# 1

# Topic Segmentation

Matthew Purver

*Queen Mary University of London*

*This chapter discusses the task of* topic segmentation*: automatically dividing single long recordings or transcripts into shorter, topically coherent segments. First, we look at the task itself, the applications which require it, and some ways to evaluate accuracy. We then explain the most influential approaches – generative and discriminative, supervised and unsupervised – and discuss their application in particular domains.*

## 1.1 Task Description

### *1.1.1 Introduction*

So far, we have mainly looked at techniques for understanding on a fine-grained, bottom-up level: identifying fundamental units of meaning or interactional structure, such as sentences, named entities and dialogue acts. In this chapter, we look at the problem from a more coarse-grained, top-down perspective: given a complete recording or transcript (which may be quite long, and include talk on all sorts of different subjects) can we divide it into shorter, more useful, topically coherent segments?

There are many reasons why we might want to do this, but perhaps the most obvious is that it makes it much easier for a user to browse or search the results. Imagine being faced with a long uninterrupted transcript of a news broadcast or a business meeting. If you want to find a particular news story, or the discussion of a particular subject, you're faced with a problem - particularly if you don't want to read or listen to the whole thing. You could search for relevant keywords, of course: but finding them doesn't tell you where the part you're interested in starts (or ends). There's no guarantee that you'll find the keywords you've chosen either, of course - particularly if ASR word error rates are high. But if you're given the same transcript divided up into segments, with each corresponding to a different topic (or news story, or agenda item), the task becomes much easier. If you can find the keywords you're looking for in a particular segment, you just have to go to the beginning of that segment and start listening (or reading). In the worst case, you could examine the start of

*Spoken Language Understanding* Gokhan Tur and Renato de Mori

each segment until you find what you're looking for: still much easier than reading the whole thing.

We can go much further than this, of course: we might want to analyse or classify the contents of each segment, so that we can relate topics from one meeting to another, or track the progress of news stories across different broadcasts. We might want to produce a condensed summary, with the highlights of each topic (the main headlines of a news story, or the final decision and action items of a meeting agenda item). We'll look at some of these more advanced tasks in later chapters. But the first pre-requisite for each of them is to understand the topic structure: when does the conversation move from one topic to another? When does one topic end, and another one start?

### *1.1.2 What is a Topic?*

The answer to that question, of course, depends to a large extent on what exactly we mean by a *topic* – and this can be hard to define. In particular applications, it may seem obvious: if we're interested in segmenting a news broadcast, our notion of a topic probably corresponds to an individual news story or report. If we want to segment a court transcript, we might be more concerned with the segments in which different arguments are being presented, or different pieces of evidence are being discussed. We might want to divide a business meeting according to the items on the agenda.

However, sometimes it's not so clear. We can imagine a discussion of a single agenda item in a meeting, but which consists of several distinct phases: perhaps a round-table discussion of the problem in hand, then a couple of presentations from individuals on their proposed solutions, and then a concluding decision-making section. While the subject matter might be broadly topically coherent, we can see the segments as representing separate *activities* (discussion vs. presentation), or in the terms of Passonneau and Litman (1997), different *intentions* (question-raising vs. information-giving vs. decision-making). Whether we want to include all of these in one segment, or treat them all separately, really depends on our intentions as users: our interests and desired application (see e.g. Niekrasz and Moore 2009, for discussion in more depth).

This means that in some domains, segmentation can be a hard task even for humans, particularly where subject matter and discourse structure is less constrained. Gruenstein et al. (2008) asked annotators to mark topic shifts in the ICSI Meeting Corpus (Janin et al. 2003) – a collection of open-domain, mostly loosely-structured meetings on subjects which the annotators themselves were not familiar with – and found that they did not agree with each other at all well, especially as the notion of topic became more fine-grained. If people have a clear idea of what they are looking for, though, agreement gets much better: Banerjee and Rudnicky (2007) found that agreement improved significantly if annotators were given more information (an agenda list from which to choose topics). Galley et al. (2003) also found that annotators could achieve reasonable agreement if they stuck to coarse-grained topics – although even then some meetings were problematic.

### *1.1.3 Linear vs. Hierarchical Segmentation*

One of the reasons that it can be hard to define exactly what a topic consists of, and where it starts and ends, is that topics (and discourse itself) often display a *hierarchical* structure

(see e.g. Grosz and Sidner 1986; Mann and Thompson 1988; Marcu 2000; Polanyi 1988). Just as stretches of dialogue can be analysed as being composed of smaller sub-episodes, we can often think of topics and their discussion as being composed of sub-topics and sub-discussions. It might be that an ideal approach to assigning topic structure would be one which assigned not only a linear segmentation, as we've discussed so far, but a hierarchical structure. Retrieval and browsing would then benefit even more, as a user could refine the level of granularity as desired.

However, producing a fine-grained segmentation turns out to be an extremely difficult task. For one thing, as we zoom in to ever-finer distinctions, the information we need to segment the discourse becomes harder to produce. While distinguishing between broad-brush topics might be achievable just from looking at the words people use, or the way people behave (as we'll see below), distinguishing between the discussion of distinct but related issues really requires us to understand something about the semantics of individual contributions and how they inter-relate: the questions people ask, the way they get answered, whether proposals are accepted or not. This is hard: while there are formal models of dialogue which do deal with these matters (see e.g. Asher and Lascarides 2003; Ginzburg 2011; Larsson 2002, amongst others), applying them to open-domain speech isn't yet achievable.

And secondly, it seems that fine sub-topic distinctions are hard for even humans to make. Both Galley et al. (2003) and Gruenstein et al. (2008) found that annotators asked to mark topic shifts over the open-domain ICSI Meeting Corpus often didn't agree with each other at all well; and while they might agree on coarser-grained top-level topics, their agreement became worse on lower-level sub-topics. It may be that these lower-level distinctions really depend on our intentions and requirements; the ideal segmentation may not be definable *a priori*, but may depend on the view we take on the data and the use we're going to put our segmentation to.

Here, then, we restrict ourselves to the task of linear, coarse-grained segmentation: a well-studied task with several alternative approaches, many of which show very encouraging performance.

## 1.2 Basic Approaches, and the Challenge of Speech

The task has been approached in many different ways, and we'll discuss a few of them in more detail below. Here, we take a quick look at the two basic insights that most of them use; while some algorithms are based on one more than another, many combine the two.

### *1.2.1 Changes in Content*

The first one is that people talk about different topics in different ways: they use different words, and refer to different things. If we are discussing a particular set of concepts, we will use words relevant to those concepts; and discussion of particular people, objects or places will involve a relevant set of names and related referring expressions. Repeated mention of the same objects or concepts will therefore be associated with repeated reference, whether by using the same words or phrases or by using co-referent or anaphoric terms (Morris and Hirst 1991). Conversely, a change in topic will be associated with the introduction of new vocabulary (Youmans 1991).

If we look at a discussion containing different topical segments, then, we should see that the vocabulary (and/or the set of referring expressions) being used remains relatively constant during the discussion of each topic, but changes markedly when we move between them. Regions with relatively small changes should then correspond to topic segments, with large changes at the segment boundaries. The same may be true for features of the non-linguistic content, depending on the domain: in multi-party dialogue we may find that different speakers are more active during the discussion of different topics, or that people are more likely to look at particular relevant objects or make characteristic gestures (see e.g. Eisenstein et al. 2008).

There are various ways we might be able to exploit this. We can use a discriminative approach: use a suitable similarity metric to measure the difference between neighbouring sections of the discourse directly, and hypothesize boundaries where this indicates large (enough) differences (Hearst 1997). We could use clustering: group together neighbouring sentences which appear very similar to each other until we build up a set of topic clusters which cover the whole discourse (Reynar 1994). We can use a generative approach: estimate language models for topics, and hypothesize boundaries by finding the most likely sequence of topic states to generate the observed discourse (Yamron et al. 1998). But all use the same basic insight: that topics are associated with content and therefore characterized by a particular set of words, concepts and referents.

### *1.2.2 Distinctive Boundary Features*

The second basic insight is that boundaries between topics have their own characteristic features, independent of the subject matter. When switching from one topic to another, we tend to signal this to our audience in various ways. Firstly, there are various cue words and phrases (*discourse markers*) that directly provide clues about discourse structure (Grosz and Sidner 1986; Hirschberg and Litman 1993), and we can signal the end of one topic, or the beginning of another, by words like *Okay*, *Anyway*, *So* or *Now*. In certain domains there can be more specific cues: formal meeting proceedings often see mention of *the next item on the agenda*, and news broadcasts see reporters sign off at the end of their reports with their name and network identifier (Beeferman et al. 1999).

There can also be cues in the prosodic features of the speech (Hirschberg and Nakatani 1998, 1996; Passonneau and Litman 1997). Just before moving to a new segment, it's common to pause for longer than usual. When starting a new segment, speakers then tend to speed up, speak louder and pause less. Non-linguistic features can be useful here too: topic changes may correspond to changes in physical posture of speaker or audience (Cassell et al. 2001), or perhaps the introduction of new documents onto a meeting table.

The features that are most indicative of topic change will often depend on the nature of the data: the domain, broadcast medium and the number of participants. But once these features have been identified (using manual or standard automatic feature extraction methods), they can be used to help segment the dialogue, either by inclusion in a discriminative classifier (e.g. Galley et al. 2003) or as observed variables in a generative model associated with a change in topic state (e.g. Dowman et al. 2008).

### 1.2.3 Monologue

Automatic topic segmentation is less important in written language: text documents have their own structure, marked more-or-less explicitly (Power et al. 2003), and we are as likely to want to segment a document on the basis of its existing sections or chapters, or a news story by its existing paragraphs, as we are to try to find our own independent segmentation. There are certainly uses for automatic text segmentation – Hearst (1997) argues that breaking up long unstructured paragraphs can aid retrieval and summarization, and Barzilay and Lee (2004) use segmentation as the basis for automatic text generation – but it is really when we look at transcripts of spoken language that segmentation becomes important.

