# Making Sense of Sound:
# Unsupervised Topic Segmentation over Acoustic Input

**Igor Malioutov, Alex Park, Regina Barzilay,** and **James Glass**
Massachusetts Institute of Technology
{igorm,malex,regina,glass}@csail.mit.edu

## Abstract

We address the task of unsupervised topic segmentation of speech data operating over raw acoustic information. In contrast to existing algorithms for topic segmentation of speech, our approach does not require input transcripts. Our method predicts topic changes by analyzing the distribution of reoccurring acoustic patterns in the speech signal corresponding to a single speaker. The algorithm robustly handles noise inherent in acoustic matching by intelligently aggregating information about the similarity profile from multiple local comparisons. Our experiments show that audio-based segmentation compares favorably with transcript-based segmentation computed over noisy transcripts. These results demonstrate the desirability of our method for applications where a speech recognizer is not available, or its output has a high word error rate.

## 1 Introduction

An important practical application of topic segmentation is the analysis of spoken data. Paragraph breaks, section markers and other structural cues common in written documents are entirely missing in spoken data. Insertion of these structural markers can benefit multiple speech processing applications, including audio browsing, retrieval, and summarization.

Not surprisingly, a variety of methods for topic segmentation have been developed in the past (Beeferman et al., 1999; Galley et al., 2003; Dielmann and Renals, 2005). These methods typically assume that a segmentation algorithm has access not only to acoustic input, but also to its transcript. This assumption is natural for applications where the transcript has to be computed as part of the system output, or it is readily available from other system components. However, for some domains and languages, the transcripts may not be available, or the recognition performance may not be adequate to achieve reliable segmentation. In order to process such data, we need a method for topic segmentation that does not require transcribed input.

In this paper, we explore a method for topic segmentation that operates directly on a raw acoustic speech signal, without using any input transcripts. This method predicts topic changes by analyzing the distribution of reoccurring acoustic patterns in the speech signal corresponding to a single speaker. In the same way that unsupervised segmentation algorithms predict boundaries based on changes in lexical distribution, our algorithm is driven by changes in the distribution of acoustic patterns. The central hypothesis here is that similar sounding acoustic sequences produced by the same speaker correspond to similar lexicographic sequences. Thus, by analyzing the distribution of acoustic patterns we could approximate a traditional content analysis based on the lexical distribution of words in a transcript.

Analyzing high-level content structure based on low-level acoustic features poses interesting computational and linguistic challenges. For instance, we need to handle the noise inherent in matching based on acoustic similarity, because of possible varia-

*Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 504–511,
Prague, Czech Republic, June 2007. ©2007 Association for Computational Linguistics

tions in speaking rate or pronunciation. Moreover, in the absence of higher-level knowledge, information about word boundaries is not always discernible from the raw acoustic input. This causes problems because we have no obvious unit of comparison. Finally, noise inherent in the acoustic matching procedure complicates the detection of distributional changes in the comparison matrix.

The algorithm presented in this paper demonstrates the feasibility of topic segmentation over raw acoustic input corresponding to a single speaker. We first apply a variant of the dynamic time warping algorithm to find similar fragments in the speech input through alignment. Next, we construct a comparison matrix that aggregates the output of the alignment stage. Since aligned utterances are separated by gaps and differ in duration, this representation gives rise to sparse and irregular input. To obtain robust similarity change detection, we invoke a series of transformations to smooth and refine the comparison matrix. Finally, we apply the minimum-cut segmentation algorithm to the transformed comparison matrix to detect topic boundaries.

We compare the performance of our method against traditional transcript-based segmentation algorithms. As expected, the performance of the latter depends on the accuracy of the input transcript. When a manual transcription is available, the gap between audio-based segmentation and transcript-based segmentation is substantial. However, in a more realistic scenario when the transcripts are fraught with recognition errors, the two approaches exhibit similar performance. These results demonstrate that audio-based algorithms are an effective and efficient solution for applications where transcripts are unavailable or highly errorful.

## 2 Related Work

**Speech-based Topic Segmentation** A variety of supervised and unsupervised methods have been employed to segment speech input. Some of these algorithms have been originally developed for processing written text (Beeferman et al., 1999). Others are specifically adapted for processing speech input by adding relevant acoustic features such as pause length and speaker change (Galley et al., 2003; Dielmann and Renals, 2005). In parallel, researchers extensively study the relationship between discourse structure and intonational variation (Hirschberg and Nakatani, 1996; Shriberg et al., 2000). However, all of the existing segmentation methods require as input a speech transcript of reasonable quality. In contrast, the method presented in this paper does not assume the availability of transcripts, which prevents us from using segmentation algorithms developed for written text.

