Chapter 16

Discourse

Applications of natural language processing often concern multi-sentence documents: from paragraph-long restaurant reviews, to 500-word newspaper articles, to 500-page novels. Yet most of the methods that we have discussed thus far are concerned with individual sentences. This chapter discusses theories and methods for handling multi-sentence linguistic phenomena, known collectively as **discourse**. There are diverse characterizations of discourse structure, and no single structure is ideal for every computational application. This chapter covers some of the most well studied discourse representations, while highlighting computational models for identifying and exploiting these structures.

16.1 Segments

A document or conversation can be viewed as a sequence of **segments**, each of which is **cohesive** in its content and/or function. In Wikipedia biographies, these segments often pertain to various aspects to the subject's life: early years, major events, impact on others, and so on. This segmentation is organized around **topics**. Alternatively, scientific research articles are often organized by **functional themes**: the introduction, a survey of previous research, experimental setup, and results.

Written texts often mark segments with section headers and related formatting devices. However, such formatting may be too coarse-grained to support applications such as the retrieval of specific passages of text that are relevant to a query (Hearst, 1997). Unformatted speech transcripts, such as meetings and lectures, are also an application scenario for segmentation (Carletta, 2007; Glass et al., 2007; Janin et al., 2003).

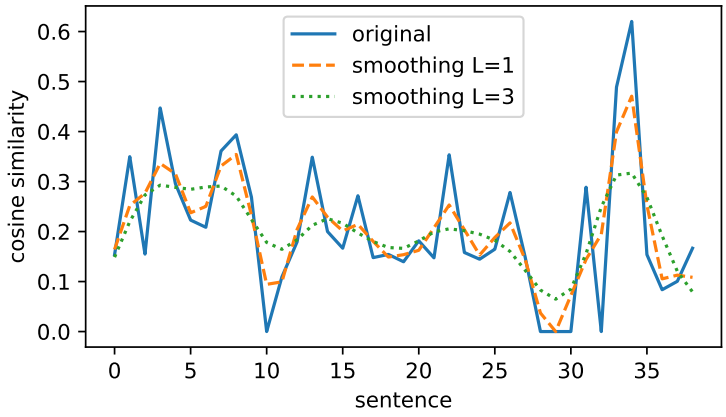


Figure 16.1: Smoothed cosine similarity among adjacent sentences in a news article. Local minima at $m = 10$ and $m = 29$ indicate likely segmentation points.

16.1.1 Topic segmentation

A cohesive topic segment forms a unified whole, using various linguistic devices: repeated references to an entity or event; the use of conjunctions to link related ideas; and the repetition of meaning through lexical choices (Halliday and Hasan, 1976). Each of these cohesive devices can be measured, and then used as features for topic segmentation. A classical example is the use of lexical cohesion in the TEXTTILING method for topic segmentation (Hearst, 1997). The basic idea is to compute the textual similarity between each pair of adjacent blocks of text (sentences or fixed-length units), using a formula such as the smoothed **cosine similarity** of their bag-of-words vectors,

$$s_m = \frac{\boldsymbol{x}_m \cdot \boldsymbol{x}_{m+1}}{||\boldsymbol{x}_m||_2 \times ||\boldsymbol{x}_{m+1}||_2} \quad [16.1]$$

$$\bar{s}_m = \sum_{\ell=0}^{L} k_\ell (s_{m+\ell} + s_{m-\ell}), \quad [16.2]$$

with k_ℓ representing the value of a smoothing kernel of size L, e.g. $\boldsymbol{k} = [1, 0.5, 0.25]^\top$. Segmentation points are then identified at local minima in the smoothed similarities $\bar{\boldsymbol{s}}$, since these points indicate changes in the overall distribution of words in the text. An example is shown in Figure 16.1.

Text segmentation can also be formulated as a probabilistic model, in which each segment has a unique language model that defines the probability over the text in the segment (Utiyama and Isahara, 2001; Eisenstein and Barzilay, 2008; Du et al., 2013).[1] A good

[1]There is a rich literature on how latent variable models (such as **latent Dirichlet allocation**) can track

segmentation achieves high likelihood by grouping segments with similar word distributions. This probabilistic approach can be extended to **hierarchical topic segmentation**, in which each topic segment is divided into subsegments (Eisenstein, 2009). All of these approaches are unsupervised. While labeled data can be obtained from well-formatted texts such as textbooks, such annotations may not generalize to speech transcripts in alternative domains. Supervised methods have been tried in cases where in-domain labeled data is available, substantially improving performance by learning weights on multiple types of features (Galley et al., 2003).

16.1.2 Functional segmentation

In some genres, there is a canonical set of communicative *functions*: for example, in scientific research articles, one such function is to communicate the general background for the article, another is to introduce a new contribution, or to describe the aim of the research (Teufel et al., 1999). A **functional segmentation** divides the document into contiguous segments, sometimes called **rhetorical zones**, in which each sentence has the same function. Teufel and Moens (2002) train a supervised classifier to identify the functional of each sentence in a set of scientific research articles, using features that describe the sentence's position in the text, its similarity to the rest of the article and title, tense and voice of the main verb, and the functional role of the previous sentence. Functional segmentation can also be performed without supervision. Noting that some types of Wikipedia articles have very consistent functional segmentations (e.g., articles about cities or chemical elements), Chen et al. (2009) introduce an unsupervised model for functional segmentation, which learns both the language model associated with each function and the typical patterning of functional segments across the article.

16.2 Entities and reference

Another dimension of discourse relates to which entities are mentioned throughout the text, and how. Consider the examples in Figure 16.2: Grosz et al. (1995) argue that the first discourse is more coherent. Do you agree? The examples differ in their choice of **referring expressions** for the protagonist *John*, and in the syntactic constructions in sentences (b) and (d). The examples demonstrate the need for theoretical models to explain how referring expressions are chosen, and where they are placed within sentences. Such models can then be used to help interpret the overall structure of the discourse, to measure discourse coherence, and to generate discourses in which referring expressions are used coherently.

topics across documents (Blei et al., 2003; Blei, 2012).