The first serious efforts at topic segmentation were made on monologue, for example stories told by individuals (Passonneau and Litman 1997) or transcripts of news broadcasts (Allan et al. 1998). Before substantial collections of audio transcripts were available, some work used simulated corpora, built by concatenating written texts without their structure (e.g. Reynar 1994, with Wall Street Journal articles) – but the intention was to simulate a particular kind of monologue in both content and structure.

Moving to spoken language must introduce speech recognition errors, of course; and given the heavily lexical nature of the basic approaches outlined above, we can see that high error rates might have quite serious effects. Being able to exploit non-lexical features such as prosody, or even non-audio features such as video scene changes or interactional changes can therefore be important. However, monologue data does have the major advantage of being (usually) well-structured: we might expect the breaks between news stories, for example, to be fairly clear.

### 1.2.4 Dialogue

Dialogue (between two or many people),[1] though, can be a trickier problem. Face-to-face human dialogue can be much harder to segment accurately than monologue data, even for humans – dialogue in informal settings, in particular, typically flows much more smoothly, with discussion often moving naturally from one subject to another without a clear break, and is much less well-structured, with topics being revisited or interleaved. As a result of this, and of the less controlled physical and audio environment that dialogues often occur in, speech recognition error rates also tend to be significantly higher.

Some genres of dialogue lend themselves better to analysis than others, though, so it is the more formal genres such as business meetings that have received most attention. Here, the structure of the discussion tends to be more constrained: a meeting may have an agenda item list at the start which drives the topic sequence. Information independent from the audio stream may also be available, too: agenda-related documents to initialise language models, observable topic-related behaviour such as note-taking, and perhaps even a set of minutes at the end.

[1] It's a common misconception that the word *dialogue* refers only to interaction between two people, and terms like *multilogue* have been proposed to cover cases with more than two. In fact, the prefix in *dialogue* is not *di-* (meaning *two*), but *dia-* (meaning *across* or *through*). While the distinction between two-party and multi-party dialogue can be very useful in some contexts, we intend the term *dialogue* to cover both here.

## 1.3 Applications and Benchmark Datasets

Clearly, this is only a useful task when applied to recordings of some length – short segments of speech such as an utterance in a typical spoken dialogue system tend already to be topically homogeneous and thus not to require segmentation. As a result, it only started to receive attention once long recordings became available.

### *1.3.1 Monologue*

#### Broadcast News

The DARPA Topic Detection and Tracking (TDT) project (TDT, Allan et al. 1998; Doddington 1998, etc.) started much of the work in topic segmentation which forms the basis of methods still used today, both for the task itself and its evaluation. The project focussed on radio and TV news broadcasts, as well as text news stories from newswire and web sources. The project involved much more than just segmentation: the overarching idea was to produce methods to identify, cluster, track and link topics, thus enabling and improving access to news stories via improved browsing and search. For the spoken rather than written part of the data (i.e. the TV and radio broadcasts), though, segmentation becomes a necessary first step before topic identification and other deeper annotation become possible.

The datasets produced are large, include manual topic segment annotations, and are available via the Linguistic Data Consortium.[2] Both the TDT2 and TDT3 collections include audio recordings of over 600 hours, in English and Chinese – see the next chapter for more details.

#### Lectures and Testimonies

Other monologue domains are also good candidates for automatic segmentation: one is university lectures, which usually consist of long recordings with a distinct topical structure. There is interest in making lectures available to students for real-time transcription or offline browsing: segmentation would aid search and improve the ease of access. MIT has set up a Lecture Browser project to work towards this, which has produced a large dataset and investigated methods for segmentation (Glass et al. 2007).[3] The European CHIL and LECTRA projects have also produced lecture sets and systems for segmentation (Fügen et al. 2006; Trancoso et al. 2006).[4]

The Shoah Foundation Institute have also built up a large dataset of 120,000 hours of largely monologue spoken testimonies from Holocaust survivors.[5] Such a large dataset requires segmentation and topic identification to make search practical, but carries its own challenges (Franz et al. 2003; Oard and Leuski 2003).

[2] See `http://projects.ldc.upenn.edu/TDT/`.
[3] See `http://web.sls.csail.mit.edu/lectures/`.
[4] See `http://chil.server.de/` and `http://www.l2f.inesc-id.pt/imt/lectra/`.
[5] See `http://college.usc.edu/vhi/`.

### *1.3.2 Dialogue*

#### Meeting Understanding

While two-person dialogue systems have been the subject of a huge amount of research (not to mention commercial interest), the dialogue they usually involve does not immediately lend itself to segmentation of this kind. Utterances are usually short, and topics are often coherent throughout dialogues and limited to a particular task (call routing, ticket booking and so on). However, in multi-party dialogue the situation becomes very different, and one case in point is business meetings: they can be long, involve several topics, and require indexing by topic segment to allow a record to be usefully browsed or searched afterwards. User studies show that people would like a meeting browser to help with general questions like *"What was discussed at the meeting?"*, as well as more specific ones such as *"What did X say about topic Y?"* (Banerjee et al. 2005; Lisowska 2003).

Two major collections of meeting data have been produced in recent years. The ICSI Meeting Corpus (Janin et al. 2003) includes 75 recorded and transcribed meetings – all real research group meetings – and is available via the LDC; topic segmentation annotations are available separately (Galley et al. 2003; Gruenstein et al. 2008).[6] The AMI Corpus (McCowan et al. 2005) contains 100 hours of recorded and transcribed meetings, including video as well as audio; most of the meetings involve actors playing a given scenario resembling a product design process. This includes topic segmentation annotations as part of the general release (Hsueh and Moore 2006).[7]

#### Tutorial Dialogues

There has been less work in topic segmentation for two-person dialogue, as discussed above; but some domains involve longer and more varied conversations. One such is tutorial dialogue, which can involve two-way dialogue which progresses between related topics; some small datasets have been produced and investigated (see e.g. Olney and Cai 2005).

## 1.4 Evaluation Metrics

The nature of segmentation as a task means that the standard evaluation metrics one might use in classification tasks aren't always suitable. In this section, we see why that is, and examine some alternatives that have been proposed.

### *1.4.1 Classification-Based*

In the majority of classification tasks, evaluation metrics generally start by comparing each instance in the classifier output to a gold standard to determine whether it is correct or incorrect, and counting up the scores. From this we can determine a raw accuracy figure, or more advanced measures such as precision, recall or $F$-score. We could apply the same approach here simply by considering each potential boundary placement as an instance.

As potential boundaries, we can take sentence (or dialogue act) ends, and assume that whatever classifier we use tells us for each candidate whether it is a boundary (a transition

[6] See `http://www.icsi.berkeley.edu/Speech/mr/`.
[7] See `http://corpus.amiproject.org/`.

from one topic segment to another) or a non-boundary (just a transition between sentences within the same segment). A hypothesized boundary in the same place as a true boundary scores a hit; as does a hypothesized non-boundary in the same place as a true non-boundary. Hypothesized boundaries where there is no true boundary are false positives; hypothesized non-boundaries where there is a true boundary are false negatives. This way, we can calculate the standard error metrics. (Automatic sentence segmentation errors may mean that we need to align the output transcript with a true gold-standard transcript first, but we'll ignore that complication here). Figure 1.1 shows an example, representing boundaries by 1 and non-boundaries by -:

| sentences | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ | $S_6$ | $S_7$ | $S_8$ | $S_9$ | $S_{10}$ | $S_{11}$ | $S_{12}$ | $S_{13}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| actual | - | 1 | - | - | - | 1 | - | - | - | - | 1 | - | 1 |
| predicted | 1 | - | - | - | 1 | 1 | - | - | 1 | - | - | 1 | 1 |
| correct? | N | N | Y | Y | N | Y | Y | Y | N | Y | N | N | Y |

**Figure 1.1** An example binary classification evaluation

Early work in topic segmentation used exactly this approach; Reynar (1994), for example, evaluates his approach in terms of recall and precision. But we can see a problem (see e.g. Beeferman et al. 1999; Passonneau and Litman 1996) if we compare the outputs of two imaginary systems, one which is pretty terrible, and one which always gets quite close – see Figure 1.2.

| sentences | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ | $S_6$ | $S_7$ | $S_8$ | $S_9$ | $S_{10}$ | $S_{11}$ | $S_{12}$ | $S_{13}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| actual | - | - | - | 1 | - | - | - | - | - | - | 1 | - | - |
| system 1 | 1 | - | - | - | - | - | 1 | - | - | - | - | - | - |
| system 2 | - | - | 1 | - | - | - | - | - | - | 1 | - | - | - |

**Figure 1.2** Evaluating two very different systems

An evaluation on this basis will score both of them the same: both have 0% accuracy, as neither get any hits. In a sentence segmentation task, this might not matter: the segments are quite short, and a hypothesized sentence with "close" boundaries might be just as useless to a parser as one whose boundaries are completely wrong. But with topic segmentation, we'd really like to score system 2 higher than system 1: each of its boundaries is only 1 sentence away from a true boundary, and its output would be quite helpful. As Beeferman et al. (1999) put it:

> In almost any conceivable application, a segmenting tool that consistently comes close—off by a sentence, say—is preferable to one that places boundaries willy-nilly. [...] It is natural to expect that in a segmenter, close should count for something.

One simple way round this problem might be to allow hypothesized boundaries to score a hit if they are "close to" true boundaries, rather than requiring that they be in exactly the same place. Reynar (1994) does exactly this, giving an alternative evaluation figures which allow matches within a 3-sentence window. The choice of window is arbitrary, of course, and might depend on the data and application of interest; but more seriously, this essentially still suffers from exactly the same problem – a hypothesized boundary just outside the "close" window will score just as badly as one further away. It also fails to distinguish between a perfect segmenter and one which always gets close. In general, then, we need a different approach.

