Minimum Cut Model for Spoken Lecture Segmentation

Igor Malioutov and **Regina Barzilay**
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
{igorm,regina}@csail.mit.edu

Abstract

We consider the task of unsupervised lecture segmentation. We formalize segmentation as a graph-partitioning task that optimizes the normalized cut criterion. Our approach moves beyond localized comparisons and takes into account long-range cohesion dependencies. Our results demonstrate that global analysis improves the segmentation accuracy and is robust in the presence of speech recognition errors.

1 Introduction

The development of computational models of text structure is a central concern in natural language processing. Text segmentation is an important instance of such work. The task is to partition a text into a linear sequence of topically coherent segments and thereby induce a content structure of the text. The applications of the derived representation are broad, encompassing information retrieval, question-answering and summarization.

Not surprisingly, text segmentation has been extensively investigated over the last decade. Following the first unsupervised segmentation approach by Hearst (1994), most algorithms assume that variations in lexical distribution indicate topic changes. When documents exhibit sharp variations in lexical distribution, these algorithms are likely to detect segment boundaries accurately. For example, most algorithms achieve high performance on synthetic collections, generated by concatenation of random text blocks (Choi, 2000). The difficulty arises, however, when transitions between topics are smooth and distributional variations are subtle. This is evident in the performance of existing unsupervised algorithms on less structured datasets, such as spoken meeting transcripts (Galley et al., 2003). Therefore, a more refined analysis of lexical distribution is needed.

Our work addresses this challenge by casting text segmentation in a graph-theoretic framework. We abstract a text into a weighted undirected graph, where the nodes of the graph correspond to sentences and edge weights represent the pairwise sentence similarity. In this framework, text segmentation corresponds to a graph partitioning that optimizes the *normalized-cut criterion* (Shi and Malik, 2000). This criterion measures both the similarity within each partition and the dissimilarity across different partitions. Thus, our approach moves beyond localized comparisons and takes into account long-range changes in lexical distribution. Our key hypothesis is that global analysis yields more accurate segmentation results than local models.

We tested our algorithm on a corpus of spoken lectures. Segmentation in this domain is challenging in several respects. Being less structured than written text, lecture material exhibits digressions, disfluencies, and other artifacts of spontaneous communication. In addition, the output of speech recognizers is fraught with high word error rates due to specialized technical vocabulary and lack of in-domain spoken data for training. Finally, pedagogical considerations call for fluent transitions between different topics in a lecture, further complicating the segmentation task.

Our experimental results confirm our hypothesis: considering long-distance lexical dependencies yields substantial gains in segmentation performance. Our graph-theoretic approach compares favorably to state-of-the-art segmentation algorithms and attains results close to the range of human agreement scores. Another attractive prop-

Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL, pages 25–32,
Sydney, July 2006.

erty of the algorithm is its robustness to noise: the accuracy of our algorithm does not deteriorate significantly when applied to speech recognition output.

2 Previous Work

Most unsupervised algorithms assume that fragments of text with homogeneous lexical distribution correspond to topically coherent segments. Previous research has analyzed various facets of lexical distribution, including lexical weighting, similarity computation, and smoothing (Hearst, 1994; Utiyama and Isahara, 2001; Choi, 2000; Reynar, 1998; Kehagias et al., 2003; Ji and Zha, 2003).

The focus of our work, however, is on an orthogonal yet fundamental aspect of this analysis — the impact of long-range cohesion dependencies on segmentation performance. In contrast to previous approaches, the homogeneity of a segment is determined not only by the similarity of its words, but also by their relation to words in other segments of the text. We show that optimizing our global objective enables us to detect subtle topical changes.

Graph-Theoretic Approaches in Vision Segmentation Our work is inspired by minimum-cut-based segmentation algorithms developed for image analysis. Shi and Malik (2000) introduced the normalized-cut criterion and demonstrated its practical benefits for segmenting static images.

