sentences, called a **discourse segment**. Roughly speaking, one can think of discourse segments as being analogous to constituents in sentence syntax.

Now that we've seen examples of coherence, we can see more clearly how a coherence relation can play a role in summarization or information extraction. For example, discourses that are coherent by virtue of the Elaboration relation are often characterized by a summary sentence followed by one or more sentences adding detail to it, as in passage (21.19). Although there are two sentences describing events in this passage, the Elaboration relation tells us that the same event is being described in each. Automatic labeling of the Elaboration relation could thus tell an information extraction or summarization system to merge the information from the sentences and produce a single event description instead of two.

21.2.1 Rhetorical Structure Theory

Rhetorical Structure Theory

RST

Another theory of coherence relations that has received broad usage is **Rhetorical Structure Theory** (**RST**), a model of text organization that was originally proposed for the study of text generation (Mann and Thompson, 1987).

Nucleus

Satellite

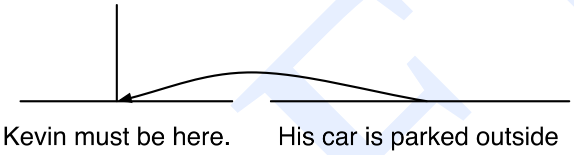
RST is based on a set of 23 *rhetorical relations* that can hold between spans of text within a discourse. Most relations hold between two text spans (often clauses or sentences), a **nucleus** and a **satellite**. The nucleus is the unit that is more central to the writer's purpose, and that is interpretable independently; the satellite is less central, and generally is only interpretable with respect to the nucleus.

Evidence

Consider the **Evidence** relation, in which a satellite presents evidence for the proposition or situation expressed in the nucleus:

(21.22) Kevin must be here. His car is parked outside.

RST relations are traditionally represented graphically; the asymmetric Nucleus-Satellite relation is represented with an arrow from the satellite to the nucleus:



In the original (Mann and Thompson, 1987) formulation, an RST relation is formally defined by a set of **constraints** on the nucleus and satellite, having to do with the goals and beliefs of the writer (W) and reader (R), and by the **effect** on the reader (R). The Evidence relation, for example, is defined as follows:

Relation Name:	Evidence
Constraints on N:	R might not believe N to a degree satisfactory to W
Constraints on S:	R believes S or will find it credible
Constraints on N+S:	R's comprehending S increases R's belief of N
Effects:	R's belief of N is increased

There are many different sets of rhetorical relations in RST and related theories and implementations. The RST TreeBank (Carlson et al., 2001), for example, defines 78 distinct relations, grouped into 16 classes. Here are some common RST relations, with definitions adapted from Carlson and Marcu (2001).

Elaboration: There are various kinds of elaboration relations; in each one, the satellite gives further information about the content of the nucleus:

[$_N$ The company wouldn't elaborate,] [$_S$ citing competitive reasons]

Attribution: The satellite gives the source of attribution for an instance of reported speech in the nucleus.

[$_S$ Analysts estimated,] [$_N$ that sales at U.S. stores declined in the quarter, too]

Contrast: This is a multinuclear relation, in which two or more nuclei contrast along some important dimension:

[$_N$ The priest was in a very bad temper,] [$_N$ but the lama was quite happy.]

List: In this multinuclear relation, a series of nuclei is given, without contrast or explicit comparison:

[$_N$ Billy Bones was the mate;] [$_N$ Long John, he was quartermaster]

Background: The satellite gives context for interpreting the nucleus:

[$_S$ T is the pointer to the root of a binary tree.] [$_N$ Initialize T.]

Just as we saw for the Hobbs coherence relations, RST relations can be hierarchically organized into an entire discourse tree. Fig. 21.4 shows one from Marcu (2000a) for the text in (21.23) from the Scientific American magazine.

(21.23) With its distant orbit–50 percent farther from the sun than Earth–and slim atmospheric blanket, Mars experiences frigid weather conditions. Surface temperatures typically average about -60 degrees Celsius (-76 degrees Fahrenheit) at the equator and can dip to -123 degrees C near the poles. Only the midday sun at tropical latitudes is warm enough to thaw ice on occasion, but any liquid water formed in this way would evaporate almost instantly because of the low atmospheric pressure.

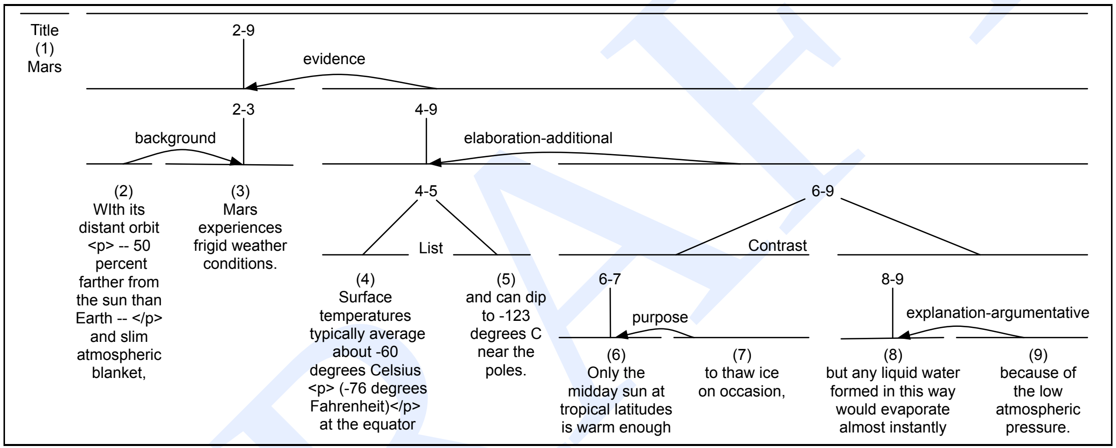

Title
(1)
Mars
2-9
evidence
2-3
4-9
background
elaboration-additional
(2) WIth its distant orbit <p> -- 50 percent farther from the sun than Earth -- </p> and slim atmospheric blanket,
(3) Mars experiences frigid weather conditions.
4-5
6-9
List
Contrast
(4) Surface temperatures typically average about -60 degrees Celsius <p> (-76 degrees Fahrenheit)</p> at the equator
(5) and can dip to -123 degrees C near the poles.
6-7
purpose
8-9
explanation-argumentative
(6) Only the midday sun at tropical latitudes is warm enough
(7) to thaw ice on occasion,
(8) but any liquid water formed in this way would evaporate almost instantly
(9) because of the low atmospheric pressure.

Figure 21.4 A discourse tree for the Scientific American text in (21.23), from Marcu (2000a). Note that asymmetric relations are represented with a curved arrow from the satellite to the nucleus.

See the end of the chapter for pointers to other theories of coherence relations and related corpora, and Ch. 23 for the application of RST and similar coherence relations to summarization.

21.2.2 Automatic Coherence Assignment

Given a sequence of sentences, how can we automatically determine the coherence relations between them? Whether we use RST, Hobbs, or one of the many other sets of relations (see the end of the chapter), we call this task **coherence relation assignment**. If we extend this task from assigning a relation between two sentences to the larger goal of extracting a tree or graph representing an entire discourse, the term **discourse parsing** is often used.

Discourse parsing

Both of these tasks are quite difficult, and remain unsolved open research problems. Nonetheless, a variety of methods have been proposed, and in this section we describe shallow algorithms based on **cue phrases**. In the following section we sketch a more sophisticated but less robust algorithm based on **abduction**.

A shallow cue-phrase-based algorithm for coherence extraction has three stages:

1. Identify the cue phrases in a text
2. Segment the text into discourse segments, using cue phrases
3. Classify the relationship between each consecutive discourse segment, using cue phrases.

Cue phrase

Discourse marker

Connective

We said earlier that a **cue phrase** (or **discourse marker** or **cue word**) is a word or phrase that functions to signal discourse structure, especially by linking together discourse segments. In Sec. 21.1 we mentioned cue phrases or features like *joining us now is* ⟨*PERSON*⟩ (for broadcast news segmentation) or *following word is the name of a neighborhood* (for real estate ad segmentation). For extracting coherence relations, we rely on cue phrases called **connectives**, which are often conjunctions or adverbs, and which give us a 'cue' to the coherence relations that hold between segments. For example, the connective *because* strongly suggests the EXPLANATION relation in passage (21.24).

(21.24) John hid Bill's car keys because he was drunk.

Sentential

Other such cue phrases include *although*, *but*, *for example*, *yet*, *with*, and *and*. Discourse markers can be quite ambiguous between these **discourse** uses and non-discourse related **sentential** uses. For example, the word *with* can be used as a cue phrase as in (21.25), or in a sentential use as in (21.26)[1]:

(21.25) **With** its distant orbit, Mars exhibits frigid weather conditions

(21.26) We can see Mars **with** an ordinary telescope.

Some simple disambiguation of the discourse versus sentential use of a cue phrase can be done with simple regular expressions, once we have sentence boundaries. For example, if the words *With* or *Yet* are capitalized and sentence-initial, they tend to be discourse markers. The words *because* or *where* tend to be discourse markers if preceded by a comma. More complete disambiguation requires the WSD techniques of Ch. 20 using many other features. If speech is available, for example, discourse markers often bear different kinds of pitch accent than sentential uses (Hirschberg and Litman, 1993).

The second step in determining the correct coherence relation is to segment the text into **discourse segments**. Discourse segments generally correspond to clauses or sentences, although sometimes they are smaller than clauses. Many algorithms approximate segmentation by using entire sentences, employing the sentence segmentation algorithm of Fig. 3.22 (page 71), or the algorithm of Sec. 8.1.1.

Often, however, a clause or clause-like unit is a more appropriate size for a discourse segment, as we see in the following examples from Sporleder and Lapata (2004):

(21.27) [We can't win] [but we must keep trying] (CONTRAST)

(21.28) [The ability to operate at these temperature is advantageous], [because the devices need less thermal insulation] (EXPLANATION)

One way to segment these clause-like units is to use hand-written segmentation rules based on individual cue phrases. For example, if the cue-phrase *Because* occurs

[1] Where perhaps it will be a cue instead for the semantic role INSTRUMENT

sentence-initially and is eventually followed by a comma (as in (21.29)), it may begin a segment (terminated by the comma) that relates to the clause after the comma. If *because* occurs sentence-medially, it may divide the sentence into a previous and following discourse segment (as in (21.30)). These cases can be distinguished by handwritten rules based on punctuation and sentence boundaries.

(21.29) [Because of the low atmospheric pressure,] [any liquid water would evaporate instantly]

(21.30) [Any liquid water would evaporate instantly] [because of the low atmospheric pressure.]

If a syntactic parser is available, we can write more complex segmentation rules making use of syntactic phrases.