### *1.4.2 Segmentation-Based*

**Beeferman et al. (1999)'s $P_k$**

To combat this problem, Beeferman et al. (1999) propose an alternative measure, $P_k$, which expresses a *probability of segmentation error*: the average probability, given two points in the dataset, that the segmenter is incorrect as to whether they are separated by a boundary or not. (Note that as $P_k$ scores are probabilities, they range between 0 and 1, but a higher score means a *less* accurate segmenter: a higher probability of error).

To calculate $P_k$, we take a window of fixed width $k$ and move it across the dataset, at each step examining whether the hypothesized segmentation is correct about the separation (or not) of the two ends of the window. For a single window position with start $i$ and end $j$, we can express this separation via the indicator function $\delta_S(i,j)$:

$$\delta_S(i,j) = \begin{cases} 1 & \text{if segmentation } S \text{ assigns } i \text{ and } j \text{ to the same segment} \\ 0 & \text{otherwise} \end{cases}$$

For a single window $(i,j)$, the correctness of a hypothesized segmentation $H$ relative to a reference segmentation $R$ can then be calculated as:

$$\delta_H(i,j) \overline{\oplus} \delta_R(i,j)$$

where $\overline{\oplus}$ is the XNOR "both or neither" operator. This evaluates to 1 only if both sides equal 0 or both equal 1, and thus only if segmentations $H$ and $R$ agree about the separation of $i$ and $j$. The inverse of this gives us our basic error function, giving 1 only if $H$ and $R$ *dis*agree:

$$1 - \delta_H(i,j) \overline{\oplus} \delta_R(i,j)$$

or equivalently:

$$\delta_H(i,j) \oplus \delta_R(i,j)$$

where $\oplus$ is the XOR operator. We can then obtain $P_k$ by moving the window across the entire dataset, summing this score, and dividing by the number of windows:

$$P_k = \frac{\sum_{i=1}^{N-k} \delta_H(i,i+k) \oplus \delta_R(i,i+k)}{(N-k)}$$

The choice of $k$ is, in principle, arbitrary; but is generally set to be half the average segment length in the reference segmentation $R$. This value ensures (under some assumptions) that the four obvious baseline algorithms (hypothesizing no boundaries, boundaries everywhere, evenly-spaced boundaries or randomly-spaced boundaries) all have $P_k = 0.5$. A perfect segmenter will score 0, of course; a score of 1.0 will only be achieved by a truly terrible segmenter which manages to hypothesize boundaries in all and only the wrong places.

It is helpful to look at the calculation of $P_k$ in a slightly different way, which helps us see how it relates to other possible measures, and when it might leave something to be desired. Figure 1.3 shows examples of the four possible situations when comparing segmentations in a fixed-width window:

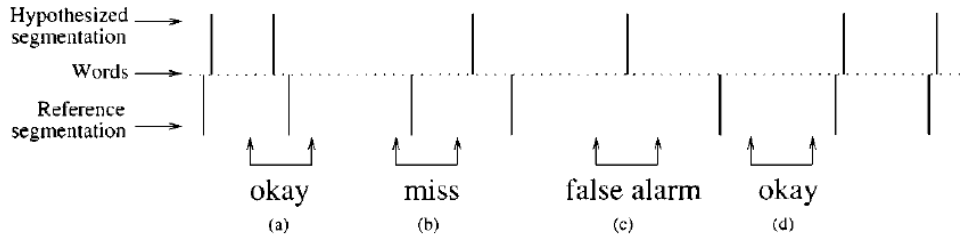


**Figure 1.3** Evaluating a hypothesized segmentation against a reference segmentation. From (Beeferman et al. 1999). ©1999 Kluwer Academic Publishers. Included here by permission.

Windows (a) and (d) are both "correct": the hypothesized and reference segmentations either both show a boundary in the window or both show none. Window (b) shows a *false negative* or *miss* – the hypothesized output has failed to spot a reference boundary; window (c) shows a *false positive* or *false alarm* – the segmenter has hypothesized a boundary where none really exists. Distinguishing these two situations can be helpful in evaluating the suitability of a segmenter for a particular application (as with precision and recall for standard classification tasks). We can do this by decomposing the calculation of $P_k$ into two parts, one expressing the probability of misses, and the probability of false alarms, using the general framework used in the description of the evaluation in the original TDT program (Allan et al. 1998):

$$P_k = P_{Miss} + P_{FalseAlarm}$$

where

$$P_{Miss} = \frac{\sum_{i=1}^{N-k} \delta_H(i, i+k) \cdot (1 - \delta_R(i, i+k))}{\sum_{i=1}^{N-k} (1 - \delta_R(i, i+k))}$$

$$P_{FalseAlarm} = \frac{\sum_{i=1}^{N-k} (1 - \delta_H(i, i+k)) \cdot \delta_R(i, i+k)}{\sum_{i=1}^{N-k} \delta_R(i, i+k)}$$

### Pevzner and Hearst (2002)'s $WD$

$P_k$ clearly gives us a more suitable measure than a simple accuracy or $F$-score, and is still perhaps the most widely-used metric for segmentation evaluation. However, Pevzner and Hearst (2002) point out that it has several shortcomings, all of which essentially stem from the fact that the underlying question it poses of the segmentations being compared is whether they agree on *whether two points are separated or not*, rather on *how many boundaries lie between them*. As Figure 1.4 shows, this can lead to situations where $P_k$ fails to penalize false alarms:

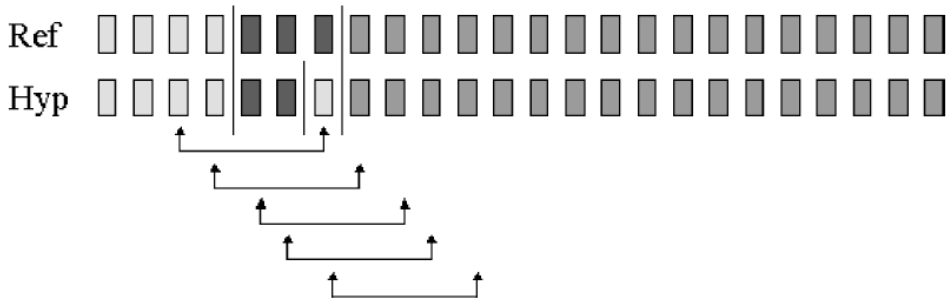


**Figure 1.4** $P_k$ can fail to penalize false alarms that fall within a window width $k$ of a true boundary. From (Pevzner and Hearst 2002).

Here, the false hypothesized boundary falls within the window width $k$ of a true reference boundary, and the $P_k$ evaluation will rate all the windows shown as "correct" in that both segmentations agree that the two ends of the windows fall into different segments. Misses (false negatives) don't suffer from the same problem, though, with the result that $P_k$ can effectively penalize them more than false alarms.

Instead, they propose a measure called WindowDiff ($WD$), which works in a similar way by moving a fixed-width window across the data; this time, though, windows are scored as "correct" if they assign the same number of segment boundaries between their start and end. If $b_S(i,j)$ is the number of boundaries between $i$ and $j$ according to segmentation $S$, the basic error function for a window becomes:

$$|b_H(i,j) - b_R(i,j)| > 0$$

Summing over windows and normalizing as before, we now obtain:

$$WD = \frac{\sum_{i=1}^{N-k}[|b(i,i+k) - b_R(i,i+k)| > 0]}{(N-k)}$$

Again, $WD$ is a measure of segmentation error - lower scores mean less error, with a perfect segmenter scoring 0. And again, if we want to examine issues of precision and recall independently, we can express it in terms of scores for misses and false alarms:

$$WD = WD_{Miss} + WD_{FalseAlarm}$$

where

$$WD_{Miss} = \frac{\sum_{i=1}^{N-k}[b_H(i,i+k) < b_R(i,i+k)]}{(N-k)}$$
$$WD_{FalseAlarm} = \frac{\sum_{i=1}^{N-k}[b_H(i,i+k) > b_R(i,i+k)]}{(N-k)}$$

**Georgescul et al. (2006a)'s $Pr_{error}$**

While $WD$ does solve many of $P_k$'s problems, it has its own problems; not least of which, it's hard to know exactly what either of them really *mean* intuitively (other than in terms of direct comparison to another system). Most recent work in topic segmentation uses both metrics when reporting performance.

Recently, Georgescul et al. (2006a) have pointed out another problem with $WD$: that it effectively assigns a lower penalty to misses than to false alarms. Looking at the formulation for $WD_{Miss}$ and $WD_{FalseAlarm}$ above, we can see that both are normalized by the number of windows $(N-k)$. While this seems correct for false alarms (there are as many opportunities for a false alarm as there are evaluation windows), it doesn't for misses. If we want to evaluate on the basis of the true probability of a miss, we must normalize the number of misses by the number of opportunities for a miss – in other words, the number of windows in which there is a boundary in the reference segmentation. They therefore propose a modified normalization for the $Miss$ term:

$$Pr_{Miss} = \frac{\sum_{i=1}^{N-k}[b_H(i,i+k) < b_R(i,i+k)]}{\sum_{i=1}^{N-k}[b_R(i,i+k) > 0]}$$
$$Pr_{FalseAlarm} = \frac{\sum_{i=1}^{N-k}[b_H(i,i+k) > b_R(i,i+k)]}{(N-k)}$$

The two terms can then be combined to give an overall error metric $Pr_{error}$. Georgescul et al. (2006a) propose that this term be weighted to allow a trade-off between the penalties for misses and false alarms, depending on the application being considered:

$$Pr_{error} = C_{Miss} \cdot Pr_{Miss} + C_{FalseAlarm} \cdot Pr_{FalseAlarm}$$

where $0 \leq C_{Miss} \leq 1$ is the cost of a miss, and $0 \leq C_{FalseAlarm} \leq 1$ is the cost of a false alarm. Setting $C_{Miss} = C_{FalseAlarm} = 0.5$ will assign equal costs, and ensure that the trivial no/all boundary baselines both get $Pr_{error}$ around 50%. This proposal hasn't seen much take-up yet, but does seem to promise an improved metric.