(16.1)
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

(16.2)
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.

Figure 16.2: Two tellings of the same story (Grosz et al., 1995). The discourse on the left uses referring expressions coherently, while the one on the right does not.

16.2.1 Centering theory

Centering theory presents a unified account of the relationship between discourse structure and entity reference (Grosz et al., 1995). According to the theory, every utterance in the discourse is characterized by a set of entities, known as *centers*.

- The **forward-looking centers** in utterance m are all the entities that are mentioned in the utterance, $\boldsymbol{c}_f(\boldsymbol{w}_m) = \{e_1, e_2, \ldots, \}$. The forward-looking centers are partially ordered by their syntactic prominence, favoring subjects over objects, and objects over other positions (Brennan et al., 1987). For example, in example (1.1a) of Figure 16.2, the ordered list of forward-looking centers in the first utterance is John, the music store, and the piano.
- The **backward-looking center** $c_b(\boldsymbol{w}_m)$ is the highest-ranked element in the set of forward-looking centers from the previous utterance $\boldsymbol{c}_f(\boldsymbol{w}_{m-1})$ that is also mentioned in $\boldsymbol{w}_m$. In example (1.1b) of item 16.1, the backward looking center is John.

Given these two definitions, centering theory makes the following predictions about the form and position of referring expressions:

1. If a pronoun appears in the utterance $\boldsymbol{w}_m$, then the backward-looking center $c_b(\boldsymbol{w}_m)$ must also be realized as a pronoun. This rule argues against the use of *it* to refer to the piano store in Example (16.2d), since JOHN is the backward looking center of (16.2d), and he is mentioned by name and not by a pronoun.

2. Sequences of utterances should retain the same backward-looking center if possible, and ideally, the backward-looking center should also be the top-ranked element in the list of forward-looking centers. This rule argues in favor of the preservation of JOHN as the backward-looking center throughout Example (16.1).

	SKYLER	WALTER	DANGER	A GUY	THE DOOR
You don't know who you're talking to,	S	-	-	-	-
so let me clue you in.	O	O	-	-	-
I am not in danger, Skyler.	X	S	X	-	-
I am the danger.	-	S	O	-	-
A guy opens his door and gets shot,	-	-	-	S	O
and you think that of me?	S	X	-	-	-
No. I am the one who knocks!	-	S	-	-	-

Figure 16.3: The entity grid representation for a dialogue from the television show *Breaking Bad*.

Centering theory unifies aspects of syntax, discourse, and anaphora resolution. However, it can be difficult to clarify exactly how to rank the elements of each utterance, or even how to partition a text or dialog into utterances (Poesio et al., 2004).

16.2.2 The entity grid

One way to formalize the ideas of centering theory is to arrange the entities in a text or conversation in an **entity grid**. This is a data structure with one row per sentence, and one column per entity (Barzilay and Lapata, 2008). Each cell $c(m, i)$ can take the following values:

$$c(m,i) = \begin{cases} S, & \text{entity } i \text{ is in subject position in sentence } m \\ O, & \text{entity } i \text{ is in object position in sentence } m \\ X, & \text{entity } i \text{ appears in sentence } m, \text{ in neither subject nor object position} \\ -, & \text{entity } i \text{ does not appear in sentence } m. \end{cases} \quad [16.3]$$

To populate the entity grid, syntactic parsing is applied to identify subject and object positions, and coreference resolution is applied to link multiple mentions of a single entity. An example is shown in Figure 16.3.

After the grid is constructed, the coherence of a document can be measured by the *transitions* between adjacent cells in each column. For example, the transition $(S \rightarrow S)$ keeps an entity in subject position across adjacent sentences; the transition $(O \rightarrow S)$ promotes an entity from object position to subject position; the transition $(S \rightarrow -)$ drops the subject of one sentence from the next sentence. The probabilities of each transition can be

estimated from labeled data, and an entity grid can then be scored by the sum of the log-probabilities across all columns and all transitions, $\sum_{i=1}^{N_e} \sum_{m=1}^{M} \log \text{p}(c(m,i) \mid c(m-1,i))$. The resulting probability can be used as a proxy for the coherence of a text. This has been shown to be useful for a range of tasks: determining which of a pair of articles is more readable (Schwarm and Ostendorf, 2005), correctly ordering the sentences in a scrambled text (Lapata, 2003), and disentangling multiple conversational threads in an online multi-party chat (Elsner and Charniak, 2010).

16.2.3 *Formal semantics beyond the sentence level

An alternative view of the role of entities in discourse focuses on formal semantics, and the construction of meaning representations for multi-sentence units. Consider the following two sentences (from Bird et al., 2009):

(16.3) a. Angus owns a dog.
b. It bit Irene.

We would like to recover the formal semantic representation,

$$\exists x. \text{DOG}(x) \wedge \text{OWN}(\text{ANGUS}, x) \wedge \text{BITE}(x, \text{IRENE}). \quad [16.4]$$

However, the semantic representations of each individual sentence are,

$$\exists x. \text{DOG}(x) \wedge \text{OWN}(\text{ANGUS}, x) \quad [16.5]$$
$$\text{BITE}(y, \text{IRENE}). \quad [16.6]$$

Unifying these two representations into the form of Equation 16.4 requires linking the unbound variable y from [16.6] with the quantified variable x in [16.5].[2] Discourse understanding therefore requires the reader to update a set of assignments, from variables to entities. This update would (presumably) link the *dog* in the first sentence of [16.3] with the unbound variable y in the second sentence, thereby licensing the conjunction in [16.4].[3] This basic idea is at the root of **dynamic semantics** (Groenendijk and Stokhof, 1991). **Segmented discourse representation theory** links dynamic semantics with a set of **discourse relations**, which explain how adjacent units of text are rhetorically or conceptually related (Lascarides and Asher, 2007). The next section explores the theory of discourse relations in more detail.

[2] Groenendijk and Stokhof (1991) treats the y variable in Equation 16.6 as unbound. Even if it were bound locally with an existential quantifier ($\exists y \text{BITE}(y, \text{IRENE})$), the variable would still need to be reconciled with the quantified variable in Equation 16.5.