The third step in coherence extraction is to automatically classify the relation between each pair of neighboring segments. We can again write rules for each discourse marker, just as we did for determining discourse segment boundaries. Thus a rule could specify that a segmenting beginning with sentence-initial *Because* is a satellite in a CAUSE relationship with a nucleus segment that follows the comma.

In general, the rule-based approach to coherence extraction does not achieve extremely high accuracy. Partly this is because cue phrases are ambiguous; *because*, for example, can indicate both CAUSE and EVIDENCE, *but* can indicate CONTRAST, ANTITHESIS, and CONCESSION, and so on. We need additional features than just the cue phrases themselves. But a deeper problem with the rule-based method is that many coherence relations are not signaled by cue phrases at all. In the RST corpus of Carlson et al. (2001), for example, Marcu and Echihabi (2002) found that only 61 of the 238 CONTRAST relations, and only 79 of the 307 EXPLANATION-EVIDENCE relations, were indicated by explicit cue phrases. Instead, many coherence relations are signalled by more implicit cues. For example, the following two sentences are in the CONTRAST relation, but there is no explicit *in contrast* or *but* connective beginning the second sentence:

(21.31) The $6 billion that some 40 companies are looking to raise in the year ending March 31 compares with only $2.7 billion raised on the capital market in the previous fiscal year

(21.32) In fiscal 1984 before Mr. Gandhi came to power, only $810 million was raised.

How can we extract coherence relations between discourse segments if no cue phrases exist? There are certainly many implicit cues that we could use. Consider the following two discourse segments:

(21.33) [I don’t want a truck;] [I’d prefer a convertible.]

The CONTRAST relation between these segments is signalled by their syntactic parallelism, by the use of negation in the first segment, and by the lexical coordinate relation between *convertible* and *truck*. But many of these features are quite lexical, requiring a large number of parameters which couldn’t be trained on the small amount of labeled coherence relation data that currently exists.

This suggests the use of **bootstrapping** to automatically label a larger corpus with coherence relations that could then be used to train these more expensive features. We can do this by relying on discourse markers that are very strong unambiguous cues for particular relations. For example *consequently* is an unambiguous signal for RESULT, *in other words* for SUMMARY, *for example* for ELABORATION, and *secondly* for CONTINUATION. We write regular expressions to extract pairs of discourse segments surrounding these cue phrases, and then remove the cue phrases themselves. The resulting sentence pairs, without the cue phrases, are used as a supervised training set for these coherence relations.

Given this labeled training set, any supervised machine learning method may be used. Marcu and Echihabi (2002), for example, use a naive Bayes classifier based only on word-pair features (w_1, w_2), where the first word w_1 occurs in the first discourse segment, and the second w_2 occurs in the following segment. This feature captures lexical relations like *convertible/truck* above. Sporleder and Lascarides (2005) include other features, including individual words, parts of speech, or stemmed words in the left and right discourse segment. They found, for example, that words like *other*, *still*, and *not* were chosen by feature selection as good cues for CONTRAST. Words like *so*, *indeed*, and *undoubtedly* were chosen as cues for RESULT.

21.3 Reference Resolution

and even Stigand, the patriotic archbishop of Canterbury, found it advisable–”’

‘Found WHAT?’ said the Duck.

‘Found IT,’ the Mouse replied rather crossly: ‘of course you know what ”it” means.’

‘I know what “it” means well enough, when I find a thing,’ said the Duck: ‘it’s generally a frog or a worm. The question is, what did the archbishop find?

Lewis Carroll, Alice in Wonderland

In order to interpret the sentences of any discourse, we need to know who or what entity is being talked about. Consider the following passage:

(21.34) Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20%, to $1.3 million, as the 37-year-old also became the Denver-based financial-services company’s president. It has been ten years since she came to Megabucks from rival Lotsabucks.

Reference

Reference resolution

In this passage, each of the underlined phrases is used by the speaker to denote one person named Victoria Chen. We refer to this use of linguistic expressions like *her* or *Victoria Chen* to denote an entity or individual as **reference**. In the next few sections of this chapter we study the problem of **reference resolution**. Reference resolution is the task of determining what entities are referred to by which linguistic expressions.

Referring expression

Referent

We first define some terminology. A natural language expression used to perform reference is called a **referring expression**, and the entity that is referred to is called the **referent**. Thus, *Victoria Chen* and *she* in passage (21.34) are referring expressions, and Victoria Chen is their referent. (To distinguish between referring expressions and their referents, we italicize the former.) As a convenient shorthand, we will sometimes speak of a referring expression referring to a referent, e.g., we might say that *she* refers

Corefer
Antecedent
Anaphora
Anaphoric

to Victoria Chen. However, the reader should keep in mind that what we really mean is that the speaker is performing the act of referring to Victoria Chen by uttering *she*. Two referring expressions that are used to refer to the same entity are said to **corefer**; thus *Victoria Chen* and *she* corefer in passage (21.34). There is also a term for a referring expression that licenses the use of another, in the way that the mention of *John* allows John to be subsequently referred to using *he*. We call *John* the **antecedent** of *he*. Reference to an entity that has been previously introduced into the discourse is called **anaphora**, and the referring expression used is said to be **anaphoric**. In passage (21.34), the pronouns *she* and *her*, and the definite NP *the 37-year-old* are therefore anaphoric.

Discorse context
Situational context

Natural languages provide speakers with a variety of ways to refer to entities. Say that your friend has a 1961 Ford Falcon automobile and you want to refer to it. Depending on the operative **discourse context**, you might say *it*, *this*, *that*, *this car*, *that car*, *the car*, *the Ford*, *the Falcon*, or *my friend's car*, among many other possibilities. However, you are not free to choose between any of these alternatives in any context. For instance, you cannot simply say *it* or *the Falcon* if the hearer has no prior knowledge of your friend's car, it has not been mentioned before, and it is not in the immediate surroundings of the discourse participants (i.e., the **situational context** of the discourse).

Discourse model

The reason for this is that each type of referring expression encodes different signals about the place that the speaker believes the referent occupies within the hearer's set of beliefs. A subset of these beliefs that has a special status form the hearer's mental model of the ongoing discourse, which we call a **discourse model** (Webber, 1978). The discourse model contains representations of the entities that have been referred to in the discourse and the relationships in which they participate. Thus, there are two components required by a system to successfully interpret (or produce) referring expressions: a method for constructing a discourse model that evolves with the dynamically-changing discourse it represents, and a method for mapping between the signals that various referring expressions encode and the hearer's set of beliefs, the latter of which includes this discourse model.

Evoke
Access

We will speak in terms of two fundamental operations to the discourse model. When a referent is first mentioned in a discourse, we say that a representation for it is **evoked** into the model. Upon subsequent mention, this representation is **accessed** from the model. The operations and relationships are illustrated in Fig. 21.5. As we will see in Sec. 21.8, the discourse model plays an important role in how coreference algorithms are evaluated.

Coreference resolution
Coreference chain

We are now ready to introduce two reference resolution tasks: **coreference resolution** and **pronominal anaphora resolution**. Coreference resolution is the task of finding referring expressions in a text that refer to the same entity, i.e. finding expressions that **corefer**. We call the set of coreferring expressions a **coreference chain**. For example, in processing (21.34), a coreference resolution algorithm would need to find four coreference chains:

1. { *Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 1994, her, the 37-year-old, the Denver-based financial-services company's president, She*}
2. { *Megabucks Banking Corp, the Denver-based financial-services company, Megabucks* }

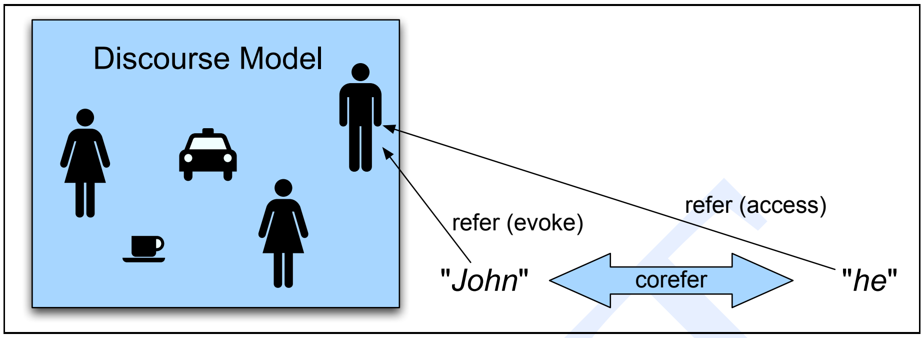


Figure 21.5 Reference operations and relationships with respect to the discourse model.

3. { *her pay* }
4. { *Lotsabucks* }

Coreference resolution thus requires finding all referring expressions in a discourse, and grouping them into coreference chains. By contrast, **pronominal anaphora resolution** is the task of finding the antecedent for a single pronoun; for example, given the pronoun *her*, our task is to decide that the antecedent of *her* is *Victoria Chen*. Thus pronominal anaphora resolution can be viewed as a subtask of coreference resolution.[2]

anaphora resolution

In the next section we introduce different kinds of reference phenomena. We then give various algorithms for reference resolution. Pronominal anaphora has received a lot of attention in speech and language processing, and so we will introduce three algorithms for pronoun processing: the **Hobbs** algorithm, a **Centering** algorithm, and a **log-linear** (MaxEnt) algorithm. We then give an algorithm for the more general coreference resolution task.

We will see that each of these algorithms focuses on resolving reference to entities or individuals. It is important to note, however, that discourses do include reference to many other types of referents than entities. Consider the possibilities in example (21.35), adapted from Webber (1991).

(21.35) According to Doug, Sue just bought a 1961 Ford Falcon.

a. But *that* turned out to be a lie.

b. But *that* was false.

c. *That* struck me as a funny way to describe the situation.

d. *That* caused a financial problem for Sue.

The referent of *that* is a speech act (see Ch. 24) in (21.35a), a proposition in (21.35b), a manner of description in (21.35c), and an event in (21.35d). The field awaits the development of robust methods for interpreting these types of reference.

[2] Although technically there are cases of anaphora that are not cases of coreference; see van Deemter and Kibble (2000) for more discussion.

21.4 Reference Phenomena

The set of referential phenomena that natural languages provide is quite rich indeed. In this section, we provide a brief description of several basic reference phenomena, surveying five types of referring expression: *indefinite noun phrases*, *definite noun phrases*, *pronouns*, *demonstratives*, and *names*. We then summarize the way these referring expressions are used to encode **given** and **new** information, along the way introducing two types of referents that complicate the reference resolution problem: *inferrables* and *generics*.

21.4.1 Five Types of Referring Expressions

Indefinite Noun Phrases Indefinite reference introduces entities that are new to the hearer into the discourse context. The most common form of indefinite reference is marked with the determiner *a* (or *an*), but it can also be marked by a quantifier such as *some* or even the determiner *this*:

(21.36) (a) Mrs. Martin was so very kind as to send Mrs. Goddard *a beautiful goose*.