### *1.4.3 Content-Based*

This directly segmentation-based approach to evaluation has become the accepted standard. It is worth noting here, though, that evaluating purely on the basis of the accuracy of boundary placement may have its drawbacks. Firstly, as ASR and automatic sentence segmentation will be errorful, the exact units (or time periods) over which to calculate the evaluation functions can be unclear.

Secondly, and perhaps more importantly, this approach takes no notice of the content of the topics themselves. Failure to detect a boundary between two very similar topics perhaps ought to be penalized less than failure to detect one between two very different topics. It may be, then, that error metrics which combine measures of segmentation accuracy with measures of topic similarity can give us a more useful tool – see (Mohri et al. 2009) for a suggestion along these lines. However, the suitability of any one method may well depend on the application in mind, and the purpose to which the derived topics are to be put. The next chapter will discuss topic classification, and suitable evaluation methods for that task; we should remember that segmentation and classification are to a large degree joint problems, and that evaluating one alone may not tell the whole story.

## 1.5 Technical Approaches

### *1.5.1 Changes in Lexical Similarity*

Some of the first successful approaches to segmentation focus on changes in lexical distribution, and this still forms the core of many current algorithms. The essential insight is that topic shifts tend to be marked by a change in the vocabulary used, which can be detected by looking for minima in some lexical cohesion metric.

#### TextTiling (Hearst 1997)

*TextTiling* (Hearst 1997; Hearst and Plaunt 1993; Hearst 1994) was one of the early algorithms to emerge from the TDT Broadcast News effort, and still forms the baseline for many recent improvements; while designed for text documents, it has since been successfully applied to spoken data. The discourse is tokenized, stemmed and divided into windows of a fixed width. Each window is represented by a lexical frequency vector: one row per distinct word type, whose value is the raw frequency of that word type in the window. Moving across the discourse, the lexical similarity is then calculated for each pair of adjacent windows, using the cosine distance between their lexical frequency vectors. The resulting curve is then smoothed, and local minima are found by calculating a *depth score* for each point based on its relative depth below its nearest peaks on either side. Points with the highest depth scores (i.e. the deepest troughs in the smoothed similarity curve) are then taken as the hypothesized boundaries – see Figure 1.5.

Other similarity metrics can be used within the same overall approach: Hearst also proposes an alternative which uses a measure of introduction of new vocabulary within each block (new topics may be expected to introduce new terms); and Reynar (1999) uses the overlap between the two sets of bigrams rather than just unigrams. Whichever exact variant is used, one advantage of this approach is that it is essentially unsupervised – although various

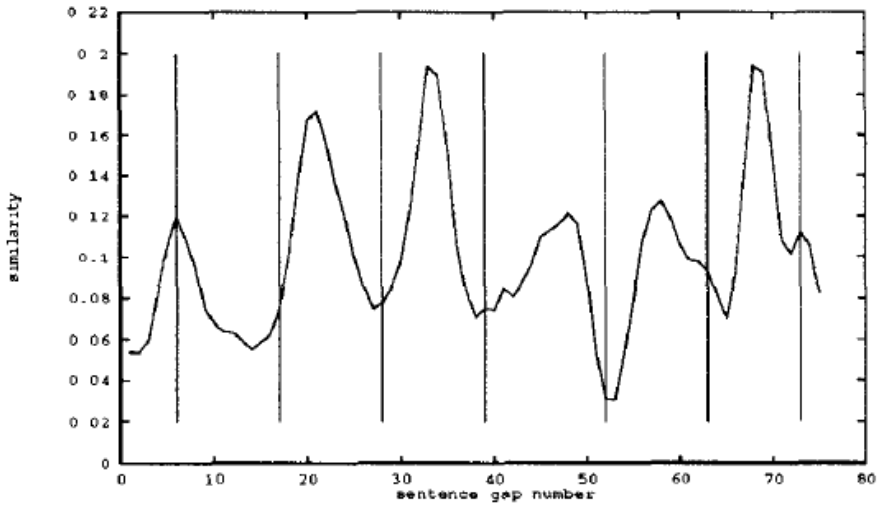


**Figure 1.5** True segment boundaries vs. minima in TextTiling's lexical similarity metric. From (Hearst and Plaunt 1993). ©1993 ACM, Inc. Included here by permission.

parameters do need to be set suitably, such as the window width and the cut-off at which depth scores cause boundaries to be hypothesized.

### Latent Concept Modelling

Of course, the success of this approach must depend on the suitability of the similarity metric chosen. Using raw lexical frequency vectors as the basis for similarity can cause problems, due to their sparseness and the fact that they necessarily treat words independently, ignoring the real dependencies between related terms. One way to combat this is to project the lexical vectors into some *latent concept* space using methods such as Latent Semantic Analysis (LSA, Landauer et al. 1998).

More details of LSA and related approaches are given in the next chapter, but the main idea is to represent topics using fewer dimensions: characterizing a segment not by counts of each distinct word type, but by weights over a smaller set of latent variables (which can be seen as semantic concepts). The simple lexical frequency approach as used by TextTiling in its original form represents any segment of text (window, topic, document) as a vector $x$ of word frequencies, with one entry for each term in the vocabulary $W$ of size $w$:

$$x = (f_1, f_2, \ldots, f_w)$$

Instead, LSA generates a set of latent concepts by matrix decomposition. Given a set of documents (or topic segments) $D$ of size $d$, we can build a lexical frequency matrix $X$ where each column is the word frequency vector for a document:

$$X = (x_1; x_2; \ldots; x_d)$$

Singular value decomposition allows us to rewrite $X$ as the product of three matrices: $U$ and $V$ are orthogonal matrices of eigenvectors (of dimension $w \times w$ and $d \times d$ respectively), and $\Sigma$ a diagonal matrix representing the corresponding eigenvalues. The eigenvectors of $V$ can now be viewed as the latent concepts:

$$X = U\Sigma V^*$$

This decomposition is exact; but by limiting $\Sigma$ to the largest $k$ values, we can closely approximate the original $X$ while reducing the effective number of dimensions to $k$. We can now project a $w$-dimensional word frequency vector $x$ into the $k$-dimensional latent concept space using (appropriately truncated) $U$ and $\Sigma$:

$$z = \Sigma^{-1} U^* x$$

The vector $z$ is now a representation of a text segment as a set of weights over the $k$ latent concepts. Similarity (or distance) between segments or windows can now be calculated using an appropriate vector distance measure (such as the cosine distance) between these $z$ vectors. This approach does require a dataset on which to learn the concept vectors; but gives a more general, less sparse, and lower-dimensional representation which allows dependencies between words to be expressed (as each concept vector may relate several distinct words).

Olney and Cai (2005) use LSA to provide a similarity metric within a TextTiling-like approach, and show that it can give more accurate segmentation on dialogue data. Their method uses LSA-based distance to compare each utterance with the windows on either side, not only in terms of its similarity (in their terms, its *relevance* to the surrounding topic), but also its difference (its *informativity*, or the amount of new information it may be providing) – and hypothesize boundaries on the basis of a combination of these factors, learnt using a regression model. Other latent concept modelling approaches are also possible – for example, Sun et al. (2008) use Latent Dirichlet Allocation (see below) to provide the basis for their similarity metric.

#### LCSeg (Galley et al. 2003)

Another variation of the basic lexical cohesion approach that has been particularly influential in dialogue segmentation is LCSeg (Galley et al. 2003). Here, the similarity metric uses the presence of *lexical chains* (Morris and Hirst 1991) – implemented here as simple term repetitions – rather than just the presence of words; the insight being that these chains will start and end at topic shifts. Chains are identified for all repeated terms in the text, and weighted according to their term frequency (more frequent terms being weighted higher) and the chain length (shorter chains being weighted higher). The cosine distance between each pair of windows' lexical chain vectors is then used as the key metric, and again the sharpest local minima are taken as the hypothesized boundaries; this simple but robust method has shown good performance on difficult data (multi-party meetings).

#### Supervised Classification (Georgescul et al. 2006b)

If suitable training data is available, the same insight can be given higher accuracy by the use of supervised classification, and Galley et al. (2003) showed that this could improve their

algorithm's performance. Georgescul et al. (2006b) went one step further by characterizing each potential boundary point in the discourse (each utterance boundary) not by a single lexical cohesion score comparing the windows on either side, but an array of lexical similarity features, one for each word in the discourse vocabulary. This results in a very sparse, high-dimensional array of features, but one which contains a large amount of information. By using support vector machines, which can operate with high dimensional feature spaces (Vapnik 1995), a classifier model can then be learnt which predicts boundaries with high accuracy, outperforming (Galley et al. 2003)'s approach on meeting transcripts. Further performance improvements are also possible by incorporating non-lexical features and latent concept representations – see below. Being a supervised method, though, it does require annotated training data.

### *1.5.2 Similarity-based Clustering*

Alternatively, we can take another viewpoint on the same basic insight: rather than looking for areas of low cohesion (the boundaries), we can look for areas of high cohesion (the topic segments). Clustering together neighbouring areas which are similar to each other leaves us with a segmentation of the discourse. This can be approached using *agglomerative* clustering (growing clusters outwards from peaks in similarity (Yaari 1997)), but *divisive* clustering has proved more effective.

#### Dot-Plotting (Reynar 1994)

Reynar (1994) uses the technique of *dot-plotting*, originally from (Church 1993), to segment text: the discourse is plotted as a two-dimensional matrix with its words (in linear order) along both axes, and a non-zero entry (a dot) wherever words match (see Figure 1.6). The diagonal is, of course, entirely non-zero, as each word matches itself; but squares can also be seen, corresponding to topics, in areas with more frequent matching between near-neighbours. The boundaries between these squares are the topic boundaries, and the best set of boundaries is that which maximizes the dot density within the squares it delineates, and minimizes the density outside those squares (i.e. finds topics which are maximally similar internally, and maximally different from other topics). This can be performed essentially unsupervised and without any training data, although it needs a search algorithm and some criterion for finishing; Reynar (1994) uses a best-first search algorithm which minimizes the outside density and assumes a known number of boundaries.