[3] This linking task is similar to coreference resolution (see chapter 15), but here the connections are between semantic variables, rather than spans of text.

16.3 Relations

In dependency grammar, sentences are characterized by a graph (usually a tree) of syntactic relations between words, such as NSUBJ and DET. A similar idea can be applied at the document level, identifying relations between discourse units, such as clauses, sentences, or paragraphs. The task of **discourse parsing** involves identifying discourse units and the relations that hold between them. These relations can then be applied to tasks such as document classification and summarization, as discussed in § 16.3.4.

16.3.1 Shallow discourse relations

The existence of discourse relations is hinted by **discourse connectives**, such as *however*, *moreover*, *meanwhile*, and *if . . . then*. These connectives explicitly specify the relationship between adjacent units of text: *however* signals a contrastive relationship, *moreover* signals that the subsequent text elaborates or strengthens the point that was made immediately beforehand, *meanwhile* indicates that two events are contemporaneous, and *if . . . then* sets up a conditional relationship. Discourse connectives can therefore be viewed as a starting point for the analysis of discourse relations.

In **lexicalized tree-adjoining grammar for discourse (D-LTAG)**, each connective anchors a relationship between two units of text (Webber, 2004). This model provides the theoretical basis for the **Penn Discourse Treebank (PDTB)**, the largest corpus of discourse relations in English (Prasad et al., 2008). It includes a hierarchical inventory of discourse relations (shown in Table 16.1), which is created by abstracting the meanings implied by the discourse connectives that appear in real texts (Knott, 1996). These relations are then annotated on the same corpus of news text used in the Penn Treebank (see § 9.2.2), adding the following information:

- Each connective is annotated for the discourse relation or relations that it expresses, if any — many discourse connectives have senses in which they do not signal a discourse relation (Pitler and Nenkova, 2009).
- For each discourse relation, the two arguments of the relation are specified as ARG1 and ARG2, where ARG2 is constrained to be adjacent to the connective. These arguments may be sentences, but they may also smaller or larger units of text.
- Adjacent sentences are annotated for **implicit discourse relations**, which are not marked by any connective. When a connective could be inserted between a pair of sentence, the annotator supplies it, and also labels its sense (e.g., example 16.5). In some cases, there is no relationship at all between a pair of adjacent sentences; in other cases, the only relation is that the adjacent sentences mention one or more shared entity. These phenomena are annotated as NOREL and ENTREL (entity relation), respectively.

- TEMPORAL
  - Asynchronous
  - Synchronous: precedence, succession
- CONTINGENCY
  - Cause: result, reason
  - Pragmatic cause: justification
  - Condition: hypothetical, general, unreal present, unreal past, real present, real past
  - Pragmatic condition: relevance, implicit assertion
- COMPARISON
  - Contrast: juxtaposition, opposition
  - Pragmatic contrast
  - Concession: expectation, contra-expectation
  - Pragmatic concession
- EXPANSION
  - Conjunction
  - Instantiation
  - Restatement: specification, equivalence, generalization
  - Alternative: conjunctive, disjunctive, chosen alternative
  - Exception
  - List

Table 16.1: The hierarchy of discourse relation in the Penn Discourse Treebank annotations (Prasad et al., 2008). For example, PRECEDENCE is a subtype of SYNCHRONOUS, which is a type of TEMPORAL relation.

Examples of Penn Discourse Treebank annotations are shown in (16.4). In (16.4), the word *therefore* acts as an explicit discourse connective, linking the two adjacent units of text. The Treebank annotations also specify the "sense" of each relation, linking the connective to a relation in the sense inventory shown in Table 16.1: in (16.4), the relation is PRAGMATIC CAUSE:JUSTIFICATION because it relates to the author's communicative intentions. The word *therefore* can also signal causes in the external world (e.g., *He was therefore forced to relinquish his plan*). In **discourse sense classification**, the goal is to determine which discourse relation, if any, is expressed by each connective. A related task is the classification of implicit discourse relations, as in (16.5). In this example, the relationship between the adjacent sentences could be expressed by the connective *because*, indicating a CAUSE:REASON relationship.

Classifying explicit discourse relations and their arguments

As suggested by the examples above, many connectives can be used to invoke multiple types of discourse relations. Similarly, some connectives have senses that are unrelated to discourse: for example, *and* functions as a discourse connective when it links propo-

(16.4) *...as this business of whaling has somehow come to be regarded among landsmen as a rather unpoetical and disreputable pursuit*; therefore, **I am all anxiety to convince ye, ye landsmen, of the injustice hereby done to us hunters of whales**.

(16.5) But a few funds have taken other defensive steps. *Some have raised their cash positions to record levels.* Implicit = BECAUSE **High cash positions help buffer a fund when the market falls.**

(16.6) Michelle lives in a hotel room, and although **she drives a canary-colored Porsche**, *she hasn't time to clean or repair it*.

(16.7) *Most oil companies*, when **they set exploration and production budgets for this year**, *forecast revenue of $15 for each barrel of crude produced*.

Figure 16.4: Example annotations of discourse relations. In the style of the Penn Discourse Treebank, the discourse connective is underlined, the first argument is shown in italics, and the second argument is shown in bold. Examples (16.5-16.7) are quoted from Prasad et al. (2008).

sitions, but not when it links noun phrases (Lin et al., 2014). Nonetheless, the senses of explicitly-marked discourse relations in the Penn Treebank are relatively easy to classify, at least at the coarse-grained level. When classifying the four top-level PDTB relations, 90% accuracy can be obtained simply by selecting the most common relation for each connective (Pitler and Nenkova, 2009). At the more fine-grained levels of the discourse relation hierarchy, connectives are more ambiguous. This fact is reflected both in the accuracy of automatic sense classification (Versley, 2011) and in interannotator agreement, which falls to 80% for level-3 discourse relations (Prasad et al., 2008).