At the same time, our work is closely related to unsupervised approaches for text segmentation. The central assumption here is that sharp changes in lexical distribution signal the presence of topic boundaries (Hearst, 1994; Choi et al., 2001). These approaches determine segment boundaries by identifying homogeneous regions within a similarity matrix that encodes pairwise similarity between textual units, such as sentences. Our segmentation algorithm operates over a distortion matrix, but the unit of comparison is the speech signal over a time interval. This change in representation gives rise to multiple challenges related to the inherent noise of acoustic matching, and requires the development of new methods for signal discretization, interval comparison and matrix analysis.

**Pattern Induction in Acoustic Data** Our work is related to research on unsupervised lexical acquisition from continuous speech. These methods aim to infer vocabulary from unsegmented audio streams by analyzing regularities in pattern distribution (de Marcken, 1996; Brent, 1999; Venkataraman, 2001). Traditionally, the speech signal is first converted into a string-like representation such as phonemes and syllables using a phonetic recognizer.

Park and Glass (2006) have recently shown the feasibility of an audio-based approach for word discovery. They induce the vocabulary from the audio stream directly, avoiding the need for phonetic transcription. Their method can accurately discover words which appear with high frequency in the audio stream. While the results obtained by Park and Glass inspire our approach, we cannot directly use their output as proxies for words in topic segmentation. Many of the content words occurring only a few times in the text are pruned away by this method. Our results show that this data that is too sparse and noisy for robustly discerning changes in lexical distribution.

## 3 Algorithm

The audio-based segmentation algorithm identifies topic boundaries by analyzing changes in the distribution of acoustic patterns. The analysis is performed in three steps. First, we identify recurring patterns in the audio stream and compute distortion between them (Section 3.1). These acoustic patterns correspond to high-frequency words and phrases, but they only cover a fraction of the words that appear in the input. As a result, the distributional profile obtained during this process is too sparse to deliver robust topic analysis. Second, we generate an acoustic comparison matrix that aggregates information from multiple pattern matches (Section 3.2). Additional matrix transformations during this step reduce the noise and irregularities inherent in acoustic matching. Third, we partition the matrix to identify segments with a homogeneous distribution of acoustic patterns (Section 3.3).

### 3.1 Comparing Acoustic Patterns

Given a raw acoustic waveform, we extract a set of acoustic patterns that occur frequently in the speech document. Continuous speech includes many word sequences that lack clear low-level acoustic cues to denote word boundaries. Therefore, we cannot perform this task through simple counting of speech segments separated by silence. Instead, we use a local alignment algorithm to search for similar speech segments and quantify the amount of distortion between them. In what follows, we first present a vector representation used in this computation, and then specify the alignment algorithm that finds similar segments.

**MFCC Representation** We start by transforming the acoustic signal into a vector representation that facilitates the comparison of acoustic sequences. First, we perform silence detection on the original waveform by registering a pause if the energy falls below a certain threshold for a duration of 2s. This enables us to break up the acoustic stream into continuous spoken utterances.

This step is necessary as it eliminates spurious alignments between silent regions of the acoustic waveform. Note that silence detection is not equivalent to word boundary detection, as segmentation by silence detection alone only accounts for 20% of word boundaries in our corpus.

Next, we convert each utterance into a time series of vectors consisting of Mel-scale cepstral coefficients (MFCCs). This compact low-dimensional representation is commonly used in speech processing applications because it approximates human auditory models.

The process of extracting MFCCs from the speech signal can be summarized as follows. First, the 16 kHz digitized audio waveform is normalized by removing the mean and scaling the peak amplitude. Next, the short-time Fourier transform is taken at a frame interval of 10 ms using a 25.6 ms Hamming window. The spectral energy from the Fourier transform is then weighted by Mel-frequency filters (Huang et al., 2001). Finally, the discrete cosine transform of the log of these Mel-frequency spectral coefficients is computed, yielding a series of 14-dimensional MFCC vectors. We take the additional step of whitening the feature vectors, which normalizes the variance and decorrelates the dimensions of the feature vectors (Bishop, 1995). This whitened spectral representation enables us to use the standard unweighted Euclidean distance metric. After this transformation, the distances in each dimension will be uncorrelated and have equal variance.