Our method, however, is not a simple application of the existing approach to a new task. First, in order to make it work in the new linguistic framework, we had to redefine the underlying representation and introduce a variety of smoothing and lexical weighting techniques. Second, the computational techniques for finding the optimal partitioning are also quite different. Since the minimization of the normalized cut is NP-complete in the general case, researchers in vision have to approximate this computation. Fortunately, we can find an exact solution due to the linearity constraint on text segmentation.

3 Minimum Cut Framework

Linguistic research has shown that word repetition in a particular section of a text is a device for creating thematic cohesion (Halliday and Hasan, 1976), and that changes in the lexical distributions usually signal topic transitions.

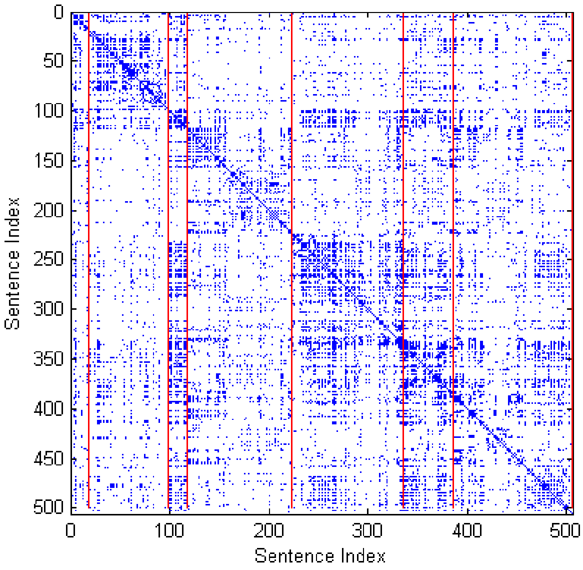

Figure 1: Sentence similarity plot for a Physics lecture, with vertical lines indicating true segment boundaries.

Figure 1 illustrates these properties in a lecture transcript from an undergraduate Physics class. We use the text Dotplotting representation by (Church, 1993) and plot the cosine similarity scores between every pair of sentences in the text. The intensity of a point (i, j) on the plot indicates the degree to which the i-th sentence in the text is similar to the j-th sentence. The true segment boundaries are denoted by vertical lines. This similarity plot reveals a block structure where true boundaries delimit blocks of text with high inter-sentential similarity. Sentences found in different blocks, on the other hand, tend to exhibit low similarity.

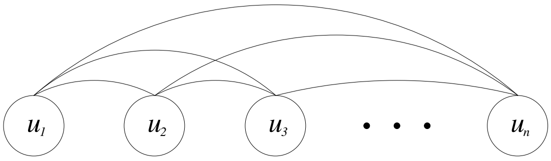

Figure 2: Graph-based Representation of Text

Formalizing the Objective Whereas previous unsupervised approaches to segmentation rested on intuitive notions of similarity density, we formalize the objective of text segmentation through cuts on graphs. We aim to jointly maximize the intra-segmental similarity and minimize the similarity between different segments. In other words, we want to find the segmentation with a maximally homogeneous set of segments that are also maximally different from each other.

Let $G = \{V, E\}$ be an undirected, weighted graph, where V is the set of nodes corresponding to sentences in the text and E is the set of weighted edges (See Figure 2). The edge weights, $w(u, v)$, define a measure of similarity between pairs of nodes u and v, where higher scores indicate higher similarity. Section 4 provides more details on graph construction.