A more challenging task for explicitly-marked discourse relations is to identify the scope of the arguments. Discourse connectives need not be adjacent to ARG1, as shown in item 16.6, where ARG1 follows ARG2; furthermore, the arguments need not be contiguous, as shown in (16.7). For these reasons, recovering the arguments of each discourse connective is a challenging subtask. Because intra-sentential arguments are often syntactic constituents (see chapter 10), many approaches train a classifier to predict whether each constituent is an appropriate argument for each explicit discourse connective (Wellner and Pustejovsky, 2007; Lin et al., 2014, e.g.,).

Classifying implicit discourse relations

Implicit discourse relations are considerably more difficult to classify and to annotate.[4] Most approaches are based on an encoding of each argument, which is then used as input

[4] In the dataset for the 2015 shared task on shallow discourse parsing, the interannotator agreement was 91% for explicit discourse relations and 81% for implicit relations, across all levels of detail (Xue et al., 2015).

to a nonlinear classifier:

$$\boldsymbol{z}^{(i)} = \text{Encode}(\boldsymbol{w}^{(i)}) \quad [16.7]$$

$$\boldsymbol{z}^{(i+1)} = \text{Encode}(\boldsymbol{w}^{(i+1)}) \quad [16.8]$$

$$\hat{y}_i = \underset{y}{\text{argmax}}\ \Psi(y, \boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i+1)}). \quad [16.9]$$

This basic framework can be instantiated in several ways, including both feature-based and neural encoders.

Feature-based approaches Each argument can be encoded into a vector of surface features. The encoding typically includes lexical features (all words, or all content words, or a subset of words such as the first three and the main verb), Brown clusters of individual words (§ 14.4), and syntactic features such as terminal productions and dependency arcs (Pitler et al., 2009; Lin et al., 2009; Rutherford and Xue, 2014). The classification function then has two parts. First, it creates a joint feature vector by combining the encodings of each argument, typically by computing the cross-product of all features in each encoding:

$$\boldsymbol{f}(y, \boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i+1)}) = \{(a \times b \times y) : (\boldsymbol{z}_a^{(i)} \boldsymbol{z}_b^{(i+1)})\} \quad [16.10]$$

The size of this feature set grows with the square of the size of the vocabulary, so it can be helpful to select a subset of features that are especially useful on the training data (Park and Cardie, 2012). After $\boldsymbol{f}$ is computed, any classifier can be trained to compute the final score, $\Psi(y, \boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i+1)}) = \boldsymbol{\theta} \cdot \boldsymbol{f}(y, \boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i+1)})$.

Neural network approaches In neural network architectures, the encoder is learned jointly with the classifier as an end-to-end model. Each argument can be encoded using a variety of neural architectures (surveyed in § 14.8): recursive (§ 10.6.1; Ji and Eisenstein, 2015), recurrent (§ 6.3; Ji et al., 2016), and convolutional (§ 3.4; Qin et al., 2017). The classification function can then be implemented as a feedforward neural network on the two encodings (chapter 3; for examples, see Rutherford et al., 2017; Qin et al., 2017), or as a simple bilinear product, $\Psi(y, \boldsymbol{z}^{(i)}, \boldsymbol{z}^{(i+1)}) = (\boldsymbol{z}^{(i)})^\top \boldsymbol{\Theta}_y \boldsymbol{z}^{(i+1)}$ (Ji and Eisenstein, 2015). The encoding model can be trained by backpropagation from the classification objective, such as the margin loss. Rutherford et al. (2017) show that neural architectures outperform feature-based approaches in most settings. While neural approaches require engineering the network architecture (e.g., embedding size, number of hidden units in the classifier), feature-based approaches also require significant engineering to incorporate linguistic resources such as Brown clusters and parse trees, and to select a subset of relevant features.

16.3.2 Hierarchical discourse relations

In sentence parsing, adjacent phrases combine into larger constituents, ultimately producing a single constituent for the entire sentence. The resulting tree structure enables structured analysis of the sentence, with subtrees that represent syntactically coherent chunks of meaning. **Rhetorical Structure Theory (RST)** extends this style of hierarchical analysis to the discourse level (Mann and Thompson, 1988).

The basic element of RST is the **discourse unit**, which refers to a contiguous span of text. **Elementary discourse units** (EDUs) are the atomic elements in this framework, and are typically (but not always) clauses.[5] Each discourse relation combines two or more adjacent discourse units into a larger, composite discourse unit; this process ultimately unites the entire text into a tree-like structure.[6]

Nuclearity In many discourse relations, one argument is primary. For example:

(16.8) [LaShawn loves animals]$_N$
[She has nine dogs and one pig]$_S$

In this example, the second sentence provides EVIDENCE for the point made in the first sentence. The first sentence is thus the **nucleus** of the discourse relation, and the second sentence is the **satellite**. The notion of nuclearity is similar to the head-modifier structure of dependency parsing (see § 11.1.1). However, in RST, some relations have multiple nuclei. For example, the arguments of the CONTRAST relation are equally important:

(16.9) [The clash of ideologies survives this treatment]$_N$
[but the nuance and richness of Gorky's individual characters have vanished in the scuffle]$_N$[7]

Relations that have multiple nuclei are called **coordinating**; relations with a single nucleus are called **subordinating**. Subordinating relations are constrained to have only two arguments, while coordinating relations (such as CONJUNCTION) may have more than two.

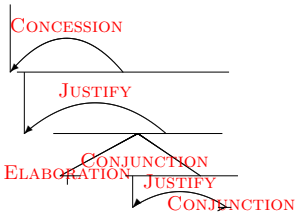
[5] Details of discourse segmentation can be found in the RST annotation manual (Carlson and Marcu, 2001).

[6] While RST analyses are typically trees, this should not be taken as a strong theoretical commitment to the principle that all coherent discourses have a tree structure. Taboada and Mann (2006) write:

> It is simply the case that trees are convenient, easy to represent, and easy to understand. There is, on the other hand, no theoretical reason to assume that trees are the only possible representation of discourse structure and of coherence relations.