**Alignment** Now, our goal is to identify acoustic patterns that occur multiple times in the audio waveform. The patterns may not be repeated exactly, but will most likely reoccur in varied forms. We capture this information by extracting pairs of patterns with an associated distortion score. The computation is performed using a sequence alignment algorithm.

Table 1 shows examples of alignments automatically computed by our algorithm. The corresponding phonetic transcriptions[1] demonstrate that the matching procedure can robustly handle variations in pronunciations. For example, two instances of the word *"direction"* are matched to one another despite different pronunciations, ("d ay" vs. "d ax" in the first syllable). At the same time, some aligned pairs form erroneous matches, such as *"my prediction"* matching *"y direction"* due to their high acoustic

[1] Phonetic transcriptions are not used by our algorithm and are provided for illustrative purposes only.

| Aligned Word(s) | Phonetic Transcription |
|---|---|
| the x direction | dh iy eh kcl k s dcl d ax r eh kcl sh ax n<br>ð i$^{y}$ ɛk̚k s d̚d ər ɛk̚ʃən |
| the y direction | dh ax w ay dcl d ay r eh kcl sh epi en<br>ð əw a$^{y}$ d̚a$^{y}$ r ɛk̚k ʃən |
| of my prediction | ax v m ay kcl k r iy l iy kcl k sh ax n<br>əv m a$^{y}$ k̚k r i$^{y}$ l i$^{y}$ k̚k ʃən |
| acceleration | eh kcl k s eh l ax r ey sh epi en<br>ɛk̚k s ɛl ər ɛ$^{y}$ ʃ- n̩ |
| acceleration | ax kcl k s ah n ax r eh n epi sh epi en<br>ək̚k s ʌn ər ɛn - ʃ- n̩ |
| the derivation | dcl d ih dx ih z dcl dh ey sh epi en<br>d̚d ɪɾɪz d̚ð ɛ$^{y}$ ʃ- n̩ |
| a demonstration | uh dcl d eh m ax n epi s tcl t r ey sh en<br>ʊd̚d ɛm ən - s t̚t r ɛ$^{y}$ ʃn̩ |

Table 1: Aligned Word Paths. Each group of rows represents audio segments that were aligned to one another, along with their corresponding phonetic transcriptions using TIMIT conventions (Garofolo et al., 1993) and their IPA equivalents.

similarity.

The alignment algorithm operates on the audio waveform represented by a list of silence-free utterances $(u_1, u_2, \ldots, u_n)$. Each utterance $u'$ is a time series of MFCC vectors $(\vec{x'_1}, \vec{x'_2}, \ldots, \vec{x'_m})$. Given two input utterances $u'$ and $u''$, the algorithm outputs a set of alignments between the corresponding MFCC vectors. The alignment distortion score is computed by summing the Euclidean distances of matching vectors.

To compute the optimal alignment we use a variant of the dynamic time warping algorithm (Huang et al., 2001). For every possible starting alignment point, we optimize the following dynamic programming objective:

$$D(i_k, j_k) = d(i_k, j_k) + \min \begin{cases} D(i_k - 1, j_k) \\ D(i_k, j_k - 1) \\ D(i_k - 1, j_k - 1) \end{cases}$$

In the equation above, $i_k$ and $j_k$ are alignment endpoints in the $k$-th subproblem of dynamic programming.

This objective corresponds to a descent through a dynamic programming trellis by choosing right, down, or diagonal steps at each stage.

During the search process, we consider not only the alignment distortion score, but also the shape of the alignment path. To limit the amount of temporal warping, we enforce the following constraint:

$$\left|\big(i_k - i_1\big) - \big(j_k - j_1\big)\right| \leq R, \forall k, \qquad (1)$$

$$i_k \leq N_x \quad \text{and} \quad j_k \leq N_y,$$

where $N_x$ and $N_y$ are the number of MFCC samples in each utterance. The value $2R + 1$ is the width of the diagonal band that controls the extent of temporal warping. The parameter $R$ is tuned on a development set.