(b) He had gone round one day to bring her *some walnuts*.

(c) I saw *this beautiful Ford Falcon* today.

Such noun phrases evoke a representation for a new entity that satisfies the given description into the discourse model.

The indefinite determiner *a* does not indicate whether the entity is identifiable to the speaker, which in some cases leads to a *specific/non-specific* ambiguity. Example (21.36a) only has the specific reading, since the speaker has a particular goose in mind, particularly the one Mrs. Martin sent. In sentence (21.37), on the other hand, both readings are possible.

(21.37) I am going to the butchers to buy a goose.

That is, the speaker may already have the goose picked out (specific), or may just be planning to pick one out that is to her liking (nonspecific).

Definite Noun Phrases Definite reference is used to refer to an entity that is identifiable to the hearer. An entity can be identifiable to the hearer because it has been mentioned previously in the text, and thus is already represented in the discourse model:

(21.38) It concerns a white stallion which I have sold to an officer. But the pedigree of *the white stallion* was not fully established.

Alternatively, an entity can be identifiable because it is contained in the hearer's set of beliefs about the world, or the uniqueness of the object is implied by the description itself, in which case it evokes a representation of the referent into the discourse model, as in (21.39):

(21.39) I read about it in *The New York Times*.

Pronouns Another form of definite reference is pronominalization, illustrated in example (21.40).

(21.40) Emma smiled and chatted as cheerfully as *she* could,

Salience

The constraints on using pronominal reference are stronger than for full definite noun phrases, requiring that the referent have a high degree of activation or **salience** in the discourse model. Pronouns usually (but not always) refer to entities that were introduced no further than one or two sentences back in the ongoing discourse, whereas definite noun phrases can often refer further back. This is illustrated by the difference between sentences (21.41d) and (21.41d').

(21.41) a. John went to Bob's party, and parked next to a classic Ford Falcon.
b. He went inside and talked to Bob for more than an hour.
c. Bob told him that he recently got engaged.
d. ?? He also said that he bought *it* yesterday.
d.' He also said that he bought *the Falcon* yesterday.

By the time the last sentence is reached, the Falcon no longer has the degree of salience required to allow for pronominal reference to it.

Cataphora

Pronouns can also participate in **cataphora**, in which they are mentioned before their referents are, as in example (21.42).

(21.42) Even before *she* saw *it*, Dorothy had been thinking about the Emerald City every day.

Here, the pronouns *she* and *it* both occur *before* their referents are introduced.

Bound

Pronouns also appear in quantified contexts in which they are considered to be **bound**, as in example (21.43).

(21.43) Every dancer brought *her* left arm forward.

Under the relevant reading, *her* does not refer to some woman in context, but instead behaves like a variable bound to the quantified expression *every dancer*. We will not be concerned with the bound interpretation of pronouns in this chapter.

Proximal

Distal

Demonstratives Demonstrative pronouns, like *this* and *that*, behave somewhat differently than simple definite pronouns like *it*. They can appear either alone or as determiners, for instance, *this ingredient*, *that spice*. *This* and *that* differ in lexical meaning; (*this*, the **proximal demonstrative**, indicating literal or metaphorical closeness, while *that*, the **distal demonstrative** indicating literal or metaphorical distance (further away in time, as in the following example)):

(21.44) I just bought a copy of Thoreau's *Walden*. I had bought one five years ago. *That one* had been very tattered; *this one* was in much better condition.

Note that *this NP* is ambiguous; in colloquial spoken English, it can be indefinite, as in (21.36), or definite, as in (21.44).

Names Names are a very common form of referring expression, including names of people, organizations, and locations, as we will see in the discussion of named entities in Sec. 22.1. Names can be used to refer to both new and old entities in the discourse:

(21.45) a. **Miss Woodhouse** certainly had not done him justice.

b. **International Business Machines** sought patent compensation from Amazon; **I.B.M.** had previously sued other companies.

21.4.2 Information Status

We noted above that the same referring expressions (such as many indefinite NPs) can be used to introduce new referents, while other expressions (such as many definite NPs, or pronouns) can be used to refer anaphorically to old referents. This idea of studying the way different referential forms are used to provide new or old information is called **information status** or **information structure**.

Information status

Information structure

There are a variety of theories that express the relation between different types of referential form and the informativity or saliency of the referent in the discourse. For example, the **givenness hierarchy** (Gundel et al., 1993) is a scale representing six kinds of information status that different referring expression are used to signal:

Givenness hierarchy

The givenness hierarchy:

in focus >	activated >	familiar >	uniquely identifiable >	referential >	type identifiable
{it}	*that* *this* *this* N	{that N}	{the N}	{indef. *this* N}	{*a* N}

Accessibility scale

The related **accessibility scale** of Ariel (2001) is based on the idea that referents that are more salient will be easier for the hearer to call to mind, and hence can be referred to with less linguistic material. By contrast, less salient entities will need a longer and more explicit referring expression to help the hearer recover the referent. The following shows a sample scale going from low to high accessibility:

> *Full name* > *long definite description* > *short definite description* > *last name* > *first name* > *distal demonstrative* > *proximate demonstrative* > *NP* > *stressed pronoun* > *unstressed pronoun*

Note that accessibility correlates with length, with less accessible NPs tending to be longer. Indeed, if we follow a coreference chain in a discourse, we will often find longer NPs (for example long definition descriptions with relative clauses) early in the discourse, and much shorter ones (for example pronouns) later in the discourse.

Another perspective, based on the work of (Prince, 1992), is to analyze information status in terms of two crosscutting dichotomies: *hearer status* and *discourse status*. The *hearer status* of a referent expresses whether it is previously known to the hearer, or whether it is new. The *discourse status* expresses whether the referent has been previously mentioned in the discourse.

The relationship between referring expression form and information status can be complicated; we summarize below three such complicating factors (the use of **inferrables**, **generics**, and **non-referential forms**):

Inferrables: In some cases, a referring expression does not refer to an entity that has been explicitly evoked in the text, but instead one that is inferentially related to an evoked entity. Such referents are called **inferrables**, **bridging inferences**, or **mediated**

Inferrbale

Bridging inference

Mediated

(Haviland and Clark, 1974; Prince, 1981; Nissim et al., 2004) Consider the expressions *a door* and *the engine* in sentence (21.46).

(21.46) I almost bought a 1961 Ford Falcon today, but *a door* had a dent and *the engine* seemed noisy.

The indefinite noun phrase *a door* would normally introduce a new door into the discourse context, but in this case the hearer is to infer something more: that it is not just any door, but one of the doors of the Falcon. Similarly, the use of the definite noun phrase *the engine* normally presumes that an engine has been previously evoked or is otherwise uniquely identifiable. Here, no engine has been explicitly mentioned, but the hearer makes a **bridging inference** to infer that the referent is the engine of the previously mentioned Falcon.

Generics: Another kind of expression that does not refer back to an entity explicitly evoked in the text is *generic* reference. Consider example (21.47).

(21.47) I'm interested in buying a Mac laptop. *They* are very stylish.

Here, *they* refers, not to a particular laptop (or even a particular set of laptops), but instead to the class of Mac laptops in general. Similarly, the pronoun *you* can be used generically in the following example:

(21.48) In March in Boulder *you* have to wear a jacket.

Non-referential uses: Finally, some non-referential forms bear a confusing superficial resemblance to referring expressions. For example in addition to its referring usages, the word *it* can be used in **pleonastic** cases like *it is raining*, in idioms like *hit it off*, or in particular syntactic situations like **clefts** (21.49a) or **extraposition** (21.49b):

Pleonastic
Cleft

(21.49) (a) *It* was Frodo who carried the ring.
(b) *It* was good that Frodo carried the ring.

21.5 Features for Pronominal Anaphora Resolution

We now turn to the task of resolving pronominal reference. In general, this problem is formulated as follows. We are given a single pronoun (*he*, *him*, *she*, *her*, *it*, and sometimes *they/them*), together with the previous context. Our task is to find the antecedent of the pronoun in this context. We present three systems for this task; but first we summarize useful constraints on possible referents.

We begin with five relatively hard-and-fast morphosyntactic features that can be used to filter the set of possible referents: **number**, **person**, **gender**, and **binding theory** constraints.

Number Agreement: Referring expressions and their referents must agree in number; for English, this means distinguishing between *singular* and *plural* references. English *she/her/he/him/his/it* are singular, *we/us/they/them* are plural, and *you* is unspecified for number. Some illustrations of the constraints on number agreement:

John has a Ford Falcon. It is red.
John has three Ford Falcons. They are red.

* John has a Ford Falcon. They are red.
* John has three Ford Falcons. It is red.

We cannot always enforce a very strict grammatical notion of number agreement, since sometimes semantically plural entities can be referred to by either *it* or *they*:

(21.50) IBM announced a new machine translation product yesterday. *They* have been working on it for 20 years.

Person Agreement: English distinguishes between three forms of person: first, second, and third. The antecedent of a pronoun must agree with the pronoun in number. A first person pronoun (*I*, *me*, *my*) must have a first person antecedent (*I*, *me*, or *my*). A second person pronoun (*you* or *your*) must have a second person antecedent (*you* or *your*). A third person pronoun (*he, she, they, him, her, them, his, her, their*) must have a third person antecedent (one of the above or any other noun phrase).

Gender Agreement: Referents also must agree with the gender specified by the referring expression. English third person pronouns distinguish between *male*, (*he, him, his*), *female*, (*she, her*) and *nonpersonal* (*it*) genders. Unlike in some languages, English male and female pronoun genders only apply to animate entities; inanimate entities are always nonpersonal/neuter. Some examples:

(21.51) John has a Ford. He is attractive. (he=John, not the Ford)

(21.52) John has a Ford. It is attractive. (it=the Ford, not John)

Binding Theory Constraints: Reference relations may also be constrained by the syntactic relationships between a referential expression and a possible antecedent noun phrase when both occur in the same sentence. For instance, the pronouns in all of the following sentences are subject to the constraints indicated in brackets.

(21.53) John bought himself a new Ford. [himself=John]

(21.54) John bought him a new Ford. [him≠John]

(21.55) John said that Bill bought him a new Ford. [him≠Bill]

(21.56) John said that Bill bought himself a new Ford. [himself=Bill]

(21.57) He said that he bought John a new Ford. [He≠John; he≠John]

Reflexive

English pronouns such as *himself*, *herself*, and *themselves* are called **reflexives**. Oversimplifying the situation, a reflexive corefers with the subject of the most immediate clause that contains it (ex. 21.53), whereas a nonreflexive cannot corefer with this subject (ex. 21.54). That this rule applies only for the subject of the most immediate clause is shown by examples (21.55) and (21.56), in which the opposite reference pattern is manifest between the pronoun and the subject of the higher sentence. On the other hand, a full noun phrase like *John* cannot corefer with the subject of the most immediate clause nor with a higher-level subject (ex. 21.57).