The appropriateness of tree structures to discourse has been challenged, e.g., by Wolf and Gibson (2005), who propose a more general graph-structured representation.

[7] from the RST Treebank (Carlson et al., 2002)



[It could have been a great movie]^{1A} [It does have beautiful scenery,]^{1B} [some of the best since Lord of the Rings.]^{1C} [The acting is well done,]^{1D} [and I really liked the son of the leader of the Samurai.]^{1E} [He was a likable chap,]^{1F} [and I hated to see him die.]^{1G} [But, other than all that, this movie is nothing more than hidden rip-offs.]^{1H}

Figure 16.5: A rhetorical structure theory analysis of a short movie review, adapted from Voll and Taboada (2007). Positive and negative sentiment words are underlined, indicating RST's potential utility in document-level sentiment analysis.

RST Relations Rhetorical structure theory features a large inventory of discourse relations, which are divided into two high-level groups: subject matter relations, and presentational relations. Presentational relations are organized around the intended beliefs of the reader. For example, in (16.8), the second discourse unit provides evidence intended to increase the reader's belief in the proposition expressed by the first discourse unit, that *LaShawn loves animals*. In contrast, subject-matter relations are meant to communicate additional facts about the propositions contained in the discourse units that they relate:

(16.10) [the debt plan was rushed to completion]$_N$
[in order to be announced at the meeting]$_S$[8]

In this example, the satellite describes a world state that is realized by the action described in the nucleus. This relationship is about the world, and not about the author's communicative intentions.

Example Figure 16.5 depicts an RST analysis of a paragraph from a movie review. Asymmetric (subordinating) relations are depicted with an arrow from the satellite to the nucleus; symmetric (coordinating) relations are depicted with lines. The elementary discourse units 1F and 1G are combined into a larger discourse unit with the symmetric CONJUNCTION relation. The resulting discourse unit is then the satellite in a JUSTIFY relation with 1E.

[8] from the RST Treebank (Carlson et al., 2002)

Hierarchical discourse parsing

The goal of discourse parsing is to recover a hierarchical structural analysis from a document text, such as the analysis in Figure 16.5. For now, let's assume a segmentation of the document into elementary discourse units (EDUs); segmentation algorithms are discussed below. After segmentation, discourse parsing can be viewed as a combination of two components: the discourse relation classification techniques discussed in § 16.3.1, and algorithms for phrase-structure parsing, such as chart parsing and shift-reduce, which were discussed in chapter 10.

Both chart parsing and shift-reduce require encoding composite discourse units, either in a discrete feature vector or a dense neural representation.[9] Some discourse parsers rely on the **strong compositionality criterion** (Marcu, 1996), which states the assumption that a composite discourse unit can be represented by its nucleus. This criterion is used in feature-based discourse parsing to determine the feature vector for a composite discourse unit (Hernault et al., 2010); it is used in neural approaches to setting the vector encoding for a composite discourse unit equal to the encoding of its nucleus (Ji and Eisenstein, 2014). An alternative neural approach is to learn a composition function over the components of a composite discourse unit (Li et al., 2014), using a recursive neural network (see § 14.8.3).

Bottom-up discourse parsing Assume a segmentation of the text into N elementary discourse units with base representations $\{\boldsymbol{z}^{(i)}\}_{i=1}^{N}$, and assume a composition function $\textsc{Compose}\left(\boldsymbol{z}^{(i)}, \boldsymbol{z}^{(j)}, \ell\right)$, which maps two encodings and a discourse relation ℓ into a new encoding. The composition function can follow the strong compositionality criterion and simply select the encoding of the nucleus, or it can do something more complex. We also need a scoring function $\Psi(\boldsymbol{z}^{(i,k)}, \boldsymbol{z}^{(k,j)}, \ell)$, which computes a scalar score for the (binarized) discourse relation ℓ with left child covering the span $i+1:k$, and the right child covering the span $k+1:j$. Given these components, we can construct vector representations for each span, and this is the basic idea underlying **compositional vector grammars** (Socher et al., 2013).

These same components can also be used in bottom-up parsing, in a manner that is similar to the CKY algorithm for weighted context-free grammars (see § 10.1): compute the score and best analysis for each possible span of increasing lengths, while storing back-pointers that make it possible to recover the optimal parse of the entire input. However, there is an important distinction from CKY parsing: for each labeled span (i, j, ℓ), we must use the composition function to construct a representation $\boldsymbol{z}^{(i,j,\ell)}$. This representation is then used to combine the discourse unit spanning $i+1:j$ in higher-level discourse relations. The representation $\boldsymbol{z}^{(i,j,\ell)}$ depends on the entire substructure of the unit span-

[9] To use these algorithms, is also necessary to binarize all discourse relations during parsing, and then to "unbinarize" them to reconstruct the desired structure (e.g., Hernault et al., 2010).

ning $i + 1 : j$, and this violates the locality assumption that underlie CKY's optimality guarantee. Bottom-up parsing with recursively constructed span representations is generally not guaranteed to find the best-scoring discourse parse. This problem is explored in an exercise at the end of the chapter.

Transition-based discourse parsing One drawback of bottom-up parsing is its cubic time complexity in the length of the input. For long documents, transition-based parsing is an appealing alternative. The shift-reduce algorithm (see § 10.6.2) can be applied to discourse parsing fairly directly (Sagae, 2009): the stack stores a set of discourse units and their representations, and each action is chosen by a function of these representations. This function could be a linear product of weights and features, or it could be a neural network applied to encodings of the discourse units. The REDUCE action then performs composition on the two discourse units at the top of the stack, yielding a larger composite discourse unit, which goes on top of the stack. All of the techniques for integrating learning and transition-based parsing, described in § 11.3, are applicable to discourse parsing.

Segmenting discourse units

In rhetorical structure theory, elementary discourse units do not cross the sentence boundary, so discourse segmentation can be performed within sentences, assuming the sentence segmentation is given. The segmentation of sentences into elementary discourse units is typically performed using features of the syntactic analysis (Braud et al., 2017). One approach is to train a classifier to determine whether each syntactic constituent is an EDU, using features such as the production, tree structure, and head words (Soricut and Marcu, 2003; Hernault et al., 2010). Another approach is to train a sequence labeling model, such as a conditional random field (Sporleder and Lapata, 2005; Xuan Bach et al., 2012; Feng et al., 2014). This is done using the BIO formalism for segmentation by sequence labeling, described in § 8.3.