This method has been extended and improved since: Choi (2000)'s *C99* algorithm, for example, works on sentences rather than words, with a cosine distance sentence similarity metric and a gradient criterion for finishing. Latent concept modelling may be applied to the similarity metric here too: Choi et al. (2001) showed that using a Latent Semantic Analysis-based metric could improve accuracy, although Popescu-Belis et al. (2005) found that the benefit was small for meeting dialogue data.

Of course, these techniques were originally developed and evaluated on text, and on corpora created artificially by concatenating distinct text documents; as Figure 1.6 shows, speech data is less cleanly separable, with smoother transitions between topics, and similarities between temporally distant topics. However, Malioutov and Barzilay (2006) have shown that the approach can be successfully applied to spoken discourse with some

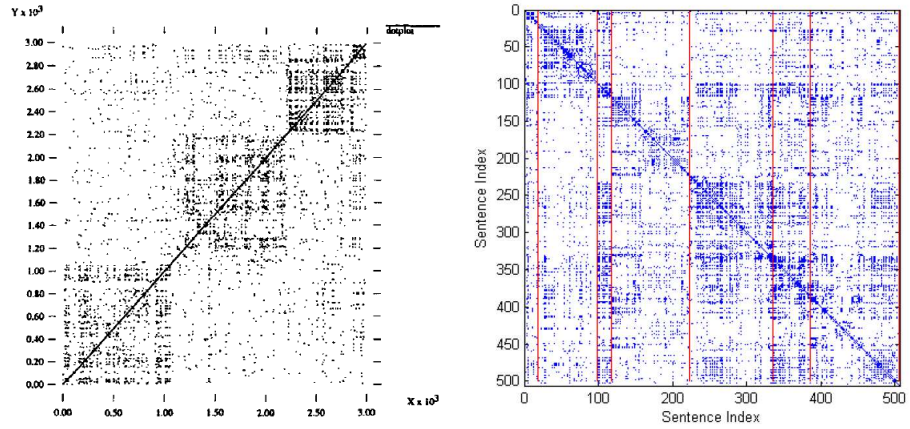


**Figure 1.6** Dotplotting Wall Street Journal articles (Reynar 1994) and a spoken lecture (Malioutov and Barzilay 2006).

modification, by formulating an algorithm to find an exact solution, using a suitable similarity metric, and limiting the long-range distance over which similarity is calculated.

### *1.5.3 Generative Models*

Yet another way of exploiting the same phenomenon is to take a generative perspective. We can model discourse as being generated, via a noisy channel, from some underlying sequence of topics, each of which has its own characteristic word distribution. When the topic changes, the vocabulary used will change; so if we can infer the most likely sequence of topics from the observed words, we can derive the positions of the boundaries between them. Note that this approach does not require us to measure the similarity between utterances or windows directly – rather, the fact that neighbouring utterances within the same topic segment are similar to each other is implicit in the fact that they have been generated from the same topic.

#### Hidden Markov Models (Mulbregt et al. 1999; Yamron et al. 1998)

This is, of course, comparable to the problem of speech recognition (ASR), where the task is to infer the most likely sequence of phonemes from the observed acoustic signal. If we can make similar assumptions about the dependencies between words and topics as ASR does about phonemes and acoustic signals, we can apply similar models, together with their well-researched inference techniques.

The most commonly used generative model in ASR is the hidden Markov model (HMM), and Figure 1.7(a) shows a HMM can be used as a simple topic model. This model assumes that the discourse is composed of a linear sequence of segments of length $L$ words, each of which is associated with a topic state $z$ from which the words $w$ are generated with probability $p(w|z)$. If we can infer the most likely sequence of topic states given the observed words, we can produce a segmentation: if the topic $z_t$ at segment $t$ is different from the topic $z_{t+1}$ at

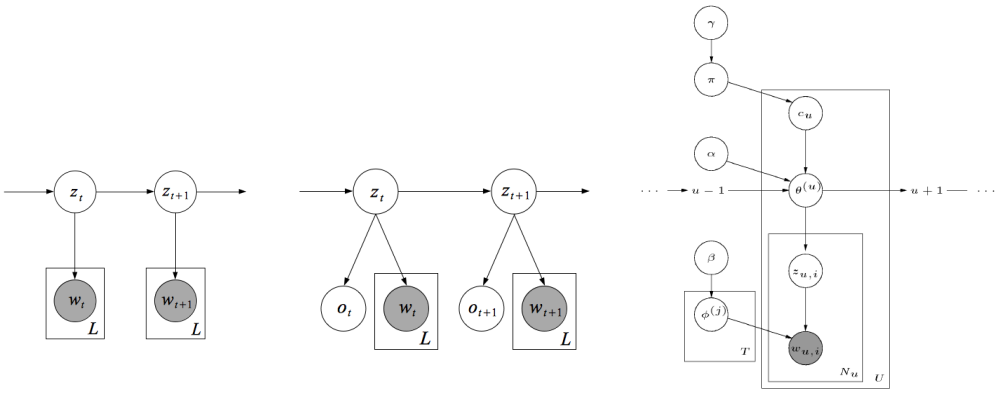


**Figure 1.7** From left to right: (a) a simple HMM topic model (Yamron et al. 1998); (b) the aspect model used in (Blei and Moreno 2001); and (c) the topic mixture model used in (Purver et al. 2006).

segment $t + 1$, we hypothesize a topic boundary between the segments. Performing inference in such a model depends on two major assumptions:

1. the probability of a word being generated depends only on the current topic (the *emission* probability $p(w|z)$);
2. the probability of a topic being discussed depends only on the previous topic (the *transition* probability $p(z_{t+1}|z_t)$).

Given estimates of the emission and transition probabilities, we can calculate the prior probability $p(Z)$ of any topic sequence $Z$, and the posterior probability $p(W|Z)$ of $Z$ generating an observed word sequence $W$. Via Bayes' rule, this allows us to calculate the probability of $Z$ given $W$:

$$p(Z|W) = p(W|Z).p(Z)/p(W)$$

and maximizing this will give us the most likely topic sequence $Z$ (given that $W$, and thus the prior probability $p(W)$, is fixed). While these assumptions may be good first approximations, note that they do not entirely hold in reality: while some words may be related only to the content of the topic at hand, many are related more to discourse or syntactic function; and the likely sequence of topics may depend on many extra-linguistic factors. However, these models are well understood, and efficient algorithms exist for learning and decoding (see e.g. Jurafsky and Martin 2009; Manning and Schütze 1999, for details).

Yamron et al. (1998) were among the first to show that they can be applied to broadcast news segmentation. First, a set of topics $z$ with their associated language models $p(w|z)$ are learnt, by clustering a set of training texts using an appropriate lexical similarity metric, and estimating the word probability distribution for each cluster (Yamron et al. used 100 topics and smoothed unigram language models). Rather than learning a full set of transition probabilities between the given topics, this is simplified to allow for a single probability of changing topic (as opposed to keeping the same topic when changing HMM state), which can be estimated based only on the average length of segments in the training data. (While

learning transition likelihoods between particular topics might be of use in some domains, in many (such as broadcast news), topics can occur in any order.) The standard Viterbi algorithm can then be used to infer likely topic state sequences, and hence segmentations.

As mentioned above, this kind of approach does not require a similarity metric, as pairwise similarity measurement between sentences or windows is not required. It does require a segmented training dataset to estimate the topic transition probability, and the topic language models – although given the dataset these are learned unsupervised via clustering.

### Latent Concept Modelling (Blei and Moreno 2001)

One drawback of this simple approach is that it treats words as being independent of each other given a topic; as discussed with relation to lexical similarity metrics in the previous sections, this is not a realistic assumption. With probabilistic models, the assumption becomes more accurate as the segment size ($L$ in the model above) increases – but this leads to reduced accuracy as the segmentation granularity necessarily becomes coarser. As above, one way to alleviate this is to use some form of latent concept modelling.

Blei and Moreno (2001) show that a probabilistic form of latent concept model, Probabilistic Latent Semantic Indexing (PLSI, Hofmann 1999), can be used within a HMM framework to give what they call an *aspect* HMM. PLSI is described in detail in the next chapter, but its essential insight is to associate a document $o$ (in this case, a topic segment) not with one fixed topic $z$ as above, but with a probability distribution over topics $p(z|o)$. The topics can therefore be seen as latent variables, as with LSA, and each topic is associated with a probability distribution over words $p(w|z)$. This allows us to account for segments which may be related to multiple underlying concepts, or which are generated from a "contentful" topic and a "syntactic" topic (see e.g. Griffiths et al. 2005).

Learning the latent topics, and the word and document distributions, is now performed via expectation-maximization (EM, see e.g. Manning and Schütze 1999), rather than matrix decomposition, over a training corpus of segments. Once the topics have been learnt, a HMM can be constructed which uses the learnt latent topics $z$ as the hidden state variables – see Figure 1.7(b). If desired, transition probabilities between the latent topics can also be learnt from the training set, via clustering documents according to their most likely topic. Decoding is less straightforward, as the relevant HMM emission probabilities are no longer just $p(w|z)$ as before, but also $p(o|z)$, which must be estimated during decoding using a version of EM. However, the model can give improved performance (and Blei and Moreno (2001) show this on radio transcripts), especially at lower values of $L$ required for finer segment granularity.