Binding theory

These constraints are often called the **binding theory** (Chomsky, 1981), and quite complicated versions of these constraints have been proposed. A complete statement of the constraints requires reference to semantic and other factors, and cannot be stated purely in terms of syntactic configuration. Nonetheless, for the algorithms discussed later in this chapter we will assume a simple syntactic account of restrictions on intrasentential coreference.

Selectional Restrictions: The selectional restrictions that a verb places on its arguments (see Ch. 19) may be responsible for eliminating referents, as in example (21.58).

(21.58) John parked his car in the garage after driving it around for hours.

There are two possible referents for *it*, the car and the garage. The verb *drive*, however, requires that its direct object denote something that can be driven, such as a car, truck, or bus, but not a garage. Thus, the fact that the pronoun appears as the object of *drive* restricts the set of possible referents to the car. Selectional restrictions can be implemented by storing a dictionary of probabilistic dependencies between the verb associated with the pronoun and the potential referent.

Recency: We next turn to features for predicting the referent of a pronoun that are less hard-and-fast. Entities introduced in recent utterances tend to be more salient than those introduced from utterances further back. Thus, in example (21.59), the pronoun *it* is more likely to refer to Jim's map than the doctor's map.

(21.59) The doctor found an old map in the captain's chest. Jim found an even older map hidden on the shelf. It described an island.

Grammatical Role: Many theories specify a salience hierarchy of entities that is ordered by the grammatical position of the expressions which denote them. These typically treat entities mentioned in subject position as more salient than those in object position, which are in turn more salient than those mentioned in subsequent positions.

Passages such as (21.60) and (21.61) lend support for such a hierarchy. Although the first sentence in each case expresses roughly the same propositional content, the preferred referent for the pronoun *he* varies with the subject in each case – John in (21.60) and Bill in (21.61).

(21.60) Billy Bones went to the bar with Jim Hawkins. He called for a glass of rum. [he = Billy]

(21.61) Jim Hawkins went to the bar with Billy Bones. He called for a glass of rum. [he = Jim]

Repeated Mention: Some theories incorporate the idea that entities that have been focused on in the prior discourse are more likely to continue to be focused on in subsequent discourse, and hence references to them are more likely to be pronominalized. For instance, whereas the pronoun in example (21.61) has Jim as its preferred interpretation, the pronoun in the final sentence of example (21.62) may be more likely to refer to Billy Bones.

(21.62) Billy Bones had been thinking about a glass of rum ever since the pirate ship docked. He hobbled over to the Old Parrot bar. Jim Hawkins went with him. He called for a glass of rum. [he = Billy]

Parallelism: There are also strong preferences that appear to be induced by parallelism effects, as in example (21.63).

(21.63) Long John Silver went with Jim to the Old Parrot. Billy Bones went with him to the Old Anchor Inn. [him = Jim]

The grammatical role hierarchy described above ranks Long John Silver as more salient than Jim, and thus should be the preferred referent of *him*. Furthermore, there is no semantic reason that Long John Silver cannot be the referent. Nonetheless, *him* is instead understood to refer to Jim.

Verb Semantics Certain verbs appear to place a semantically-oriented emphasis on one of their argument positions, which can have the effect of biasing the manner in which subsequent pronouns are interpreted. Compare sentences (21.64) and (21.65).

(21.64) John telephoned Bill. He lost the laptop.

(21.65) John criticized Bill. He lost the laptop.

These examples differ only in the verb used in the first sentence, yet the subject pronoun in passage (21.64) is typically resolved to John, whereas the pronoun in passage (21.65) is resolved to Bill. It has been argued that this effect results from what the "implicit causality" of a verb: the implicit cause of a "criticizing" event is considered to be its object, whereas the implicit cause of a "telephoning" event is considered to be its subject. This emphasis results in a higher degree of salience for the entity in this argument position.

21.6 Three algorithms for pronominal anaphora resolution

21.6.1 Pronominal Anaphora Baseline: The Hobbs Algorithm

Hobbs algorithm

The first of the three algorithms we present for pronominal anaphora resolution is the **Hobbs algorithm**. The Hobbs algorithm (the simpler of two algorithms presented originally in Hobbs (1978)) depends only on a syntactic parser plus a morphological gender and number checker. For this reason it is often used as a baseline when evaluating new pronominal anaphora resolution algorithms.

The input to the Hobbs algorithm is a pronoun to be resolved, together with a syntactic parse of the sentences up to and including the current sentence. The algorithm searches for an antecedent noun phrase in these trees. The intuition of the algorithm is to start with the target pronoun and walk up the parse tree to the root *S*. For each *NP* or *S* node that it finds, it does a breadth-first left-to-right search of the node's children to the left of the target. As each candidate noun phrase is proposed, it is checked for gender, number, and person agreement with the pronoun. If no referent is found, the algorithm performs the same breadth-first search on preceding sentences.

The Hobbs algorithm does not capture all the constraints and preferences on pronominalization described above. It does, however, approximate the *binding theory*, *recency*, and *grammatical role* preferences by the order in which the search is performed, and the *gender*, *person*, and *number* constraints by a final check.

An algorithm that searches parse trees must also specify a grammar, since the assumptions regarding the structure of syntactic trees will affect the results. A fragment for English that the algorithm uses is given in Fig. 21.6. The steps of the **Hobbs algorithm** are as follows:

1. Begin at the noun phrase (NP) node immediately dominating the pronoun.

$$
\begin{aligned}
& S \rightarrow NP\ VP \\
& NP \rightarrow \left\{ \begin{array}{l} (Det)\quad Nominal \quad \left(\left\{ \begin{array}{l} PP \\ Rel \end{array} \right\} \right)^{*} \\ pronoun \end{array} \right\} \\
& Det \rightarrow \left\{ \begin{array}{l} determiner \\ NP\ \text{'s} \end{array} \right\} \\
& PP \rightarrow preposition\ NP \\
& Nominal \rightarrow noun\ (PP)^{*} \\
& Rel \rightarrow wh\text{-}word\ S \\
& VP \rightarrow verb\ NP\ (PP)^{*}
\end{aligned}
$$

Figure 21.6 A grammar fragment for the Tree Search algorithm.

2. Go up the tree to the first NP or sentence (S) node encountered. Call this node X, and call the path used to reach it *p*.
3. Traverse all branches below node X to the left of path *p* in a left-to-right, breadth-first fashion. Propose as the antecedent any NP node that is encountered which has an NP or S node between it and X.
4. If node X is the highest S node in the sentence, traverse the surface parse trees of previous sentences in the text in order of recency, the most recent first; each tree is traversed in a left-to-right, breadth-first manner, and when an NP node is encountered, it is proposed as antecedent. If X is not the highest S node in the sentence, continue to step 5.
5. From node X, go up the tree to the first NP or S node encountered. Call this new node X, and call the path traversed to reach it *p*.
6. If X is an NP node and if the path *p* to X did not pass through the Nominal node that X immediately dominates, propose X as the antecedent.
7. Traverse all branches below node X to the *left* of path p in a left-to-right, breadth-first manner. Propose any NP node encountered as the antecedent.
8. If X is an S node, traverse all branches of node X to the *right* of path *p* in a left-to-right, breadth-first manner, but do not go below any NP or S node encountered. Propose any NP node encountered as the antecedent.
9. Go to Step 4.

Demonstrating that this algorithm yields the correct coreference assignments for an example sentence is left as Exercise 2.

Most parsers return number information (singular or plural), and person information is easily encoded by rule for the first and second person pronouns. But parsers for English rarely return gender information for common or proper nouns. Thus the only additional requirement to implementing the Hobbs algorithm, besides a parser, is an algorithm for determining gender for each antecedent noun phrase.

One common way to assign gender to a noun phrase is to extract the head noun, and then use WordNet (Ch. 19) to look at the hypernyns of the head noun. Ancestors like *person* or *living thing* indicate an animate noun. Ancestors like *female* indicate a female noun. A list of personal names associated with genders, or patterns like `Mr.` can also be used (Cardie and Wagstaff, 1999).

More complex algorithms exist, such as that of Bergsma and Lin (2006); Bergsma and Lin also make freely available a large list of nouns and their (automatically extracted) genders.

21.6.2 A Centering Algorithm for Anaphora Resolution

Centering theory

The Hobbs algorithm does not use an explicit representation of a discourse model. By contrast **Centering theory**, (Grosz et al., 1995b, henceforth GJW) is a family of models which has an explicit representation of a discourse model, and incorporates an additional claim: that there is a single entity being "centered" on at any given point in the discourse which is to be distinguished from all other entities that have been evoked. Centering theory has been applied to many problems in discourse, such as the computation of **entity-based coherence**; in this section we see its application to anaphora resolution.

Backward looking center

Forward looking center

There are two main representations tracked in the Centering theory discourse model. In what follows, take U_n and U_{n+1} to be two adjacent utterances. The **backward looking center** of U_n, denoted as $C_b(U_n)$, represents the entity currently being focused on in the discourse after U_n is interpreted. The **forward looking centers** of U_n, denoted as $C_f(U_n)$, form an ordered list containing the entities mentioned in U_n, all of which could serve as the C_b of the following utterance. In fact, $C_b(U_{n+1})$ is by definition the most highly ranked element of $C_f(U_n)$ mentioned in U_{n+1}. (The C_b of the first utterance in a discourse is undefined.) As for how the entities in the $C_f(U_n)$ are ordered, for simplicity's sake we can use the grammatical role hierarchy below.[3]

> subject $>$ existential predicate nominal $>$ object $>$ indirect object or oblique
> $>$ demarcated adverbial PP

As a shorthand, we will call the highest-ranked forward-looking center C_p (for "preferred center").

We describe a centering-based algorithm for pronoun interpretation due to Brennan et al. (1987, henceforth BFP). (See also Walker et al. (1994) and the end of the chapter for other centering algorithms). In this algorithm, preferred referents of pronouns are computed from relations that hold between the forward and backward looking centers in adjacent sentences. Four intersentential relationships between a pair of utterances U_n and U_{n+1} are defined which depend on the relationship between $C_b(U_{n+1})$, $C_b(U_n)$, and $C_p(U_{n+1})$; these are shown in Fig. 21.7.

	$C_b(U_{n+1}) = C_b(U_n)$ or undefined $C_b(U_n)$	$C_b(U_{n+1}) \neq C_b(U_n)$
$C_b(U_{n+1}) = C_p(U_{n+1})$	Continue	Smooth-Shift
$C_b(U_{n+1}) \neq C_p(U_{n+1})$	Retain	Rough-Shift

Figure 21.7 Transitions in the BFP algorithm.

The following rules are used by the algorithm:

[3] This is an extended form of the hierarchy used in Brennan et al. (1987), described below.