16.3.3 Argumentation

An alternative view of text-level relational structure focuses on **argumentation** (Stab and Gurevych, 2014b). Each segment (typically a sentence or clause) may support or rebut another segment, creating a graph structure over the text. In the following example (from Peldszus and Stede, 2013), segment $S2$ provides argumentative support for the proposition in the segment $S1$:

(16.11) [We should tear the building down,]$_{S1}$
[because it is full of asbestos]$_{S2}$.

Assertions may also support or rebut proposed links between two other assertions, creating a **hypergraph**, which is a generalization of a graph to the case in which edges can

join any number of vertices. This can be seen by introducing another sentence into the example:

(16.12) [In principle it is possible to clean it up,]$_{S3}$
[but according to the mayor that is too expensive.]$_{S4}$

S3 acknowledges the validity of $S2$, but **undercuts** its support of $S1$. This can be represented by introducing a hyperedge, $(S3, S2, S1)_{\text{undercut}}$, indicating that $S3$ undercuts the proposed relationship between $S2$ and $S1$. $S4$ then undercuts the relevance of $S3$.

Argumentation mining is the task of recovering such structures from raw texts. At present, annotations of argumentation structure are relatively small: Stab and Gurevych (2014a) have annotated a collection of 90 persuasive essays, and Peldszus and Stede (2015) have solicited and annotated a set of 112 paragraph-length "microtexts" in German.

16.3.4 Applications of discourse relations

The predominant application of discourse parsing is to select content within a document. In rhetorical structure theory, the nucleus is considered the more important element of the relation, and is more likely to be part of a summary of the document; it may also be more informative for document classification. The D-LTAG theory that underlies the Penn Discourse Treebank lacks this notion of nuclearity, but arguments may have varying importance, depending on the relation type. For example, the span of text constituting ARG1 of an expansion relation is more likely to appear in a summary, while the sentence constituting ARG2 of an implicit relation is less likely (Louis et al., 2010). Discourse relations may also signal segmentation points in the document structure. Explicit discourse markers have been shown to correlate with changes in subjectivity, and identifying such change points can improve document-level sentiment classification, by helping the classifier to focus on the subjective parts of the text (Trivedi and Eisenstein, 2013; Yang and Cardie, 2014).

Extractive Summarization

Text **summarization** is the problem of converting a longer text into a shorter one, while still conveying the key facts, events, ideas, and sentiments from the original. In **extractive summarization**, the summary is a subset of the original text; in **abstractive summarization**, the summary is produced *de novo*, by paraphrasing the original, or by first encoding it into a semantic representation (see § 19.2). The main strategy for extractive summarization is to maximize coverage, choosing a subset of the document that best covers the concepts mentioned in the document as a whole; typically, coverage is approximated by bag-of-words overlap (Nenkova and McKeown, 2012). Coverage-based objectives can be supplemented by hierarchical discourse relations, using the principle of nuclearity: in any subordinating discourse relation, the nucleus is more critical to the overall meaning

of the text, and is therefore more important to include in an extractive summary (Marcu, 1997a).[10] This insight can be generalized from individual relations using the concept of **discourse depth** (Hirao et al., 2013): for each elementary discourse unit e, the discourse depth d_e is the number of relations in which a discourse unit containing e is the satellite.

Both discourse depth and nuclearity can be incorporated into extractive summarization, using constrained optimization. Let $\boldsymbol{x}_n$ be a bag-of-words vector representation of elementary discourse unit n, let $y_n \in \{0, 1\}$ indicate whether n is included in the summary, and let d_n be the depth of unit n. Furthermore, let each discourse unit have a "head" h, which is defined recursively:

- if a discourse unit is produced by a subordinating relation, then its head is the head of the (unique) nucleus;
- if a discourse unit is produced by a coordinating relation, then its head is the head of the left-most nucleus;
- for each elementary discourse unit, its parent $\pi(n) \in \{\varnothing, 1, 2, \ldots, N\}$ is the head of the smallest discourse unit containing n whose head is not n;
- if n is the head of the discourse unit spanning the whole document, then $\pi(n) = \varnothing$.

With these definitions in place, discourse-driven extractive summarization can be formalized as (Hirao et al., 2013),

$$\begin{aligned} \max_{\boldsymbol{y}=\{0,1\}^N} & \sum_{n=1}^{N} y_n \frac{\Psi\left(\boldsymbol{x}_n, \{\boldsymbol{x}_{1:N}\}\right)}{d_n} \\ \text{s.t.} & \sum_{n=1}^{N} y_n (\sum_{j=1}^{V} x_{n,j}) \leq L \\ & y_{\pi(n)} \geq y_n, \quad \forall n \text{ s.t. } \pi(n) \neq \varnothing \end{aligned} \tag{16.11}$$

where $\Psi\left(\boldsymbol{x}_n, \{\boldsymbol{x}_{1:N}\}\right)$ measures the coverage of elementary discourse unit n with respect to the rest of the document, and $\sum_{j=1}^{V} x_{n,j}$ is the number of tokens in $\boldsymbol{x}_n$. The first constraint ensures that the number of tokens in the summary has an upper bound L. The second constraint ensures that no elementary discourse unit is included unless its parent is also included. In this way, the discourse structure is used twice: to downweight the contributions of elementary discourse units that are not central to the discourse, and to ensure that the resulting structure is a subtree of the original discourse parse. The opti-

[10] Conversely, the arguments of a multi-nuclear relation should either both be included in the summary, or both excluded (Durrett et al., 2016).