We consider the problem of partitioning the graph into two disjoint sets of nodes A and B. We aim to minimize the cut, which is defined to be the sum of the crossing edges between the two sets of nodes. In other words, we want to split the sentences into two maximally dissimilar classes by choosing A and B to minimize:

$$cut(A, B) = \sum_{u \in A, v \in B} w(u, v)$$

However, we need to ensure that the two partitions are not only maximally different from each other, but also that they are themselves homogeneous by accounting for intra-partition node similarity. We formulate this requirement in the framework of normalized cuts (Shi and Malik, 2000), where the cut value is normalized by the volume of the corresponding partitions. The volume of the partition is the sum of its edges to the whole graph:

$$vol(A) = \sum_{u \in A, v \in V} w(u, v)$$

The normalized cut criterion ($Ncut$) is then defined as follows:

$$Ncut(A, B) = \frac{cut(A, B)}{vol(A)} + \frac{cut(A, B)}{vol(B)}$$

By minimizing this objective we simultaneously minimize the similarity across partitions and maximize the similarity within partitions. This formulation also allows us to decompose the objective into a sum of individual terms, and formulate a dynamic programming solution to the multiway cut problem.

This criterion is naturally extended to a k-way normalized cut:

$$Ncut_k(V) = \frac{cut(A_1, V - A_1)}{vol(A_1)} + \ldots + \frac{cut(A_k, V - A_k)}{vol(A_k)}$$

where $A_1 \ldots A_k$ form a partition of the graph, and $V - A_k$ is the set difference between the entire graph and partition k.

Decoding Papadimitriou proved that the problem of minimizing normalized cuts on graphs is NP-complete (Shi and Malik, 2000). However, in our case, the multi-way cut is constrained to preserve the linearity of the segmentation. By segmentation linearity, we mean that all of the nodes between the leftmost and the rightmost nodes of a particular partition have to belong to that partition. With this constraint, we formulate a dynamic programming algorithm for exactly finding the minimum normalized multiway cut in polynomial time:

$$C[i, k] = \min_{j<k} \left[C[i-1, j] + \frac{cut[A_{j,k}, V - A_{j,k}]}{vol[A_{j,k}]} \right] \quad (1)$$

$$B[i, k] = \operatorname*{argmin}_{j<k} \left[C[i-1, j] + \frac{cut[A_{j,k}, V - A_{j,k}]}{vol[A_{j,k}]} \right] \quad (2)$$

$$\text{s.t. } C[0, 1] = 0, C[0, k] = \infty, \; 1 < k \le N \quad (3)$$

$$B[0, k] = 1, \; 1 \le k \le N \quad (4)$$

$C[i, k]$ is the normalized cut value of the optimal segmentation of the first k sentences into i segments. The i-th segment, $A_{j,k}$, begins at node u_j and ends at node u_k. $B[i, k]$ is the back-pointer table from which we recover the optimal sequence of segment boundaries. Equations 3 and 4 capture respectively the condition that the normalized cut value of the trivial segmentation of an empty text into one segment is zero and the constraint that the first segment starts with the first node.

The time complexity of the dynamic programming algorithm is $O(KN^2)$, where K is the number of partitions and N is the number of nodes in the graph or sentences in the transcript.

4 Building the Graph

Clearly, the performance of our model depends on the underlying representation, the definition of the pairwise similarity function, and various other model parameters. In this section we provide further details on the graph construction process.

Preprocessing Before building the graph, we apply standard text preprocessing techniques to the text. We stem words with the Porter stemmer (Porter, 1980) to alleviate the sparsity of word counts through stem equivalence classes. We also remove words matching a prespecified list of stop words.

Graph Topology As we noted in the previous section, the normalized cut criterion considers long-term similarity relationships between nodes. This effect is achieved by constructing a fully-connected graph. However, considering all pairwise relations in a long text may be detrimental to segmentation accuracy. Therefore, we discard edges between sentences exceeding a certain threshold distance. This reduction in the graph size also provides us with computational savings.