Other forms of latent concept modelling have also been successfully used; one that has recently become more popular is Latent Dirichlet Allocation (LDA, Blei et al. 2003). One advantage of LDA is that it requires less supervision. While PLSI requires a segmented training corpus to provide direct estimates of the probability distributions over topics $p(z)$ and documents $p(o|z)$, LDA takes a fully Bayesian approach: it assumes a range of possible distributions, constrained by being drawn from Dirichlet distributions. This allows a latent topic model to be learnt entirely unsupervised, allowing the model to be maximally relevant to the data being segmented (and less dependent on the domain of the training set and the problems associated with human segmentation annotation). Purver et al. (2006) use this approach in an unsupervised generative model applied to meeting dialogues, and show performance competitive with LCSeg's similarity-based approach. As shown in

Figure 1.7(c), each utterance $u$ is now associated with a *distribution* $\theta$ over possible latent topics $z$, each of which has its own probability distribution over words $\phi$; switching to a new topic segment with probability $c$ means changing the topic distribution $\theta$. Given the assumption that $\theta$, $\phi$ and $c$ are drawn from Dirichlet distributions (or a Beta distribution in the case of $c$) with fixed parameters $\alpha$, $\beta$ and $\gamma$, inference is possible: however, exact solutions are no longer tractable and must be approximated by (computationally intensive) sampling – see (Blei et al. 2003; Griffiths and Steyvers 2004) and the next chapter for details.

**Compact Language Modelling (Utiyama and Isahara 2001)**

Another prominent variant of the generative approach that allows essentially unsupervised segmentation is that of Utiyama and Isahara (2001)'s *TextSeg*. Here, the underlying assumption is similar: that the discourse is generated from a sequence of topics, each of which is associated with its own language model (a probability distribution over words); and the underlying approach to segmentation also involves placing the boundaries so as to maximize the likelihood of the data given these language models. However, in this approach there is no attempt to learn the most likely set of topic models from a training dataset; rather, the most compact set are chosen given the data being segmented.

This is estimated as part of the segmentation process itself: given any hypothesized segmentation (set of boundaries) $S$, language models representing each segment can be calculated from the observed words within that segment, given an appropriately smoothed estimation procedure. The likelihood of the data can then be calculated, as above, as the posterior probability of the words given the segmentation $p(W|S)$, and the prior probability of the segmentation itself $p(S)$ – where they assume that $p(S)$ can be calculated from the average length of segments, either known *a priori* or derived from some training set. By comparing the likelihood of the data calculated from different segmentations, the maximum-probability segmentation can be chosen; and this can be performed via an efficient dynamic programming algorithm.

Of course, estimating the language models only from the data being segmented (rather than a large set of possibly clustered training documents) makes the choice of smoothing essential. Utiyama and Isahara (2001) use a form of Laplacian smoothing; but Eisenstein and Barzilay (2008) have since shown that another way to provide this is to generalize the approach into a fully Bayesian version. Rather than estimating one particular smoothed language model, they use an approach similar to LDA to marginalize over all possible language models, and show that this improves segmentation accuracy.

As well as obviating the need for an explicit fixed similarity metric, and being amenable to unsupervised learning, another potential advantage of generative modelling is the acquisition of the topic models themselves: the ability to characterize the topics in terms of their associated language models can be useful for topic summarization, classification or browsing – see later chapters. Depending on the application, approaches which learn models in latent concept spaces, or models which are common across training and/or test datasets, may be more or less advantageous.

### 1.5.4 Discriminative Boundary Detection

A rather different approach is to look for the characteristic features of the boundaries themselves: the cue phrases people use to signal topic change, the prosodic features often exhibited in speech at the beginnings and ends of topic discussions, or the introduction of new referents.

Passonneau and Litman (1997) showed that all of the above could be useful in segmenting spoken stories. Topic boundaries were correlated with the presence of one of a list of cue phrases from Hirschberg and Litman (1993), such as *So* and *Anyway*, with the presence of significant pauses or sentence-final intonation, and with the absence of noun phrases whose referents were found in, or inferable from, the preceding utterance. These three features seem to be complementary: by combining them in a decision tree, they could produce segmentations with higher accuracy than using any one alone. The intonational information used was hand-coded in that case; but Tür et al. (2001) showed that automatically extracted pitch pattern, pause and vowel information could also be used successfully in broadcast news segmentation (again by combining features in a decision tree classifier).

As well as general cues such as *So*, *Anyway*, different domains often have their own specific cue phrases. In broadcast news, phrases such as *Joining us*, *This just in* and *Welcome back* are strongly indicative of topic shifts. Maybury (1998) describes a system which uses these cues in a finite-state automaton to detect and segment news story structure; and Reynar (1999) found that by including cues together with lexical cohesion measures in a maximum entropy model, accuracy could be increased over using lexical cohesion alone.

Of course, as the most useful set of cue phrases will vary between domains, there might be an advantage to learning that set automatically, rather than having to define it manually. Beeferman et al. (1999) showed how this can be done using log-linear models which combine many possibly dependent features: by starting with many possible features and using an iterative procedure for selecting the most informative features, a suitable subset can be derived. If the initial features considered consist of each possible word in suitable "cue" positions (immediately before or after a potential boundary, in the next sentence, etc.), this will automatically produce an empirically based cue word set. For their broadcast news dataset, they derive domain-related terms such as *Joins*, *Live* and the letters of *CNN*, as well as those more specific to the content of their data such as *Haiti* and *Clinton*.

Dialogue can bring its own distinctive features to topic shifts. It is often the case that different speakers are more active during the discussion of different topics, resulting in an observable change in relative speaker activity at segment boundaries. The early sections of topic discussions also tend to be more ordered, with less overlap and interruption between speakers, as new subjects are introduced and set out. Galley et al. (2003) showed that features based on these observations could be helpful in segmenting ICSI dialogues – see below.

### 1.5.5 Combined Approaches, and the State of the Art

Of course, given these different approaches with their different insights into the problem, one way towards higher accuracy might be to combine them, and this is the direction that many of the most effective recent systems have taken.

### Combining Lexical Cohesion and Boundary Detection

The evidence used by lexical cohesion-based approaches (whether from differential, clustering or generative perspectives) seems entirely complementary to the evidence used by the boundary-detection approach. Combining them therefore seems a natural step.

One way to do this is to combine the outputs of the two different approaches via some suitable classifier. Galley et al. (2003) tried exactly this, using a decision tree classifier whose input features were taken partly from the output of their LCSeg segmenter (Section 1.5.1) – both the raw lexical similarity metric and the smoothed peak hypotheses – and partly from distinctive boundary characteristics. The latter features followed Passonneau and Litman (1997) in including a list of cue phrases, and significant pauses, although they found that pauses should be limited to those which do not occur after a question or in the middle of one participant's speech. They also included measures of *speech overlap* (overlapping utterances tend to be rare at the beginning of topic segments) and *speaker change* (in dialogue, new topics are often associated with changes in the relative activity of the participants), calculating the latter via the change in the distribution across speakers of the number of words uttered immediately before and after a potential boundary. This combination resulted in a large improvement in performance, with $P_k$ improving from 0.32 to 0.23 on the ICSI meeting dialogue data. Arguello and Rosé (2006) used a similar approach within a Naïve Bayes classifier, combining lexical cohesion scores with syntactic features (expressed as part-of-speech bigrams) and information about the identity of the speaker, in two-person tutorial dialogues.

Another way is to include distinctive boundary features directly into the main classifier, alongside the features used to express lexical similarity. For example, in Georgescul et al. (2006b)'s discriminative approach, lexical similarity is encoded as an array of features in an SVM, one for each distinct word type in the vocabulary; new features related to silence, overlaps or cue phrases can be added directly to the feature vectors, and they found that this gave a small improvement (Georgescul et al. 2007). Within a generative approach, boundary features can be added as new observed variables associated with a special topic-change state; Tür et al. (2001) show how to incorporate this within Yamron et al. (1998)'s HMM approach, Dowman et al. (2008) similarly within Purver et al. (2006)'s LDA-based model and unsupervised learning procedure, and Eisenstein and Barzilay (2008) show how to treat cue phrases as generated from boundary states in their language modelling approach.

There is some evidence, though, that this combined approach may become less helpful as the desired granularity of topics becomes finer. Hsueh et al. (2006) examined segmentation of the AMI corpus at relatively coarse- and fine-grained levels, where the coarse-grained level often corresponded to broad changes in the activity or state of the meeting, such as introductions or closing review, while the fine-grained level corresponded to lower-level changes in subject matter. They found that boundary features such as cue phrases, silence and speaker activity were only helpful for the coarse-grained segmentation.

### Combining Generative and Discriminative Approaches

Benefits can also be gained by combining some of the benefits of generative approaches, such as the ability to learn models of (latent) topics, with the accuracy of discriminative approaches. Georgescul et al. (2008) used generative models similar to those of Blei and Moreno (2001) and Purver et al. (2006) described above to learn topic models in latent

concept spaces, via PLSI and LDA respectively. Using these models to provide vector-space representations of windows of discourse, they could then apply the same discriminative SVM classification approach as in (Georgescul et al. 2006b) to hypothesize boundaries, but with a more compact feature representation; as they show, results on the ICSI dialogue data were improved.

Tür et al. (2001) took a slightly different approach, seeking to exploiting the fact that different classification approaches can lend themselves better to different phenomena, with discriminative classifiers often dealing particularly well with prosodic feature data. Their system combined a version of Yamron et al. (1998)'s HMM-based lexical model with a decision tree trained on prosodic boundary features, and they experimented with two ways of achieving this: firstly adding the HMM's output posterior boundary probabilities into an overall decision tree, and secondly using the prosodic decision tree to provide emission probabilities for a boundary state in the HMM. On their broadcast news data, the latter approach was more successful.

### State of the Art Performance

***Monologue***

The TDT dataset allows us to compare the accuracy of various algorithms on broadcast news data. On manual transcripts, the algorithms developed for general text (and initially evaluated using artificial corpora) perform reasonably well, with Choi (2000)'s C99 achieving $P_k$ between 0.21 and Utiyama and Isahara (2001)'s TextSeg 0.14, improving to 0.18 and 0.11 respectively when given knowledge of number of boundaries (see Georgescul et al. 2006a).