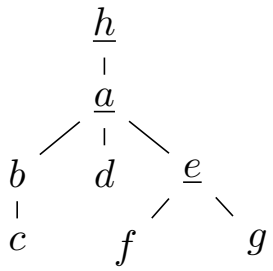


Figure 16.6: A discourse depth tree (Hirao et al., 2013) for the discourse parse from Figure 16.5, in which each elementary discourse unit is connected to its parent. The discourse units in one valid summary are underlined.

mization problem in 16.11 can be solved with **integer linear programming**, described in § 13.2.2.[11]

Figure 16.6 shows a discourse depth tree for the RST analysis from Figure 16.5, in which each elementary discourse is connected to (and below) its parent. The underlined discourse units in the figure constitute the following summary:

(16.13) It could have been a great movie, and I really liked the son of the leader of the Samurai. But, other than all that, this movie is nothing more than hidden rip-offs.

Document classification

Hierarchical discourse structures lend themselves naturally to text classification: in a subordinating discourse relation, the nucleus should play a stronger role in the classification decision than the satellite. Various implementations of this idea have been proposed.

- Focusing on within-sentence discourse relations and lexicon-based classification (see § 4.1.2), Voll and Taboada (2007) simply ignore the text in the satellites of each discourse relation.
- At the document level, elements of each discourse relation argument can be reweighted, favoring words in the nucleus, and disfavoring words in the satellite (Heerschop et al., 2011; Bhatia et al., 2015). This approach can be applied recursively, computing weights across the entire document. The weights can be relation-specific, so that the features from the satellites of contrastive relations are discounted or even reversed.
- Alternatively, the hierarchical discourse structure can define the structure of a **recursive neural network** (see § 10.6.1). In this network, the representation of each

[11] Formally, 16.11 is a special case of the **knapsack problem**, in which the goal is to find a subset of items with maximum value, constrained by some maximum weight (Cormen et al., 2009).

discourse unit is computed from its arguments and from a parameter corresponding to the discourse relation (Ji and Smith, 2017).

Shallow, non-hierarchical discourse relations have also been applied to document classification. One approach is to impose a set of constraints on the analyses of individual discourse units, so that adjacent units have the same polarity when they are connected by a discourse relation indicating agreement, and opposite polarity when connected by a contrastive discourse relation, indicating disagreement (Somasundaran et al., 2009; Zirn et al., 2011). Yang and Cardie (2014) apply explicitly-marked relations from the Penn Discourse Treebank to the problem of sentence-level sentiment polarity classification (see § 4.1). They impose the following soft constraints:

- When a CONTRAST relation appears at the beginning of a sentence, the sentence should have the opposite sentiment polarity as its predecessor.
- When an EXPANSION or CONTINGENCY appears at the beginning of a sentence, it should have the same polarity as its predecessor.
- When a CONTRAST relation appears *within* a sentence, the sentence should have neutral polarity, since it is likely to express both sentiments.

These discourse-driven constraints are shown to improve performance on two datasets of product reviews.

Coherence

Just as **grammaticality** is the property shared by well-structured sentences, **coherence** is the property shared by well-structured discourses. One application of discourse processing is to measure (and maximize) the coherence of computer-generated texts like translations and summaries (Kibble and Power, 2004). Coherence assessment is also used to evaluate human-generated texts, such as student essays (e.g., Miltsakaki and Kukich, 2004; Burstein et al., 2013).

Coherence subsumes a range of phenomena, many of which have been highlighted earlier in this chapter: e.g., that adjacent sentences should be lexically cohesive (Foltz et al., 1998; Ji et al., 2015; Li and Jurafsky, 2017), and that entity references should follow the principles of centering theory (Barzilay and Lapata, 2008; Nguyen and Joty, 2017). Discourse relations also bear on the coherence of a text in a variety of ways:

- Hierarchical discourse relations tend to have a "canonical ordering" of the nucleus and satellite (Mann and Thompson, 1988): for example, in the ELABORATION relation from rhetorical structure theory, the nucleus always comes first, while in the JUSTIFICATION relation, the satellite tends to be first (Marcu, 1997b).

- Discourse relations should be signaled by connectives that are appropriate to the semantic or functional relationship between the arguments: for example, a coherent text would be more likely to use *however* to signal a COMPARISON relation than a *temporal* relation (Kibble and Power, 2004).

- Discourse relations tend to be ordered in appear in predictable sequences: for example, COMPARISON relations tend to immediately precede CONTINGENCY relations (Pitler et al., 2008). This observation can be formalized by generalizing the entity grid model (§ 16.2.2), so that each cell (i, j) provides information about the role of the discourse argument containing a mention of entity j in sentence i (Lin et al., 2011). For example, if the first sentence is ARG1 of a comparison relation, then any entity mentions in the sentence would be labeled COMP.ARG1. This approach can also be applied to RST discourse relations (Feng et al., 2014).

Datasets One difficulty with evaluating metrics of discourse coherence is that human-generated texts usually meet some minimal threshold of coherence. For this reason, much of the research on measuring coherence has focused on synthetic data. A typical setting is to permute the sentences of a human-written text, and then determine whether the original sentence ordering scores higher according to the proposed coherence measure (Barzilay and Lapata, 2008). There are also small datasets of human evaluations of the coherence of machine summaries: for example, human judgments of the summaries from the participating systems in the 2003 Document Understanding Conference are available online.[12] Researchers from the Educational Testing Service (an organization which administers several national exams in the United States) have studied the relationship between discourse coherence and student essay quality (Burstein et al., 2003, 2010). A public dataset of essays from second-language learners, with quality annotations, has been made available by researchers at Cambridge University (Yannakoudakis et al., 2011). At the other extreme, Louis and Nenkova (2013) analyze the structure of professionally written scientific essays, finding that discourse relation transitions help to distinguish prize-winning essays from other articles in the same genre.

Additional resources

For a manuscript-length discussion of discourse processing, see Stede (2011). Article-length surveys are offered by Webber et al. (2012) and Webber and Joshi (2012).

[12] http://homepages.inf.ed.ac.uk/mlap/coherence/

Exercises

1. Some discourse connectives tend to occur between their arguments; others can precede both arguments, and a few can follow both arguments. Indicate whether the following connectives can occur between, before, and after their arguments: *however*, *but*, *while* (contrastive, not temporal), *although*, *therefore*, *nonetheless*.