Rule 1: If any element of $C_f(U_n)$ is realized by a pronoun in utterance U_{n+1}, then $C_b(U_{n+1})$ must be realized as a pronoun also.
Rule 2: Transition states are ordered. Continue is preferred to Retain is preferred to Smooth-Shift is preferred to Rough-Shift.

Having defined these concepts and rules, the algorithm is defined as follows.

1. Generate possible C_b-C_f combinations for each possible set of reference assignments .
2. Filter by constraints, e.g., syntactic coreference constraints, selectional restrictions, centering rules and constraints.
3. Rank by transition orderings.

The pronominal referents that get assigned are those which yield the most preferred relation in Rule 2, assuming that Rule 1 and other coreference constraints (gender, number, syntactic, selectional restrictions) are not violated.

Let us step through passage (21.66) to illustrate the algorithm.

(21.66) John saw a beautiful 1961 Ford Falcon at the used car dealership. (U_1)
He showed it to Bob. (U_2)
He bought it. (U_3)

Using the grammatical role hierarchy to order the C$_f$, for sentence U_1 we get:

$C_f(U_1)$: {John, Ford, dealership}
$C_p(U_1)$: John
$C_b(U_1)$: undefined

Sentence U_2 contains two pronouns: *he*, which is compatible with John, and *it*, which is compatible with the Ford or the dealership. John is by definition $C_b(U_2)$, because he is the highest ranked member of $C_f(U_1)$ mentioned in U_2 (since he is the only possible referent for *he*). We compare the resulting transitions for each possible referent of *it*. If we assume *it* refers to the Falcon, the assignments would be:

$C_f(U_2)$: {John, Ford, Bob}
$C_p(U_2)$: John
$C_b(U_2)$: John
Result: Continue ($C_p(U_2)$=$C_b(U_2)$; $C_b(U_1)$ undefined)

If we assume *it* refers to the dealership, the assignments would be:

$C_f(U_2)$: {John, dealership, Bob}
$C_p(U_2)$: John
$C_b(U_2)$: John
Result: Continue ($C_p(U_2)$=$C_b(U_2)$; $C_b(U_1)$ undefined)

Since both possibilities result in a Continue transition, the algorithm does not say which to accept. For the sake of illustration, we will assume that ties are broken in terms of the ordering on the previous C_f list. Thus, we will take *it* to refer to the Falcon instead of the dealership, leaving the current discourse model as represented in the first possibility above.

In sentence U_3, *he* is compatible with either John or Bob, whereas *it* is compatible with the Ford. If we assume *he* refers to John, then John is $C_b(U_3)$ and the assignments would be:

$C_f(U_3)$: {John, Ford}
$C_p(U_3)$: John
$C_b(U_3)$: John
Result: Continue $(C_p(U_3){=}C_b(U_3){=}C_b(U_2))$

If we assume *he* refers to Bob, then Bob is $C_b(U_3)$ and the assignments would be:

$C_f(U_3)$: {Bob, Ford}
$C_p(U_3)$: Bob
$C_b(U_3)$: Bob
Result: Smooth-Shift $(C_p(U_3){=}C_b(U_3);\ C_b(U_3){\neq}C_b(U_2))$

Since a Continue is preferred to a Smooth-Shift per Rule 2, John is correctly taken to be the referent.

The main salience factors that the centering algorithm implicitly incorporates include the grammatical role, recency, and repeated mention preferences. The manner in which the grammatical role hierarchy affects salience is indirect, since it is the resulting transition type that determines the final reference assignments. In particular, a referent in a low-ranked grammatical role will be preferred to one in a more highly ranked role if the former leads to a more highly ranked transition. Thus, the centering algorithm may incorrectly resolve a pronoun to a low salience referent. For instance, in example (21.67),

(21.67) Bob opened up a new dealership last week. John took a look at the Fords in his lot. He ended up buying one.

the centering algorithm will assign Bob as the referent of the subject pronoun *he* in the third sentence – since Bob is $C_b(U_2)$, this assignment results in a Continue relation whereas assigning John results in a Smooth-Shift relation. On the other hand, the Hobbs algorithm will correctly assign John as the referent.

Like the Hobbs algorithm, the centering algorithm requires a full syntactic parse as well as morphological detectors for gender.

Centering theory is also a model of entity coherence, and hence has implications for other discourse applications like summarization; see the end of the chapter for pointers.

21.6.3 A Log-Linear model for Pronominal Anaphora Resoluton

As our final model of pronominal anaphora resolution, we present a simple supervised machine learning approach, in which we train a log-linear classifier on a corpus in which the antecedents are marked for each pronoun. Any supervised classifier can be used for this purpose; log-linear models are popular, but Naive Bayes and other classifiers have been used as well.

For training, the system relies on a hand-labeled corpus in which each pronoun has been linked by hand with the correct antecedent. The system needs to extract positive and negative examples of anaphoric relations. Positive examples occur directly in the training set. Negative examples can be found by pairing each pronoun with some other noun phrase. Features (discussed in the next section) are extracted for each training observation, and a classifier is trained to predict *1* for the true pronoun-antecedent pairs, and *0* for the incorrect pronoun-antecedent pairs.

For testing, just as we saw with as with the Hobbs and Centering classifiers, the log-linear classifier takes as input a pronoun (*he, him, his, she, her, it, they, them, their*), together with the current and preceding sentences.

In order to deal with non-referential pronouns, we first filter out pleonastic pronouns (like the pleonastic *it is raining*), using hand-written rules based on frequent lexical patterns.

The classifier then extracts all potential antecedents by doing a parse of the current and previous sentences, either using a full parser or a simple chunker. Next, each NP in the parse is considered a potential antecedent for each following pronoun. Each pronoun-potential antecedent pair is then presented to the classifier.

21.6.4 Features

Some commonly used features for pronominal anaphora resolution between a pronoun Pro_i and a potential referent NP_j include:

1. **strict gender [true or false]**. True if there is a strict match in gender (e.g. male pronoun Pro_i with male antecedent NP_j).
2. **compatible gender [true or false]**. True if Pro_i and NP_j are merely compatible (e.g. male pronoun Pro_i with antecedent NP_j of unknown gender).
3. **strict number [true or false]** True if there is a strict match in number (e.g. singular pronoun with singular antecedent)
4. **compatible number [true or false]**. True if Pro_i and NP_j are merely compatible (e.g. singular pronoun Pro_i with antecedent NP_j of unknown number).
5. **sentence distance [0, 1, 2, 3,...]**. The number of sentences between pronoun and potential antecedent.
6. **Hobbs distance [0, 1, 2, 3,...]**. The number of noun groups that the Hobbs algorithm has to skip, starting backwards from the pronoun Pro_i, before the potential antecedent NP_j is found.
7. **grammatical role [subject, object, PP]**. Whether the potential antecedent is a syntactic subject, direct object, or is embedded in a PP.
8. **linguistic form [proper, definite, indefinite, pronoun]**. Whether the potential antecedent NP_j is a proper name, definite description, indefinite NP, or a pronoun.

Fig. 21.8 shows some sample feature values for potential antecedents for the final *He* in U_3:

(21.68) John saw a beautiful 1961 Ford Falcon at the used car dealership. (U_1)
He showed it to Bob. (U_2)
He bought it. (U_3)

The classifier will learn weights indicating which of these features are more likely to be good predictors of a successful antecedent (e.g. being nearby the pronoun, in

	He (U_2)	it (U_2)	Bob (U_2)	John (U_1)
strict number	1	1	1	1
compatible number	1	1	1	1
strict gender	1	0	1	1
compatible gender	1	0	1	1
sentence distance	1	1	1	2
Hobbs distance	2	1	0	3
grammatical role	subject	object	PP	subject
linguistic form	pronoun	pronoun	proper	proper

Figure 21.8 Feature values in log-linear classifier, for various pronouns from (21.68).

subject position, agreeing in gender and number). Thus where the Hobbs and Centering algorithms rely on hand-built heuristics for antecedent selection, the machine learning classifiers learn the importance of these different features based on their co-occurrence in the training set.

21.7 Coreference Resolution

In the previous few sections, we concentrated on interpreting a particular subclass of the reference phenomena that we outlined in Sec. 21.4: the personal pronouns such as *he*, *she*, and *it*. But for the general coreference task we'll need to decide whether any pair of noun phrases corefer. This means we'll need to deal with the other types of referring expressions from Sec. 21.4, the most common of which are *definite noun phrases* and *names*. Let's return to our coreference example, repeated below:

(21.69) Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20%, to $1.3 million, as the 37-year-old also became the Denver-based financial-services company's president. It has been ten years since she came to Megabucks from rival Lotsabucks.

Recall that we need to extract four coreference chains from this data:

1. { *Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 1994, her, the 37-year-old, the Denver-based financial-services company's president, She*}
2. { *Megabucks Banking Corp, the Denver-based financial-services company, Megabucks* }
3. { *her pay* }
4. { *Lotsabucks* }

As before, we have to deal with pronominal anaphora (figuring out that *her* refers to *Victoria Chen*). And we still need to filter out non-referential pronouns like the pleonastic *It* in *It has been ten years*), as we did for pronominal anaphora.

But for full NP coreference we'll also need to deal with definite noun phrases, to figure out that *the 37-year-old* is coreferent with *Victoria Chen*, and *the Denver-based financial-services company* is the same as *Megabucks*. And we'll need to deal with names, to realize that *Megabucks* is the same as *Megabucks Banking Corp*.

An algorithm for coreference resolution can use the same log-linear classifier architecture we saw for pronominal anaphora. Thus we'll build a binary classifier which is given an anaphor and a potential antecedent and returns true (the two are coreferential) or false (the two are not coreferential). We'll use this classifier in the resolution algorithm as follows. We process a document from left to right. For each NP_j we encounter, we'll search backwards through the document examining each previous NP. For each such potential antecedent NP_i, we'll run our classifier, and if it returns true, we successfully coindex NP_i and NP_j. The process for each NP_j terminates when we either find a successful antecedent NP_i or reach the beginning of the document. We then move on to the next anaphor NP_j.

In order to train our binary coreference classifier, just as for pronoun resolution, we'll need a labeled training set in which each anaphor NP_i has been linked by hand with the correct antecedent. In order to build a classifier, we'll need both positive and negative training examples of coreference relations. A positive examples for NP_i is the noun phrase NP_j which is marked as coindexed. We get negative examples by pairing the anaphor NP_j with the intervening NPs NP_{i+1}, NP_{i+2} which occur between the true antecedent NP_i and the anaphor NP_j.

Next features are extracted for each training observation, and a classifier is trained to predict whether an (NP_j,NP_i) pair corefer or not. Which features should we use in the binary coreference classifier? We can use all the features we used for anaphora resolution; number, gender, syntactic position, and so on. But we will also need to add new features to deal with phenomena that are specific to names and definite noun phrases. For example, we'll want a feature representing the fact that *Megabucks* and *Megabucks Banking Corp* share the word *Megabucks*, or that *Megabucks Banking Corp* and *the Denver-based financial-services company* both end in words (*Corp* and *company*) indicating a corporate organization.