The systems which were developed as part of the TDT effort achieve good performance even on ASR output: Beeferman et al. (1999)'s supervised maximum entropy classifier achieved $P_k$ of 0.15, with Yamron et al. (1998)'s HMM method giving 0.16. Since then, Tür et al. (2001)'s method including prosodic features has outperformed those with $P_k = 0.14$; and Beeferman et al. (1999) claim $P_k = 0.08$ on a CNN portion of TDT news data.

For spoken lecture segmentation, Malioutov and Barzilay (2006)'s divisive clustering method achieves $P_k$ of 0.30 on manual transcripts, dropping slightly to 0.32 on ASR output. In comparison, C99 and TextSeg give $P_k$ values between 0.31 and 0.37 on the same data.

***Dialogue***

Multi-party dialogue data is trickier, of course, and accuracies on the ICSI and AMI meeting datasets are correspondingly lower. Approaches developed for text or monologue show only limited accuracy: Georgescul et al. (2006a) tested TextTiling, C99 and TextSeg on the ICSI corpus, achieving $P_k$ results ranging between 0.55 and 0.38, although this improved to 0.35 when supplying information about the expected number of segment boundaries.[8]

LCSeg has shown much better accuracy on ICSI, and has become a common baseline to quote: its unsupervised version achieves $P_k = 0.32$, and the supervised version including boundary features achieves $P_k = 0.23$. Higher unsupervised accuracies have now been achieved by Dowman et al. (2008) and Eisenstein and Barzilay (2008)'s Bayesian generative approaches, with $P_k = 0.26$; and higher supervised accuracies by Georgescul et al. (2007) discriminative SVM classifier, with $P_k = 0.21$. Most dialogue segmentation efforts to date

[8]Note though that Banerjee and Rudnicky (2006) report more success with TextTiling on a different multi-party dataset.

have used manual transcripts, but some results using ASR output are now available with little, if any, reduction in segmentation accuracy (Hsueh et al. 2006; Purver et al. 2006).

Comparing system performance on two-person dialogue is difficult, as this area has received less attention (in terms of topic segmentation) and has little in the way of standard datasets for comparison. However, Arguello and Rosé (2006) give results on two corpora of tutorial dialogues: their supervised classifier, with $P_k$ ranging between 0.10 and 0.40, outperforms Olney and Cai (2005)'s lexical cohesion method with $P_k$ of 0.28 to 0.49.

## 1.6 New Trends and Future Directions

### Multi-Modality

Including multiple sources of information has become common, as explained above: segmentation accuracy can be improved by including not only lexical information (from lexical cohesion or language model probabilities), but also speech signal information (e.g prosody), discourse information (cue phrases) and pragmatic information (speaker activity). Some recent work has gone beyond this to look at information from streams other than speech; what is available depends, of course, on the data and the application at hand.

Where video is available as well as audio, useful visual features may be extracted and used for segmentation. The task of segmenting TV news broadcasts can be aided if scene changes or commercial breaks can be detected: Maybury (1998) used the presence of black screens and logos to aid segmentation, and since then methods have advanced to include face detection and classification of scenes as reports, single or double anchorperson presentation, outdoor shots and so on (see e.g. Avrithis et al. 2000; Chaisorn et al. 2003). In face-to-face dialogue, video information on participant pose and gesture can be helpful. Eisenstein et al. (2008) investigated the use of hand gesture features: as well as showing cohesion of lexical form, coherent topic segments often show cohesion of gestural form, and they incorporate this to help segment the discourse within a Bayesian model. Other modalities have been used too, for example note-taking in the meeting domain. Banerjee and Rudnicky (2007) provided meeting participants with a note-taking tool, and used their interaction with that tool to constrain and improve the output of their TextTiling-based segmenter.

Information external to the discourse itself may also be available, in particular details of the content of the topics likely to be discussed and/or their likely order. For meeting dialogue, this might take the form of pre-defined agenda, something often distributed prior to formal meetings. This can certainly aid segmentation: Banerjee and Rudnicky (2007), for example, also exploit some knowledge of the defined agenda items and their related words. For broadcast monologue, this might take the form of a defined running list, or a model of how content is usually structured in a given domain. Barzilay and Lee (2004) show how to learn such a model, without supervision, for particular text types such as earthquake and accident reports, and use this to segment text for summarization purposes.

### Topic Identification and Adaptation

In many applications, topic segmentation is a first step before topic *identification*: classifying or clustering the actual topics discussed within each segment (see the following chapter). As mentioned above, this can be one advantage of the use of generative models, as they effectively treat segmentation and identification as joint problems: as well as producing

a segmentation, they derive models of the topics themselves, in terms either of language models (probability distributions over words) or the equivalent in some latent concept space (probability distributions over word vectors). These models can then be used to characterize the topics themselves, extract lists of descriptive keywords or word clouds to present to a user, or cluster related topics in different broadcasts.

However, as discussed at the start of this chapter, the conception of topic – and therefore the segmentation associated with it – can vary depending on the application at hand, the domain and even the interests and intentions of the user. A possible solution to this problem might be to use an *adaptive* approach to segmentation, allowing the segmentation (and the associated topics) to change as indicated by the user's behaviour or the emerging dataset. One way to approach this is via unsupervised methods which learn underlying topic models from entire datasets, such as the Bayesian approaches of Dowman et al. (2008) and Eisenstein and Barzilay (2008), for example. As more data is added to a user's personal dataset (as they browse new news broadcasts in which they are interested, or attend new relevant business meetings), the topic models learnt and the corresponding segmentation will change – and this can be achieved online using suitable algorithms (AlSumait et al. 2008).

Another approach might be via supervised methods, using observed user behaviour as direct or indirect supervision. Adaptive topic modelling has been investigated as part of TDT, with models of document topic relevance adjusted according to user feedback as to whether a document is on- or off-topic (see e.g. Allan et al. 2000; Lo and Gauvain 2001); but the effect on segmentation has received less attention. However, Banerjee and Rudnicky (2007) show how supervision can be exploited when segmenting meetings, by providing users with a note-taking tool annotated with agenda items: by observing the times when notes are made against particular agenda items, they can improve the accuracy of their agenda item segmenter. By allowing users to define their own topics, a user-specific model could be learnt; one can also imagine this being extended to other domains by observing suitable behaviour such as user interaction with a browser.

## References


Allan J, Carbonell J, Doddington G, Yamron J and Yang Y 1998 Topic detection and tracking pilot study: Final report *Proceedings of the 1998 DARPA Broadcast News Transcription and Understanding Workshop*.

Allan J, Lavrenko V, Frey D and Khandelwal V 2000 UMass at TDT 2000 *Proceedings of the Topic Detection and Tracking workshop*, pp. 109–115.

AlSumait L, Barbará D and Domeniconi C 2008 On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking *Proceedings of the IEEE International Conference on Data Mining*, Pisa, Italy.

Arguello J and Rosé C 2006 Topic segmentation of dialogue *Proceedings of the HLT-NAACL Workshop on Analyzing Conversations in Text and Speech*, New York, NY.

Asher N and Lascarides A 2003 *Logics of Conversation*. Cambridge University Press.

Avrithis Y, Tsapatsoulis N and Kollias S 2000 Broadcast news parsing using visual cues: A robust face detection approach *IEEE International Conference on Multimedia and Expo*, New York, NY.

Banerjee S and Rudnicky A 2006 Smartnotes: Implicit labeling of meeting data through user note-taking and browsing *Proceedings of the Human Language Techonolgy Conference of the NAACL, Companion Volume*.

Banerjee S and Rudnicky A 2007 Segmenting meetings into agenda items by extracting implicit supervision from human note-taking *Prooceedings of the International Conference on Intelligent User Interfaces (IUI'07)*. ACM, Honolulu, Hawaii.

Banerjee S, Rosé C and Rudnicky A 2005 The necessity of a meeting recording and playback system, and the benefit of topic-level annotations to meeting browsing *Proceedings of the 10th International Conference on Human-Computer Interaction (CHI)*.

Barzilay R and Lee L 2004 Catching the drift: Probabilistic content models, with applications to generation and summarization *Proceedings of the 2004 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)*, pp. 113–120.

Beeferman D, Berger A and Lafferty JD 1999 Statistical models for text segmentation. *Machine Learning* **34**(1-3), 177–210.

Blei D and Moreno P 2001 Topic segmentation with an aspect hidden Markov model *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 343–348.

Blei D, Ng A and Jordan M 2003 Latent Dirichlet allocation. *Journal of Machine Learning Research* **3**, 993–1022.

Cassell J, Nakano Y, Bickmore TW, Sidner CL and Rich C 2001 Non-verbal cues for discourse structure *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pp. 114–123. Association for Computational Linguistics, Toulouse, France.

Chaisorn L, Chua TS, Koh CK, Zhao Y, Xu H, Feng H and Tian Q 2003 A two-level multi-modal approach for story segmentation of large news video corpus *Proceedings of TRECVID*.

Choi FY 2000 Advances in domain independent linear text segmentation *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Choi FYY, Wiemer-Hastings P and Moore J 2001 Latent semantic analysis for text segmentation *Proceedings of EMNLP*.

Church K 1993 Char align: A program for aligning parallel texts at the character level *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pp. 1–8. Association for Computational Linguistics, Columbus, Ohio, USA.

Doddington G 1998 The topic detection and tracking phase 2 (TDT2) evaluation plan *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, pp. 223–229. Morgan Kaufmann, Lansdowne, VA.

Dowman M, Savova V, Griffiths TL, Körding KP, Tenenbaum JB and Purver M 2008 A probabilistic model of meetings that combines words and discourse features. *IEEE Transactions on Audio, Speech, and Language Processing* **16**(7), 1238–1248.

Eisenstein J and Barzilay R 2008 Bayesian unsupervised topic segmentation *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pp. 334–343. Association for Computational Linguistics, Honolulu, Hawaii.