This alignment procedure may produce paths with high distortion subpaths. Therefore, we trim each path to retain the subpath with lowest average distortion and length at least $L$. More formally, given an alignment of length $N$, we seek to find $m$ and $n$ such that:

$$\underset{1 \leq m \leq n \leq N}{\arg\min} \frac{1}{n - m + 1} \sum_{k=m}^{n} d(i_k, j_k) \qquad n - m \geq L$$

We accomplish this by computing the length constrained minimum average distortion subsequence of the path sequence using an $O(N \log(L))$ algorithm proposed by Lin et al (2002). The length parameter, $L$, allows us to avoid overtrimming and control the length of alignments that are found. After trimming, the distortion of each alignment path is normalized by the path length.

Alignments with a distortion exceeding a prespecified threshold are pruned away to ensure that the aligned phrasal units are close acoustic matches. This parameter is tuned on a development set.

In the next section, we describe how to aggregate information from multiple noisy matches into a representation that facilitates boundary detection.

### 3.2 Construction of Acoustic Comparison Matrix

The goal of this step is to construct an acoustic comparison matrix that will guide topic segmentation. This matrix encodes variations in the distribution of acoustic patterns for a given speech document. We construct this matrix by first discretizing the acoustic signal into constant-length blocks and then computing the distortion between pairs of blocks.

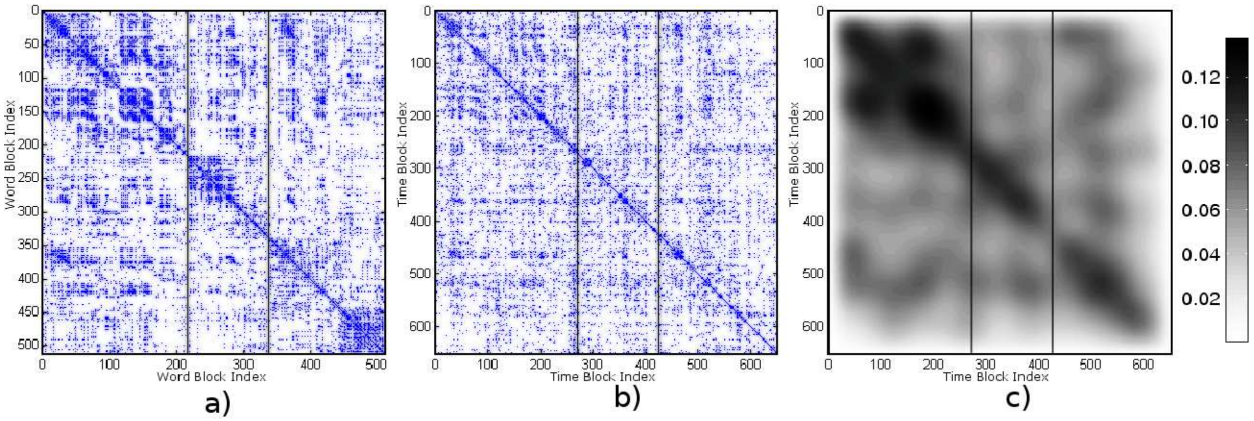

Figure 1: a) Similarity matrix for a Physics lecture constructed using a manual transcript. b) Similarity matrix for the same lecture constructed from acoustic data. The intensity of a pixel indicates the degree of block similarity. c) Acoustic comparison matrix after 2000 iterations of anisotropic diffusion. Vertical lines correspond to the reference segmentation.

Unfortunately, the paths and distortions generated during the alignment step (Section 3.1) cannot be mapped directly to an acoustic comparison matrix. Since we compare only commonly repeated acoustic patterns, some portions of the signal correspond to gaps between alignment paths. In fact, in our corpus only 67% of the data is covered by alignment paths found during the alignment stage. Moreover, many of these paths are not disjoint. For instance, our experiments show that 74% of them overlap with at least one additional alignment path. Finally, these alignments vary significantly in duration, ranging from 0.350 ms to 2.7 ms in our corpus.

**Discretization and Distortion Computation** To compensate for the irregular distribution of alignment paths, we quantize the data by splitting the input signal into uniform contiguous time blocks. A time block does not necessarily correspond to any one discovered alignment path. It may contain several complete paths and also portions of other paths. We compute the aggregate distortion score $D(x, y)$ of two blocks $x$ and $y$ by summing the distortions of all alignment paths that fall within $x$ and $y$.