Here are some commonly used features for coreference between an anaphor NP_i and a potential antecedent NP_j (in addition to the features for pronominal anaphora resolution listed on page 721):

1. **anaphor edit distance [0,1,2,...,]**. The character **minimum edit distance** from the potential antecedent to the anaphor. Recall from Ch. 3 that the character minimum edit distance is the minimum number of character editing operations (insertions, substitutions, deletions) necessary to turn one string into another. More formally,
$$100 \times \frac{m-(s+i+d)}{m}$$
given the antecedent length m, and the number of substitutions s, insertions i, and deletions d.
2. **antecedent edit distance [0,1,2,...,]**. The **minimum edit distance** from the anaphor to the antecedent. Given the anaphor length n:
$$100 \times \frac{n-(s+i+d)}{n}$$
3. **alias [true or false]**: A multi-part feature proposed by Soon et al. (2001) which requires a **named entity tagger**. Returns true if NP_i and NP_j are both named entities of the same type, and NP_i is an **alias** of NP_j. The meaning of **alias** depends

on the types; two dates are aliases of each other if they refer to the same date. For type PERSON, prefixes like *Dr.* or *Chairman* are stripped off and then the NPs are checked to see if they are identical. For type ORGANIZATION, the alias function checks for acronyms (e.g., *IBM* for *International Business Machines Corp*).

4. **appositive [true** or **false]**: True if the anaphor is in the syntactic apposition relation to the antecedent. For example the NP *Chief Financial Officer of Megabucks Banking Corp* is in apposition to the NP *Victoria Chen*. These can be detected using a parser, or more shallowly by looking for commas and requiring that neither NP have a verb and one of them be a name.

5. **linguistic form [proper, definite, indefinite, pronoun]**. Whether the potential anaphor NP_j is a proper name, definite description, indefinite NP or a pronoun.

21.8 Evaluating Coreference Resolution

One standard way of evaluating coreference is the Model-Theoretic coreference scoring scheme (Vilain et al., 1995), originally proposed for the MUC-6 and MUC-7 information extraction evaluation (Sundheim, 1995a).

The evaluation is based on a human-labeled gold standard for coreference between referring expressions. We can represent this gold information as a set of identity links between referring expressions. For example, the fact that referring expression A and referring expression B are coreferent could be represented as a link A-B. If A, B, and C are coreferent, this can be represented as the two links A-B, B-C (or alternatively as A-C, B-C). We can call this set of correct links the **reference** or **key** set of links. Similarly, the **hypothesis** or **response** from a coreference algorithm can be viewed as a set of links.

What we'd like to do is compute the precision and recall of the **response** links against the **key** links. But recall that if entities A, B, and C are coreferent in the key, this can be represented either via (A-B, B-C) or via (A-C, B-C). As long as our coreference system correctly figures out that A, B, and C are coreferent, we don't want to penalize it for representing this fact in a different set of links than happen to be in the key.

For example, suppose that A, B, C, and D are coreferent, and this happens to be represented in the key by links (A-B, B-C, C-D). Suppose further that a particular coreference algorithm returns (A-B, C-D). What score should be given to this response? Intuitively the precision should be 1 (since both links correctly join referring expressions that indeed corefer). The recall should be 2/3, since intuitively it takes three links to correctly indicate that 4 expressions are coreferent, and the algorithm returned two of these three links. The details of this intuition are fleshed out in the Vilain et al. (1995) algorithm, which is based on computing the number of equivalence classes of expressions generated by the key.

21.9 Advanced: Inference-Based Coherence Resolution

The algorithms we have seen in this chapter for the resolution of coherence and coreference have relied solely on shallow information like cue phrases and other lexical and simple syntactic cues. But many problems in resolution seem to require much more sophisticated kinds of knowledge. Consider the following example of coreference, adapted from Winograd (1972b):

(21.70) The city council denied the demonstrators a permit because

a. they feared violence.

b. they advocated violence.

Determining the correct antecedent for the pronoun *they* requires understanding first that the second clause is intended as an **Explanation** of the first clause, and also that city councils are perhaps more likely than demonstrators to fear violence, and demonstrators might be more likely to advocate violence. A more advanced method for coherence resolution might assign this Explanation relation and in doing so help us figure out the referents of both pronouns.

We might perform this kind of more sophisticated coherence resolution by relying on the semantic constraints that are associated with each coherence relation, assuming a parser that could assign a reasonable semantics to each clause.

Deduction

Applying these constraints requires a method for performing inference. Perhaps the most familiar type of inference is **deduction**; recall from Sec. 17.3 that the central rule of deduction is modus ponens:

$$\frac{\begin{array}{c}\alpha \Rightarrow \beta \\ \alpha\end{array}}{\beta}$$

An example of modus ponens is the following:

$$\frac{\begin{array}{c}\text{All Falcons are fast.} \\ \text{John's car is an Falcon.}\end{array}}{\text{John's car is fast.}}$$

Sound inference

Deduction is a form of **sound inference**: if the premises are true, then the conclusion must be true.

Abduction

However, much of language understanding is based on inferences that are not sound. While the ability to draw unsound inferences allows for a greater range of inferences to be made, it can also lead to false interpretations and misunderstandings. A method for such inference is logical **abduction** (Peirce, 1955). The central rule of abductive inference is:

$$\frac{\begin{array}{c}\alpha \Rightarrow \beta \\ \beta\end{array}}{\alpha}$$

Whereas deduction runs an implication relation forward, abduction runs it backward, reasoning from an effect to a potential cause. An example of abduction is the following:

$$\begin{array}{c}\text{All Falcons are fast.}\\ \text{John's car is fast.}\\ \hline \text{John's car is an Falcon.}\end{array}$$

Obviously, this may be an incorrect inference: John's car may be made by another manufacturer yet still be fast.

In general, a given effect β may have many potential causes α_i. We generally will not want to merely reason from a fact to a *possible* explanation of it, we want to identify the *best* explanation of it. To do this, we need a method for comparing the quality of alternative abductive proofs. This can be done with probabilistic models (Charniak and Goldman, 1988; Charniak and Shimony, 1990), or with heuristic strategies (Charniak and McDermott, 1985, Chapter 10), such as preferring the explanation with the smallest number of assumptions, or the most specific explanation. We will illustrate a third approach to abductive interpretation, due to Hobbs et al. (1993), which applies a more general cost-based strategy that combines features of the probabilistic and heuristic approaches. To simplify the discussion, however, we will largely ignore the cost component of the system, keeping in mind that one is nonetheless necessary.

Hobbs et al. (1993) apply their method to a broad range of problems in language interpretation; here we focus on its use in establishing discourse coherence, in which world and domain knowledge are used to determine the most plausible coherence relation holding between utterances. Let us step through the analysis that leads to establishing the coherence of passage (21.4). First, we need axioms about coherence relations themselves. Axiom (21.71) states that a possible coherence relation is the Explanation relation; other relations would have analogous axioms.

(21.71) $$\forall e_i, e_j \; Explanation(e_i, e_j) \Rightarrow CoherenceRel(e_i, e_j)$$

The variables e_i and e_j represent the events (or states) denoted by the two utterances being related. In this axiom and those given below, quantifiers always scope over everything to their right. This axiom tells us that, given that we need to establish a coherence relation between two events, one possibility is to abductively assume that the relation is Explanation.

The Explanation relation requires that the second utterance express the cause of the effect that the first sentence expresses. We can state this as axiom (21.72).

(21.72) $$\forall e_i, e_j \; cause(e_j, e_i) \Rightarrow Explanation(e_i, e_j)$$

In addition to axioms about coherence relations, we also need axioms representing general knowledge about the world. The first axiom we use says that if someone is drunk, then others will not want that person to drive, and that the former causes the latter (for convenience, the state of not wanting is denoted by the *diswant* predicate).

(21.73) $$\begin{aligned}&\forall x, y, e_i \; drunk(e_i, x) \Rightarrow\\ &\quad \exists e_j, e_k \; diswant(e_j, y, e_k) \wedge drive(e_k, x) \wedge cause(e_i, e_j)\end{aligned}$$

Before we move on, a few notes are in order concerning this axiom and the others we will present. First, axiom (21.73) is stated using universal quantifiers to bind several of the variables, which essentially says that in all cases in which someone is drunk, all people do not want that person to drive. Although we might hope that this is generally the case, such a statement is nonetheless too strong. The way in which this is handled in the Hobbs et al. system is by including an additional relation, called an *etc* predicate, in the antecedent of such axioms. An *etc* predicate represents all the other properties that must be true for the axiom to apply, but which are too vague to state explicitly. These predicates therefore cannot be proven, they can only be assumed at a corresponding cost. Because rules with high assumption costs will be dispreferred to ones with low costs, the likelihood that the rule applies can be encoded in terms of this cost. Since we have chosen to simplify our discussion by ignoring costs, we will similarly ignore the use of *etc* predicates.

Second, each predicate has what may look like an “extra” variable in the first argument position; for instance, the *drive* predicate has two arguments instead of one. This variable is used to reify the relationship denoted by the predicate so that it can be referred to from argument places in other predicates. For instance, reifying the *drive* predicate with the variable e_k allows us to express the idea of not wanting someone to drive by referring to it in the final argument of the *diswant* predicate.

Picking up where we left off, the second world knowledge axiom we use says that if someone does not want someone else to drive, then they do not want this person to have his car keys, since car keys enable someone to drive.

(21.74)
$$\begin{aligned}&\forall x,y,e_j,e_k\ diswant(e_j,y,e_k)\wedge drive(e_k,x)\ \Rightarrow\\ &\quad \exists z,e_l,e_m\ diswant(e_l,y,e_m)\wedge have(e_m,x,z)\\ &\qquad \wedge carkeys(z,x)\wedge cause(e_j,e_l)\end{aligned}$$

The third axiom says that if someone doesn’t want someone else to have something, he might hide it from him.

(21.75)
$$\begin{aligned}&\forall x,y,z,e_l,e_m\ diswant(e_l,y,e_m)\wedge have(e_m,x,z)\ \Rightarrow\\ &\quad \exists e_n\ hide(e_n,y,x,z)\wedge cause(e_l,e_n)\end{aligned}$$

The final axiom says simply that causality is transitive, that is, if e_i causes e_j and e_j causes e_k, then e_i causes e_k.

(21.76)
$$\forall e_i,e_j,e_k\ cause(e_i,e_j)\wedge cause(e_j,e_k)\ \Rightarrow\ cause(e_i,e_k)$$

Finally, we have the content of the utterances themselves, that is, that John hid Bill’s car keys (from Bill),

(21.77) $hide(e_1,John,Bill,ck)\wedge carkeys(ck,Bill)$

and that someone described using the pronoun “he” was drunk; we will represent the pronoun with the free variable *he*.