Eisenstein J, Barzilay R and Davis R 2008 Gestural cohesion for topic segmentation *Proceedings of ACL-08: HLT*, pp. 852–860. Association for Computational Linguistics, Columbus, Ohio.

Franz M, Ramabhadran B, Ward T and Picheny M 2003 Automated transcription and topic segmentation of large spoken archives *Proceedings of Eurospeech*, pp. 953–956.

Fügen C, Wölfel M, McDonough J, Ikbal S, Kraft F, Laskowski K, Ostendorf M, Stüker S and Kumatani K 2006 Advances in lecture recognition: The ISL RT-06S evaluation system *Proceedings of Interspeech-ICSLP*.

Galley M, McKeown K, Fosler-Lussier E and Jing H 2003 Discourse segmentation of multi-party conversation *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL)*.

Georgescul M, Clark A and Armstrong S 2006a An analysis of quantitative aspects in the evaluation of thematic segmentation algorithms *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, pp. 144–151. Association for Computational Linguistics, Sydney, Australia.

Georgescul M, Clark A and Armstrong S 2006b Word distributions for thematic segmentation in a support vector machine approach *Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL-X)*, pp. 101–108, New York City, New York.

Georgescul M, Clark A and Armstrong S 2007 Exploiting structural meeting-specific features for topic segmentation *Actes de la 14ème Conférence sur le Traitement Automatique des Langues Naturelles* Association pour le Traitement Automatique des Langues, Toulouse, France.

Georgescul M, Clark A and Armstrong S 2008 A comparative study of mixture models for automatic topic segmentation of multiparty dialogues *The 3rd International Joint Conference on Natural Language Processing (IJCNLP)*, Hyderabad, India.

Ginzburg J 2011 *The Interactive Stance: Meaning for Conversation*. Oxford University Press.

Glass J, Hazen T, Cypers S, Malioutov I, Huynh D and Barzilay R 2007 Recent progress in the MIT spoken lecture processing project *Proceedings of Interspeech*, Antwerp, Belgium.

Griffiths T and Steyvers M 2004 Finding scientific topics. *Proceedings of the National Academy of Science* **101**, 5228–5235.

Griffiths T, Steyvers M, Blei D and Tenenbaum J 2005 Integrating topics and syntax *Proceedings of NIPS '04, Advances in Neural Information Processing Systems 17*.

Grosz BJ and Sidner CL 1986 Attention, intentions, and the structure of discourse. *Computational Linguistics* **12**(3), 175–204.

Gruenstein A, Niekrasz J and Purver M 2008 Meeting structure annotation: Annotations collected with a general purpose toolkit In *Recent Trends in Discourse and Dialogue* (ed. Dybkjaer L and Minker W) vol. 39 of *Text, Speech and Language Technology* Springer Dordrecht pp. 247–274.

Hearst M 1997 TextTiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics* **23**(1), 33–64.

Hearst M and Plaunt C 1993 Subtopic structuring for full-length document access *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 59–68, Pittsburgh, PA.

Hearst MA 1994 Multi-paragraph segmentation of expository text *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL)*, Los Cruces, New Mexico.

Hirschberg J and Litman D 1993 Empirical studies on the disambiguation of cue phrases. *Computational Linguistics* **19**(3), 501–530.

Hirschberg J and Nakatani C 1998 Acoustic indicators of topic segmentation *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP)*.

Hirschberg J and Nakatani CH 1996 A prosodic analysis of discourse segments in direction-giving monologues *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 286–293. Association for Computational Linguistics, Santa Cruz, California, USA.

Hofmann T 1999 Probablistic latent semantic indexing *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 50–57.

Hsueh PY and Moore J 2006 Automatic topic segmentation and lablelling in multiparty dialogue *Proceedings of the 1st IEEE/ACM Workshop on Spoken Language Technology (SLT)*, Palm Beach, Aruba.

Hsueh PY, Moore J and Renals S 2006 Automatic segmentation of multiparty dialogue *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Janin A, Baron D, Edwards J, Ellis D, Gelbart D, Morgan N, Peskin B, Pfau T, Shriberg E, Stolcke A and Wooters C 2003 The ICSI meeting corpus *Proceedings of the 2003 International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

Jurafsky D and Martin J 2009 *Speech and Language Processing* 2nd edn. Pearson Prentice Hall.

Landauer TK, Foltz PW and Laham D 1998 Introduction to latent semantic analysis. *Discourse Processes* **25**, 259–284.

Larsson S 2002 *Issue-based Dialogue Management* PhD thesis Göteborg University. Also published as Gothenburg Monographs in Linguistics 21.

Lisowska A 2003 Multimodal interface design for the multimodal meeting domain: Preliminary indications from a query analysis study. Technical Report IM2.MDM-11, ISSCO, University of Geneva.

Lo Y and Gauvain J 2001 The LIMSI topic tracking system for TDT 2001 *Proceedings of the Topic Detection and Tracking workshop*.

Malioutov I and Barzilay R 2006 Minimum cut model for spoken lecture segmentation *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 25–32. Association for Computational Linguistics, Sydney, Australia.

Mann W and Thompson S 1988 Rhetorical structure theory: Toward a functional theory of text organization. *Text* **8**(3), 243–281.

Manning C and Schütze H 1999 *Foundations of Statistical Natural Language Processing*. MIT Press.

Marcu D 2000 *The Theory and Practice of Discourse Parsing and Summarization*. MIT Press, Cambridge, MA.

Maybury MT 1998 Discourse cues for broadcast news segmentation *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 2*, pp. 819–822. Association for Computational Linguistics, Montreal, Quebec, Canada.

McCowan I, Carletta J, Kraaij W, Ashby S, Bourban S, Flynn M, Guillemot M, Hain T, Kadlec J, Karaiskos V, Kronenthal M, Lathoud G, Lincoln M, Lisowska A, Post W, Reidsma D and Wellner P 2005 The AMI Meeting Corpus *Proceedings of Measuring Behavior 2005, the 5th International Conference on Methods and Techniques in Behavioral Research*, Wageningen, Netherlands.

Mohri M, Moreno P and Weinstein E 2009 A new quality measure for topic segmentation of text and speech *Conference of the International Speech Communication Association (Interspeech)*, Brighton, UK.

Morris J and Hirst G 1991 Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Computational Linguistics* **17**(1), 21–48.

Mulbregt PV, Carp I, Gillick L, Lowe S and Yamron J 1999 Segmentation of automatically transcribed broadcast news text *Proceedings of the DARPA Broadcast News Workshop*, pp. 77–80. Morgan Kaufmann.

Niekrasz J and Moore J 2009 Participant subjectivity and involvement as a basis for discourse segmentation *Proceedings of the SIGDIAL 2009 Conference*, pp. 54–61. Association for Computational Linguistics, London, UK.

Oard DW and Leuski A 2003 Searching recorded speech based on the temporal extent of topic labels *Proceedings of AAAI Spring Symposium on Intelligent Multimedia Knowledge Management*, Palo Alto, CA.

Olney A and Cai Z 2005 An orthonormal basis for topic segmentation in tutorial dialogue *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing HLT/EMNLP*, pp. 971–978. Association for Computational Linguistics, Vancouver, BC.

Passonneau RJ and Litman DJ 1997 Discourse segmentation by human and automated means. *Computational Linguistics* **23**(1), 103–139.

Passonneau RJ and Litman DJL 1996 Empirical analysis of three dimensions of spoken discourse: Segmentation, coherence and linguistic devices In *Interdisciplinary Perspectives on Discourse* (ed. Hovy E and Scott D) Springer-Verlag.

Pevzner L and Hearst M 2002 A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics* **28**(1), 19–36.

Polanyi L 1988 A formal model of discourse structure. *Journal of Pragmatics* **12**, 601–638.

Popescu-Belis A, Clark A, Georgescul M, Lalanne D and Zufferey S 2005 Shallow dialogue processing using machine learning algorithms (or not) In *Machine Learning for Multimodal Interaction: First International Workshop, MLMI 2004, Revised Selected Papers* (ed. Bengio S and Bourlard H) vol. 3361 of *Lecture Notes in Computer Science* Springer pp. 277–290.

Power R, Scott D and Bouayad-Agha N 2003 Document structure. *Computational Linguistics* **29**, 211–260.

Purver M, Körding K, Griffiths T and Tenenbaum J 2006 Unsupervised topic modelling for multi-party spoken discourse *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL)*, pp. 17–24. Association for Computational Linguistics, Sydney, Australia.

Reynar J 1994 An automatic method of finding topic boundaries *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 331–333. Association for Computational Linguistics, Las Cruces, NM.

Reynar J 1999 Statistical models for topic segmentation *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 357–364.

Sun Q, Li R, Luo D and Wu X 2008 Text segmentation with LDA-based Fisher kernel *Proceedings of ACL-08: HLT, Short Papers*, pp. 269–272. Association for Computational Linguistics, Columbus, Ohio.

Trancoso I, Nunes R and Neves L 2006 Recognition of classroom lectures in European Portuguese *Proceedings of Interspeech-ICSLP*.

Tür G, Hakkani-Tür D, Stolcke A and Shriberg E 2001 Integrating prosodic and lexical cues for automatic topic segmentation. *Computational Linguistics* **27**(1), 31–57.

Utiyama M and Isahara H 2001 A statistical model for domain-independent text segmentation *Proceedings of 39th Annual Meeting of the Association for Computational Linguistics*, pp. 499–506. Association for Computational Linguistics, Toulouse, France.

Vapnik VN 1995 *The Nature of Statistical Learning Theory*. Springer.

Yaari Y 1997 Segmentation of expository texts by hierarchical agglomerative clustering *Proceedings of RANLP*.

Yamron J, Carp I, Gillick L, Lowe S and van Mulbregt P 1998 A hidden Markov model approach to text segmentation and event tracking *Proceedings of the IEEE Conference on Acoustics, Speech, and Signal Processing*.

Youmans G 1991 A new tool for discourse analysis: The vocabulary-management profile. *Language* **67**(4), 763–789.