2. This exercise is to be done in pairs. Each participant selects an article from today's news, and replaces all mentions of individual people with special tokens like PERSON1, PERSON2, and so on. The other participant should then use the rules of centering theory to guess each type of referring expression: full name (*Captain Ahab*), partial name (e.g., *Ahab*), nominal (e.g., *the ship's captain*), or pronoun. Check whether the predictions match the original text, and whether the text conforms to the rules of centering theory.

3. In this exercise, you will produce a figure similar to Figure 16.1.

   a) Implement the smoothed cosine similarity metric from Equation 16.2, using the smoothing kernel $\boldsymbol{k} = [.5, .3, .15, .05]$.

   b) Download the text of a news article with at least ten paragraphs.

   c) Compute and plot the smoothed similarity $\overline{\boldsymbol{s}}$ over the length of the article.

   d) Identify *local minima* in $\overline{\boldsymbol{s}}$ as follows: first find all sentences m such that $\overline{s}_m < \overline{s}_{m\pm1}$. Then search among these points to find the five sentences with the lowest $\overline{s}_m$.

   e) How often do the five local minima correspond to paragraph boundaries?

      - The fraction of local minima that are paragraph boundaries is the **precision-at-k**, where in this case, $k = 5$.
      - The fraction of paragraph boundaries which are local minima is the **recall-at-k**.
      - Compute precision-at-k and recall-at-k for $k = 3$ and $k = 10$.

4. One way to formulate text segmentation as a probabilistic model is through the use of the **Dirichlet Compound Multinomial** (DCM) distribution, which computes the probability of a bag-of-words, $\text{DCM}(\boldsymbol{x}; \boldsymbol{\alpha})$, where the parameter $\boldsymbol{\alpha}$ is a vector of positive reals. This distribution can be configured to assign high likelihood to bag-of-words vectors that are internally coherent, such that individual words appear repeatedly: for example, this behavior can be observed for simple parameterizations, such as $\boldsymbol{\alpha} = \alpha\mathbf{1}$ with $\alpha < 1$.

   Let $\psi_\alpha(i, j)$ represent the log-probability of a segment $\boldsymbol{w}_{i+1:j}$ under a DCM distribution with parameter $\boldsymbol{\alpha}$. Give a dynamic program for segmenting a text into a total

of K segments maximizing the sum of log-probabilities $\sum_{k=1}^{K} \psi_\alpha(s_{k-1}, s_k)$, where s_k indexes the last token of segment k, and $s_0 = 0$. The time complexity of your dynamic program should not be worse than quadratic in the length of the input and linear in the number of segments.

5. Building on the previous problem, you will now adapt the CKY algorithm to perform hierarchical segmentation. Define a hierarchical segmentation as a set of segmentations $\{\{s_k^{(\ell)}\}_{k=1}^{K^{(\ell)}}\}_{\ell=1}^{L}$, where L is the segmentation depth. To ensure that the segmentation is hierarchically valid, we require that each segmentation point $s_k^{(\ell)}$ at level ℓ is also a segmentation point at level $\ell - 1$, where $\ell > 1$.

   For simplicity, this problem focuses on binary hierarchical segmentation, so that each segment at level $\ell > 1$ has exactly 2 subsegments. Define the score of a hierarchical segmentation as the sum of the scores of all segments (at all levels), using the the DCM log-probabilities from the previous problem as the segment scores. Give a CKY-like recurrence such that the optimal "parse" of the text is the maximum log-probability binary segmentation with exactly L levels.

6. The entity grid representation of centering theory can be used to compute a score for adjacent sentences, as described in § 16.2.2. Given a set of sentences, these scores can be used to compute an optimal ordering. Show that finding the ordering with the maximum log probability is NP-complete, by reduction from a well-known problem.

7. In § 16.3.2, it is noted that bottom-up parsing with compositional vector representations of each span is not guaranteed to be optimal. In this exercise, you will construct a minimal example proving this point. Consider a discourse with four units, with base representations $\{\boldsymbol{z}^{(i)}\}_{i=1}^{4}$. Construct a scenario in which the parse selected by bottom-up parsing is not optimal, and give the precise mathematical conditions under which this suboptimal parse is selected. You may ignore the relation labels ℓ for the purpose of this example.

8. As noted in § 16.3.3, arguments can described by hypergraphs, in which a segment may **undercut** a proposed edge between two other segments. Extend the model of extractive summarization described in § 16.3.4 to arguments, adding the follwoing constraint: if segment i undercuts an argumentative relationship between j and k, then i cannot be included in the summary unless both j and k are included. Your solution should take the form of a set of *linear* constraints on an integer linear program — that is, each constraint can only involve addition and subtraction of variables.

In the next two exercises, you will explore the use of discourse connectives in a real corpus. Using NLTK, acquire the Brown corpus, and identify sentences that begin with any of the following connectives: *however, nevertheless, moreover, furthermore, thus*.

9. Both lexical consistency and discourse connectives contribute to the **cohesion** of a text. We might therefore expect adjacent sentences that are joined by explicit discourse connectives to also have higher word overlap. Using the Brown corpus, test this theory by computing the average cosine similarity between adjacent sentences that are connected by one of the connectives mentioned above. Compare this to the average cosine similarity of all other adjacent sentences. If you know how, perform a two-sample t-test to determine whether the observed difference is statistically significant.

10. Group the above connectives into the following three discourse relations:

    - Expansion: *moreover, furthermore*
    - Comparison: *however, nevertheless*
    - Contingency: *thus*

    Focusing on pairs of sentences which are joined by one of these five connectives, build a classifier to predict the discourse relation from the text of the two adjacent sentences — taking care to ignore the connective itself. Use the first 30000 sentences of the Brown corpus as the training set, and the remaining sentences as the test set. Compare the performance of your classifier against simply choosing the most common class. Using a bag-of-words classifier, it is hard to do much better than this baseline, so consider more sophisticated alternatives!