**Matrix Smoothing** Equipped with a block distortion measure, we can now construct an acoustic comparison matrix. In principle, this matrix can be processed employing standard methods developed for text segmentation. However, as Figure 1 illustrates, the structure of the acoustic matrix is quite different from the one obtained from text. In a transcript similarity matrix shown in Figure 1 a), reference boundaries delimit homogeneous regions with high internal similarity. On the other hand, looking at the acoustic similarity matrix[2] shown in Figure 1 b), it is difficult to observe any block structure corresponding to the reference segmentation.

This deficiency can be attributed to the sparsity of acoustic alignments. Consider, for example, the case when a segment is interspersed with blocks that contain very few or no complete paths. Even though the rest of the blocks in the segment could be closely related, these path-free blocks dilute segment homogeneity. This is problematic because it is not always possible to tell whether a sudden shift in scores signifies a transition or if it is just an artifact of irregularities in acoustic matching. Without additional matrix processing, these irregularities will lead the system astray.

We further refine the acoustic comparison matrix using *anisotropic diffusion*. This technique has been developed for enhancing edge detection accuracy in image processing (Perona and Malik, 1990), and has been shown to be an effective smoothing method in text segmentation (Ji and Zha, 2003). When applied to a comparison matrix, anisotropic diffusion reduces score variability within homogeneous re-

[2] We converted the original comparison distortion matrix to the similarity matrix by subtracting the component distortions from the maximum alignment distortion score.

gions of the matrix and makes edges between these regions more pronounced. Consequently, this transformation facilitates boundary detection, potentially increasing segmentation accuracy. In Figure 1 c), we can observe that the boundary structure in the diffused comparison matrix becomes more salient and corresponds more closely to the reference segmentation.

### 3.3 Matrix Partitioning

Given a target number of segments $k$, the goal of the partitioning step is to divide a matrix into $k$ square submatrices along the diagonal. This process is guided by an optimization function that maximizes the homogeneity within a segment or minimizes the homogeneity across segments. This optimization problem can be solved using one of many unsupervised segmentation approaches (Choi et al., 2001; Ji and Zha, 2003; Malioutov and Barzilay, 2006).

In our implementation, we employ the minimum-cut segmentation algorithm (Shi and Malik, 2000; Malioutov and Barzilay, 2006). In this graph-theoretic framework, segmentation is cast as a problem of partitioning a weighted undirected graph that minimizes the *normalized-cut criterion*. The minimum-cut method achieves robust analysis by jointly considering all possible partitionings of a document, moving beyond localized decisions. This allows us to aggregate comparisons from multiple locations, thereby compensating for the noise of individual matches.

## 4 Evaluation Set-Up

**Data** We use a publicly available[3] corpus of introductory Physics lectures described in our previous work (Malioutov and Barzilay, 2006). This material is a particularly appealing application area for an audio-based segmentation algorithm — many academic subjects lack transcribed data for training, while a high ratio of in-domain technical terms limits the use of out-of-domain transcripts. This corpus is also challenging from the segmentation perspective because the lectures are long and transitions between topics are subtle.

The corpus consists of 33 lectures, with an average length of 8500 words and an average duration of 50 minutes. On average, a lecture was annotated with six segments, and a typical segment corresponds to two pages of a transcript. Three lectures from this set were used for development, and 30 lectures were used for testing. The lectures were delivered by the same speaker.

To evaluate the performance of traditional transcript-based segmentation algorithms on this corpus, we also use several types of transcripts at different levels of recognition accuracy. In addition to manual transcripts, our corpus contains two types of automatic transcripts, one obtained using speaker-dependent (SD) models and the other obtained using speaker-independent (SI) models. The speaker-independent model was trained on 85 hours of out-of-domain general lecture material and contained no speech from the speaker in the test set. The speaker-dependent model was trained by using 38 hours of audio data from other lectures given by the speaker. Both recognizers incorporated word statistics from the accompanying class textbook into the language model. The word error rates for the speaker-independent and speaker-dependent models are 44.9% and 19.4%, respectively.

**Evaluation Metrics** We use the $P_k$ and WindowDiff measures to evaluate our system (Beeferman et al., 1999; Pevzner and Hearst, 2002). The $P_k$ measure estimates the probability that a randomly chosen pair of words within a window of length $k$ words is inconsistently classified. The WindowDiff metric is a variant of the $P_k$ measure, which penalizes false positives and near misses equally. For both of these metrics, lower scores indicate better segmentation accuracy.