(21.78) $drunk(e_2,he)$

We can now see how reasoning with the content of the utterances along with the aforementioned axioms allows the coherence of passage (21.4) to be established under the Explanation relation. The derivation is summarized in Fig. 21.9; the sentence interpretations are shown in boxes. We start by assuming there is a coherence relation, and using axiom (21.71) hypothesize that this relation is Explanation,

(21.79) $Explanation(e_1, e_2)$

which, by axiom (21.72), means we hypothesize that

(21.80) $cause(e_2, e_1)$

holds. By axiom (21.76), we can hypothesize that there is an intermediate cause e_3,

(21.81) $cause(e_2, e_3) \wedge cause(e_3, e_1)$

and we can repeat this again by expanding the first conjunct of (21.81) to have an intermediate cause e_4.

(21.82) $cause(e_2, e_4) \wedge cause(e_4, e_3)$

We can take the *hide* predicate from the interpretation of the first sentence in (21.77) and the second *cause* predicate in (21.81), and, using axiom (21.75), hypothesize that John did not want Bill to have his car keys:

(21.83) $diswant(e_3, John, e_5) \wedge have(e_5, Bill, ck)$

From this, the *carkeys* predicate from (21.77), and the second *cause* predicate from (21.82), we can use axiom (21.74) to hypothesize that John does not want Bill to drive:

(21.84) $diswant(e_4, John, e_6) \wedge drive(e_6, Bill)$

From this, axiom (21.73), and the second *cause* predicate from (21.82), we can hypothesize that Bill was drunk:

(21.85) $drunk(e_2, Bill)$

But now we find that we can “prove” this fact from the interpretation of the second sentence if we simply assume that the free variable *he* is bound to Bill. Thus, the establishment of coherence has gone through, as we have identified a chain of reasoning between the sentence interpretations – one that includes unprovable assumptions about axiom choice and pronoun assignment – that results in $cause(e_2, e_1)$, as required for establishing the Explanation relationship.

This derivation illustrates a powerful property of coherence establishment, namely its ability to cause the hearer to infer information about the situation described by the discourse that the speaker has left unsaid. In this case, the derivation required the assumption that John hid Bill’s keys because he did not want him to drive (presumably out of fear of him having an accident, or getting stopped by the police), as opposed to some other explanation, such as playing a practical joke on him. This cause is not stated anywhere in passage (21.4); it arises only from the inference process triggered by the need to establish coherence. In this sense, the meaning of a discourse is greater than the sum of the meanings of its parts. That is, a discourse typically communicates far more information than is contained in the interpretations of the individual sentences that comprise it.

We now return to passage (21.5), repeated below as (21.87), which was notable in that it lacks the coherence displayed by passage (21.4), repeated below as (21.86).

(21.86) John hid Bill’s car keys. He was drunk.

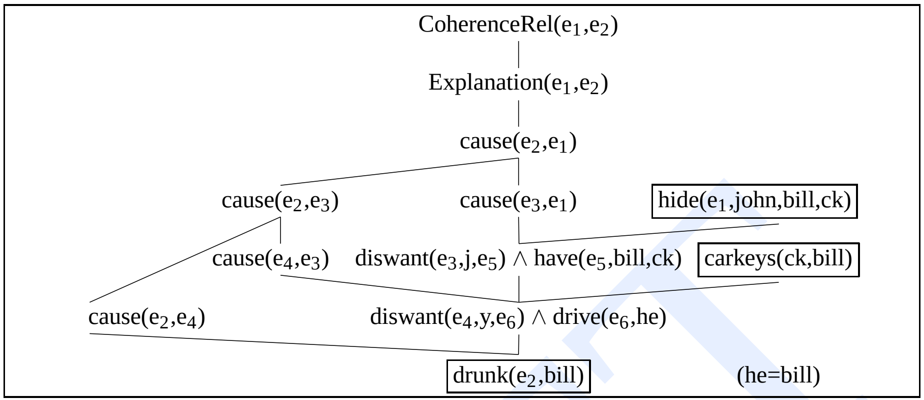


Figure 21.9 Establishing the coherence of passage (21.4).

(21.87) ?? John hid Bill's car keys. He likes spinach.

We can now see why this is: there is no analogous chain of inference capable of linking the two utterance representations, in particular, there is no causal axiom analogous to (21.73) that says that liking spinach might cause someone to not want you to drive. Without additional information that can support such a chain of inference (such as the aforementioned scenario in which someone promised John spinach in exchange for hiding Bill's car keys), the coherence of the passage cannot be established.

Because abduction is a form of unsound inference, it must be possible to subsequently retract the assumptions made during abductive reasoning, that is, abductive inferences are **defeasible**. For instance, if passage (21.86) was followed by sentence (21.88),

Defeasible

(21.88) Bill's car isn't here anyway; John was just playing a practical joke on him.

the system would have to retract the original chain of inference connecting the two clauses in (21.86), and replace it with one utilizing the fact that the hiding event was part of a practical joke.

In a more general knowledge base designed to support a broad range of inferences, one would want axioms that are more general than those we used to establish the coherence of passage (21.86). For instance, consider axiom (21.74), which says that if you do not want someone to drive, then you do not want them to have their car keys. A more general form of the axiom would say that if you do not want someone to perform an action, and an object enables them to perform that action, then you do not want them to have the object. The fact that car keys enable someone to drive would then be encoded separately, along with many other similar facts. Likewise, axiom (21.73) says that if someone is drunk, you don't want them to drive. We might replace this with an axiom that says that if someone does not want something to happen, then they don't want something that will likely cause it to happen. Again, the facts that people typically don't want other people to get into car accidents, and that drunk driving causes accidents, would be encoded separately.

While it is important to have computational models that shed light on the coherence establishment problem, large barriers remain for employing this and similar methods on a wide-coverage basis. In particular, the large number of axioms that would be required to encode all of the necessary facts about the world, and the lack of a robust mechanism for constraining inference with such a large set of axioms, makes these methods largely impractical in practice. Nonetheless, approximations to these kinds of knowledge and inferential rules can already play an important role in natural language understanding systems.

21.10 Psycholinguistic Studies of Reference and Coherence

To what extent do the techniques described in this chapter model human discourse comprehension? We summarize here a few selected results from the substantial body of psycholinguistic research; for reasons of space we focus here solely on anaphora resolution.

reading time experiments

A significant amount of work has been concerned with the extent to which people use the preferences described in Section 21.5 to interpret pronouns, the results of which are often contradictory. Clark and Sengal (1979) studied the effects that sentence recency plays in pronoun interpretation using a set of **reading time experiments**. After receiving and acknowledging a three sentence context to read, human subjects were given a target sentence containing a pronoun. The subjects pressed a button when they felt that they understood the target sentence. Clark and Sengal found that the reading time was significantly faster when the referent for the pronoun was evoked from the most recent clause in the context than when it was evoked from two or three clauses back. On the other hand, there was no significant difference between referents evoked from two clauses and three clauses back, leading them to claim that “the last clause processed grants the entities it mentions a privileged place in working memory”.

question answering
Referent naming task

Crawley et al. (1990) compared the grammatical role parallelism preference with a grammatical role preference, in particular, a preference for referents evoked from the subject position of the previous sentence over those evoked from object position. Unlike previous studies which conflated these preferences by considering only subject-to-subject reference effects, Crawley et al. studied pronouns in object position to see if they tended to be assigned to the subject or object of the last sentence. They found that in two task environments – a **question answering task** which revealed how the human subjects interpreted the pronoun, and a **referent naming task** in which the subjects identified the referent of the pronoun directly – the human subjects resolved pronouns to the subject of the previous sentence more often than the object.

However, Smyth (1994) criticized the adequacy of Crawley et al.’s data for evaluating the role of parallelism. Using data that met more stringent requirements for assessing parallelism, Smyth found that subjects overwhelmingly followed the parallelism preference in a referent naming task. The experiment supplied weaker support for the preference for subject referents over object referents, which he posited as a default strategy when the sentences in question are not sufficiently parallel.

Sentence completion task

Caramazza et al. (1977) studied the effect of the “implicit causality” of verbs on pronoun resolution. Verbs were categorized in terms of having subject bias or object bias using a **sentence completion task**. Subjects were given sentence fragments such as (21.89).

(21.89) John telephoned Bill because he

The subjects provided completions to the sentences, which identified to the experimenters what referent for the pronoun they favored. Verbs for which a large percentage of human subjects indicated a grammatical subject or object preference were categorized as having that bias. A sentence pair was then constructed for each biased verb: a "congruent" sentence in which the semantics supported the pronoun assignment suggested by the verb's bias, and an "incongruent" sentence in which the semantics supported the opposite prediction. For example, sentence (21.90) is congruent for the subject-bias verb "telephoned", since the semantics of the second clause supports assigning the subject *John* as the antecedent of *he*, whereas sentence (21.91) is incongruent since the semantics supports assigning the object *Bill*.

(21.90) John telephoned Bill because he wanted some information.

(21.91) John telephoned Bill because he withheld some information.

In a referent naming task, Caramazza et al. found that naming times were faster for the congruent sentences than for the incongruent ones. Perhaps surprisingly, this was even true for cases in which the two people mentioned in the first clause were of different genders, thus rendering the reference unambiguous.

Matthews and Chodorow (1988) analyzed the problem of intrasentential reference and the predictions of syntactically-based search strategies. In a question answering task, they found that subjects exhibited slower comprehension times for sentences in which a pronoun antecedent occupied an early, syntactically deep position than for sentences in which the antecedent occupied a late, syntactically shallow position. This result is consistent with the search process used in Hobbs's tree search algorithm.

There has also been psycholinguistic work concerned with testing the principles of centering theory. In a set of reading time experiments, Gordon et al. (1993) found that reading times were slower when the current backward-looking center was referred to using a full noun phrase instead of a pronoun, even though the pronouns were ambiguous and the proper names were not. This effect – which they called a **repeated name penalty** – was found only for referents in subject position, suggesting that the C_b is preferentially realized as a subject. Brennan (1995) analyzed how choice of linguistic form correlates with centering principles. She ran a set of experiments in which a human subject watched a basketball game and had to describe it to a second person. She found that the human subjects tended to refer to an entity using a full noun phrase in subject position before subsequently pronominalizing it, even if the referent had already been introduced in object position.

repeated name penalty

21.11 Summary

In this chapter, we saw that many of the problems that natural language processing systems face operate between sentences, that is, at the *discourse* level. Here is a summary of some of the main points we discussed:

- Discourses, like sentences, have hierarchical structure. In the simplest kind of structure detection, we assume a simpler linear structure, and segment a discourse on topic or other boundaries. The main cues for this are **lexical cohesion** as well as discourse markers/cue phrases.