**Baseline** We use the state-of-the-art mincut segmentation system by Malioutov and Barzilay (2006) as our point of comparison. This model is an appropriate baseline, because it has been shown to compare favorably with other top-performing segmentation systems (Choi et al., 2001; Utiyama and Isahara, 2001). We use the publicly available implementation of the system.

As additional points of comparison, we test the uniform and random baselines. These correspond to segmentations obtained by uniformly placing

[3] See `http://www.csail.mit.edu/~igorm/acl06.html`

| | $P_k$ | WindowDiff |
|---|---|---|
| MAN | 0.298 | 0.311 |
| SD | 0.340 | 0.351 |
| **AUDIO** | **0.358** | **0.370** |
| SI | 0.378 | 0.390 |
| RAND | 0.472 | 0.497 |
| UNI | 0.476 | 0.484 |

Table 2: Segmentation accuracy for audio-based segmentor (AUDIO), random (RAND), uniform (UNI) and three transcript-based segmentation algorithms that use manual (MAN), speaker-dependent (SD) and speaker-independent (SI) transcripts. For all of the algorithms, the target number of segments is set to the reference number of segments.

boundaries along the span of the lecture and selecting random boundaries, respectively.

To control for segmentation granularity, we specify the number of segments in the reference segmentation for both our system and the baselines.

**Parameter Tuning** We tuned the number of quantized blocks, the edge cutoff parameter of the minimum cut algorithm, and the anisotropic diffusion parameters on a heldout set of three development lectures. We used the same development set for the baseline segmentation systems.

## 5 Results

The goal of our evaluation experiments is two-fold. First, we are interested in understanding the conditions in which an audio-based segmentation is advantageous over a transcript-based one. Second, we aim to analyze the impact of various design decisions on the performance of our algorithm.

**Comparison with Transcript-Based Segmentation** Table 2 shows the segmentation accuracy of the audio-based segmentation algorithm and three transcript-based segmentors on the set of 30 Physics lectures. Our algorithm yields an average $P_k$ measure of 0.358 and an average WindowDiff measure of 0.370. This result is markedly better than the scores attained by uniform and random segmentations. As expected, the best segmentation results are obtained using manual transcripts. However, the gap between audio-based segmentation and transcript-based segmentation narrows when the recognition accuracy decreases. In fact, performance of the audio-based segmentation beats the transcript-based segmentation baseline obtained using speaker-independent (SI) models (0.358 for AUDIO versus $P_k$ measurements of 0.378 for SI).

**Analysis of Audio-based Segmentation** A central challenge in audio-based segmentation is how to overcome the noise inherent in acoustic matching. We addressed this issue by using anisotropic diffusion to refine the comparison matrix. We can quantify the effects of this smoothing technique by generating segmentations directly from the similarity matrix. We obtain similarities from the distortions in the comparison matrix by subtracting the distortion scores from the maximum distortion:

$$S(x, y) = \max_{s_i, s_j} [D(s_i, s_j)] - D(x, y)$$

Using this matrix with the min-cut algorithm, segmentation accuracy drops to a $P_k$ measure of 0.418 (0.450 WindowDiff). This difference in performance shows that anisotropic diffusion compensates for noise introduced during acoustic matching.

An alternative solution to the problem of irregularities in audio-based matching is to compute clusters of acoustically similar utterances. Each of the derived clusters can be thought of as a unique word type.[4] We compute these clusters, employing a method for unsupervised vocabulary induction developed by Park and Glass (2006). Using the output of their algorithm, the continuous audio stream is transformed into a sequence of word-like units, which in turn can be segmented using any standard transcript-based segmentation algorithm, such as the minimum-cut segmentor. On our corpus, this method achieves disappointing results — a $P_k$ measure of 0.423 (0.424 WindowDiff). The result can be attributed to the sparsity of clusters[5] generated by this method, which focuses primarily on discovering the frequently occurring content words.

## 6 Conclusion and Future Work

We presented an unsupervised algorithm for audio-based topic segmentation. In contrast to existing

[4] In practice, a cluster can correspond to a phrase, word, or word fragment (See Table 1 for examples).