- Discourses are not arbitrary collections of sentences; they must be *coherent*. Among the factors that make a discourse coherent are coherence relations between the sentences and entity-based coherence.
- Various sets of **coherence relations** and rhetorical relations have been proposed. Algorithms for detecting these coherence relations can use surface-based cues (cue phrases, syntactic information).
- Discourse interpretation requires that one build an evolving representation of discourse state, called a *discourse model*, that contains representations of the entities that have been referred to and the relationships in which they participate.
- Natural languages offer many ways to refer to entities. Each form of reference sends its own signals to the hearer about how it should be processed with respect to her discourse model and set of beliefs about the world.
- Pronominal reference can be used for referents that have an adequate degree of *salience* in the discourse model. There are a variety of lexical, syntactic, semantic, and discourse factors that appear to affect salience.
- The Hobbs, Centering, and Log-linear models for pronominal anaphora offer different ways of drawing on and combining various of these constraints.
- The full NP coreference task also has to deal with names and definite NPs. String edit distance is a useful features for these.
- Advanced algorithms for establishing coherence apply constraints imposed by one or more coherence relations, often leads to the inference of additional information left unsaid by the speaker. The unsound rule of logical *abduction* can be used for performing such inference.

Bibliographical and Historical Notes

Building on the foundations set by early systems for natural language understanding (Woods et al., 1972a; Winograd, 1972b; Woods, 1978), much of the fundamental work in computational approaches to discourse was performed in the late 70's. Webber's (1978, 1983) work provided fundamental insights into how entities are represented in the discourse model and the ways in which they can license subsequent reference. Many of the examples she provided continue to challenge theories of reference to this day. Grosz (1977a) addressed the focus of attention that conversational participants maintain as the discourse unfolds. She defined two levels of focus; entities relevant to the entire discourse were said to be in *global* focus, whereas entities that are locally in focus (i.e., most central to a particular utterance) were said to be in *immediate* focus. Sidner (1979, 1983) described a method for tracking (immediate) discourse foci and their use in resolving pronouns and demonstrative noun phrases. She made a distinction between the current discourse focus and potential foci, which are the predecessors to the backward and forward looking centers of centering theory respectively.

The roots of the centering approach originate from papers by Joshi and Kuhn (1979) and Joshi and Weinstein (1981), who addressed the relationship between immediate focus and the inferences required to integrate the current utterance into the discourse model. Grosz et al. (1983) integrated this work with the prior work of Sidner and Grosz. This led to a manuscript on centering which, while widely circulated since 1986, remained unpublished until Grosz et al. (1995b). A series of papers on centering based on this manuscript/paper were subsequently published (Kameyama, 1986; Brennan et al., 1987; Di Eugenio, 1990; Walker et al., 1994; Di Eugenio, 1996; Strube and Hahn, 1996; Kehler, 1997a, inter alia). A collection of later centering papers appears in Walker et al. (1998), and see Poesio et al. (2004) for more recent work. We have focused in this chapter on Centering and anaphora resolution; Sse Karamanis (2003, 2007), Barzilay and Lapata (2007) and related papers discussed in Ch. 23 for the application of Centering to entity-based coherence.

There is a long history in linguistics of studies of *information status* (Chafe, 1976; Prince, 1981; Ariel, 1990; Prince, 1992; Gundel et al., 1993; Lambrecht, 1994, inter alia).

Beginning with Hobbs's (1978) tree-search algorithm, researchers have pursued syntax-based methods for identifying reference robustly in naturally occurring text. An early system for a weighted combination of different syntactic and other features was Lappin and Leass (1994), which we described in detail in our first edition. Kennedy and Boguraev (1996) describe a similar system that does not rely on a full syntactic parser, but merely a mechanism for identifying noun phrases and labeling their grammatical roles. Both approaches use Alshawi's (1987) framework for integrating salience factors. An algorithm that uses this framework for resolving references in a multimodal (i.e., speech and gesture) human-computer interface is described in Huls et al. (1995). A discussion of a variety of approaches to reference in operational systems can be found in Mitkov and Boguraev (1997).

Methods for reference resolution based on supervised learning were proposed quite early (Connolly et al., 1994; Aone and Bennett, 1995; McCarthy and Lehnert, 1995; Kehler, 1997b; Ge et al., 1998, inter alia). More recently both supervised and unsupervised approaches have received a lot of research attention, focused both on anaphora resolution Kehler et al. (2004), Bergsma and Lin (2006) and full NP coreference (Cardie and Wagstaff, 1999; Ng and Cardie, 2002b; Ng, 2005). For definite NP reference, there are general algorithms (Poesio and Vieira, 1998; Vieira and Poesio, 2000), as well as specific algorithms that focus on deciding if a particular definite NP is anaphoric or not (Bean and Riloff, 1999, 2004; Ng and Cardie, 2002a; Ng, 2004).

See Mitkov (2002) for an excellent comprehensive overview of anaphora resolution and Branco et al. (2002) for a collection of papers.

The idea of using cohesion for linear discourse segmentation was implicit in the groundbreaking work of (Halliday and Hasan, 1976), but was first explicitly implemented by Morris and Hirst (1991), and quickly picked up by many other researchers, including (Kozima, 1993; Reynar, 1994; Hearst, 1994, 1997; Reynar, 1999; Kan et al., 1998; Choi, 2000; Choi et al., 2001; Brants et al., 2002; Bestgen, 2006). Power et al. (2003) studies discourse structure, while Filippova and Strube (2006), Sporleder and Lapata (2004, 2006) focus on paragraph segmentation.

The use of cue phrases in segmentation has been widely studied, including work on many textual genres as well as speech (Passonneau and Litman, 1993; Hirschberg and Litman, 1993; Manning, 1998; Kawahara et al., 2004)

Many researchers have posited sets of coherence relations that can hold between utterances in a discourse (Halliday and Hasan, 1976; Hobbs, 1979a; Longacre, 1983; Mann and Thompson, 1987; Polanyi, 1988; Hobbs, 1990; Sanders et al., 1992; Carlson et al., 2001, 2002; Asher and Lascarides, 2003; Baldridge et al., 2007, inter alia). A compendium of over 350 relations that have been proposed in the literature can be found in Hovy (1990).

There are a wide variety of approaches to coherence extraction. The cue-phrase based model described in Sec. 21.2.2 is due to Daniel Marcu and colleagues (Marcu, 2000b, 2000a; Carlson et al., 2001, 2002). The Linguistic Discourse Model (Polanyi, 1988; Scha and Polanyi, 1988; Polanyi et al., 2004a, 2004b) is a framework in which discourse syntax is more heavily emphasized; in this approach, a discourse parse tree is built on a clause-by-clause basis in direct analogy with how a sentence parse tree is built on a constituent-by-constituent basis. Corston-Oliver (1998) also explores syntactic and parser-based features. A more recent line of work has applied a version of the tree-adjoining grammar formalism to discourse parsing (Webber et al., 1999; Webber, 2004). This model has also been used to annotate the Penn Discourse Treebank (Miltsakaki et al., 2004b, 2004a). See Asher and Lascarides (2003) and Baldridge et al. (2007) on Segmented Discourse Representation Structure (**SDRT**). Wolf and Gibson (2005) argue that coherence structure includes crossed bracketings which make it impossible to represent as a tree, and propose a graph representation instead.

SDRT

In addition to determining discourse structure and meaning, theories of discourse coherence have been used in algorithms for interpreting discourse-level linguistic phenomena, including pronoun resolution (Hobbs, 1979a; Kehler, 2000), verb phrase ellipsis and gapping (Prüst, 1992; Asher, 1993; Kehler, 1993, 1994a), and tense interpretation (Lascarides and Asher, 1993; Kehler, 1994b, 2000). An extensive investigation into the relationship between coherence relations and discourse connectives can be found in Knott and Dale (1994).

Exercises

21.1 Early work in syntactic theory attempted to characterize rules for pronominalization through purely syntactic means. A rule was proposed in which a pronoun was interpreted by deleting it from the syntactic structure of the sentence that contains it, and replacing it with the syntactic representation of the antecedent noun phrase.

Explain why the following sentences (called "Bach-Peters" sentences) are problematic for such an analysis:

(21.92) The man who deserves it gets the prize he wants.
(21.93) The pilot who shot at it hit the MIG that chased him.

What other types of reference discussed on pages 710–713 are problematic for this type of analysis?

21.2 Draw syntactic trees for example (21.66) on page 719 and apply Hobbs's tree search algorithm to it, showing each step in the search.

21.3 Hobbs (1977) cites the following examples from his corpus as being problematic for his tree-search algorithm:

(21.94) The positions of pillars in one hall were marked by river boulders and a shaped convex cushion of bronze that had served as their footings.

(21.95) They were at once assigned an important place among the scanty remains which record the physical developments of the human race from the time of its first appearance in Asia.

(21.96) Sites at which the coarse grey pottery of the Shang period has been discovered do not extend far beyond the southernmost reach of the Yellow river, or westward beyond its junction with the Wei.

(21.97) The thin, hard, black-burnished pottery, made in shapes of angular profile, which archaeologists consider as the clearest hallmark of the Lung Shan culture, developed in the east. The site from which it takes its name is in Shantung. It is traced to the north-east as far as Liao-ning province.

(21.98) He had the duty of performing the national sacrifices to heaven and earth: his role as source of honours and material rewards for services rendered by feudal lords and ministers is commemorated in thousands of inscriptions made by the recipients on bronze vessels which were eventually deposited in their graves.

In each case, identify the correct referent of the underlined pronoun and the one that the algorithm will identify incorrectly. Discuss any factors that come into play in determining the correct referent in each case, and what types of information might be necessary to account for them.

21.4 Implement the Hobbs algorithm. Test it on a sample of the Penn TreeBank. You will need to modify the algorithm to deal with differences between the Hobbs and TreeBank grammars.

21.5 Consider the following passage, from Brennan et al. (1987):

(21.99) Brennan drives an Alfa Romeo.
She drives too fast.
Friedman races her on weekends.
She goes to Laguna Seca.

Identify the referent that the BFP algorithm finds for the pronoun in the final sentence. Do you agree with this choice, or do you find the example ambiguous? Discuss why introducing a new noun phrase in subject position, with a pronominalized reference in object position, might lead to an ambiguity for a subject pronoun in the next sentence. What preferences are competing here?

21.6 Consider passages (21.100a-b), adapted from Winograd (1972b).

(21.100) The city council denied the demonstrators a permit because
a. they feared violence.

b. they advocated violence.

What are the correct interpretations for the pronouns in each case? Sketch out an analysis of each in the interpretation as abduction framework, in which these reference assignments are made as a by-product of establishing the Explanation relation.

21.7 Select an editorial column from your favorite newspaper, and determine the discourse structure for a 10-20 sentence portion. What problems did you encounter? Were you helped by superficial cues the speaker included (e.g., discourse connectives) in any places?