[5] We tuned the number of clusters on the development set.

algorithms for speech segmentation, our approach does not require an input transcript. Thus, it can be used in domains where a speech recognizer is not available or its output is too noisy. Our approach approximates the distribution of cohesion ties by considering the distribution of acoustic patterns. Our experimental results demonstrate the utility of this approach: audio-based segmentation compares favorably with transcript-based segmentation computed over noisy transcripts.

The segmentation algorithm presented in this paper focuses on one source of linguistic information for discourse analysis — lexical cohesion. Multiple studies of discourse structure, however, have shown that prosodic cues are highly predictive of changes in topic structure (Hirschberg and Nakatani, 1996; Shriberg et al., 2000). In a supervised framework, we can further enhance audio-based segmentation by combining features derived from pattern analysis with prosodic information. We can also explore an unsupervised fusion of these two sources of information; for instance, we can induce informative prosodic cues by using distributional evidence.

Another interesting direction for future research lies in combining the results of noisy recognition with information obtained from distribution of acoustic patterns. We hypothesize that these two sources provide complementary information about the audio stream, and therefore can compensate for each other's mistakes. This combination can be particularly fruitful when processing speech documents with multiple speakers or background noise.

## 7 Acknowledgements

The authors acknowledge the support of the Microsoft Faculty Fellowship and the National Science Foundation (CAREER grant IIS-0448168, grant IIS-0415865, and the NSF Graduate Fellowship). Any opinions, findings, conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. We would like to thank T.J. Hazen for his assistance with the speech recognizer and to acknowledge Tara Sainath, Natasha Singh, Ben Snyder, Chao Wang, Luke Zettlemoyer and the three anonymous reviewers for their valuable comments and suggestions.

## References

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

C. Bishop, 1995. *Neural Networks for Pattern Recognition*, pg. 38. Oxford University Press, New York, 1995.

M. R. Brent. 1999. An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, 34(1-3):71–105.

F. Choi, P. Wiemer-Hastings, J. Moore. 2001. Latent semantic analysis for text segmentation. In *Proceedings of EMNLP*, 109–117.

C. G. de Marcken. 1996. *Unsupervised Language Acquisition*. Ph.D. thesis, Massachusetts Institute of Technology.

A. Dielmann, S. Renals. 2005. Multistream dynamic Bayesian network for meeting segmentation. In *Proceedings Multimodal Interaction and Related Machine Learning Algorithms Workshop (MLMI–04)*, 76–86.

M. Galley, K. McKeown, E. Fosler-Lussier, H. Jing. 2003. Discourse segmentation of multi-party conversation. In *Proceedings of the ACL*, 562–569.

J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallet, N. Dahlgren, V. Zue. 1993. TIMIT Acoustic-Phonetic Continuous Speech Corpus. Linguistic Data Consortium, 1993.

M. Hearst. 1994. Multi-paragraph segmentation of expository text. In *Proceedings of the ACL*, 9–16.

J. Hirschberg, C. H. Nakatani. 1996. A prosodic analysis of discourse segments in direction-giving monologues. In *Proceedings of the ACL*, 286–293.

X. Huang, A. Acero, H.-W. Hon. 2001. *Spoken Language Processing*. Prentice Hall.

X. Ji, H. Zha. 2003. Domain-independent text segmentation using anisotropic diffusion and dynamic programming. In *Proceedings of SIGIR*, 322–329.

Y.-L. Lin, T. Jiang, K.-M. Chao. 2002. Efficient algorithms for locating the length-constrained heaviest segments with applications to biomolecular sequence analysis. *J. Computer and System Sciences*, 65(3):570–586.

I. Malioutov, R. Barzilay. 2006. Minimum cut model for spoken lecture segmentation. In *Proceedings of the COLING/ACL*, 25–32.

A. Park, J. R. Glass. 2006. Unsupervised word acquisition from speech using pattern discovery. In *Proceedings of ICASSP*.

P. Perona, J. Malik. 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639.

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

J. Shi, J. Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905.

E. Shriberg, A. Stolcke, D. Hakkani-Tur, G. Tur. 2000. Prosody-based automatic segmentation of speech into sentences and topics. *Speech Communication*, 32(1-2):127–154.

M. Utiyama, H. Isahara. 2001. A statistical model for domain-independent text segmentation. In *Proceedings of the ACL*, 499–506.

A. Venkataraman. 2001. A statistical model for word discovery in transcribed speech. *Computational Linguistics*, 27(3):353–372.