Similarity Computation In computing pairwise sentence similarities, sentences are represented as vectors of word counts. Cosine similarity is commonly used in text segmentation (Hearst, 1994). To avoid numerical precision issues when summing a series of very small scores, we compute exponentiated cosine similarity scores between pairs of sentence vectors:

$$w(s_i, s_j) = e^{\frac{s_i \cdot s_j}{||s_i|| \times ||s_j||}}$$

We further refine our analysis by smoothing the similarity metric. When comparing two sentences, we also take into account similarity between their immediate neighborhoods. The smoothing is achieved by adding counts of words that occur in adjoining sentences to the current sentence feature vector. These counts are weighted in accordance to their distance from the current sentence:

$$\tilde{s}_i = \sum_{j=i}^{i+k} e^{-\alpha(j-i)} s_j,$$

where s_i are vectors of word counts, and α is a parameter that controls the degree of smoothing.

In the formulation above we use sentences as our nodes. However, we can also represent graph nodes with non-overlapping blocks of words of fixed length. This is desirable, since the lecture transcripts lack sentence boundary markers, and short utterances can skew the cosine similarity scores. The optimal length of the block is tuned on a heldout development set.

Lexical Weighting Previous research has shown that weighting schemes play an important role in segmentation performance (Ji and Zha, 2003; Choi et al., 2001). Of particular concern are words that may not be common in general English discourse but that occur throughout the text for a particular lecture or subject. For example, in a lecture about support vector machines, the occurrence of the term “SVM” is not going to convey a lot of information about the distribution of sub-topics, even though it is a fairly rare term in general English and bears much semantic content. The same words can convey varying degrees of information across different lectures, and term weighting specific to individual lectures becomes important in the similarity computation.

		Segments per	Total Word	ASR WER
Corpus	Lectures	Lecture	Tokens	Accuracy
Physics	33	5.9	232K	19.4%
AI	22	12.3	182K	×

Table 1: Lecture Corpus Statistics

In order to address this issue, we introduce a variation on the *tf-idf* scoring scheme used in the information-retrieval literature (Salton and Buckley, 1988). A transcript is split uniformly into N chunks; each chunk serves as the equivalent of documents in the *tf-idf* computation. The weights are computed separately for each transcript, since topic and word distributions vary across lectures.

5 Evaluation Set-Up

In this section we present the different corpora used to evaluate our model and provide a brief overview of the evaluation metrics. Next, we describe our human segmentation study on the corpus of spoken lecture data.

5.1 Parameter Estimation

A heldout development set of three lectures is used for estimating the optimal word block length for representing nodes, the threshold distances for discarding node edges, the number of uniform chunks for estimating *tf-idf* lexical weights, the alpha parameter for smoothing, and the length of the smoothing window. We use a simple greedy search procedure for optimizing the parameters.

5.2 Corpora

We evaluate our segmentation algorithm on three sets of data. Two of the datasets we use are new segmentation collections that we have compiled for this study,[1] and the remaining set includes a standard collection previously used for evaluation of segmentation algorithms. Various corpus statistics for the new datasets are presented in Table 1. Below we briefly describe each corpus.

Physics Lectures Our first corpus consists of spoken lecture transcripts from an undergraduate

[1] Our materials are publicly available at `http://www.csail.mit.edu/~igorm/acl06.html`

Physics class. In contrast to other segmentation datasets, our corpus contains much longer texts. A typical lecture of 90 minutes has 500 to 700 sentences with 8500 words, which corresponds to about 15 pages of raw text. We have access both to manual transcriptions of these lectures and also output from an automatic speech recognition system. The word error rate for the latter is 19.4%,[2] which is representative of state-of-the-art performance on lecture material (Leeuwis et al., 2003).

The Physics lecture transcript segmentations were produced by the teaching staff of the introductory Physics course at the Massachusetts Institute of Technology. Their objective was to facilitate access to lecture recordings available on the class website. This segmentation conveys the high-level topical structure of the lectures. On average, a lecture was annotated with six segments, and a typical segment corresponds to two pages of a transcript.

Artificial Intelligence Lectures Our second lecture corpus differs in subject matter, lecturing style, and segmentation granularity. The graduate Artificial Intelligence class has, on average, twelve segments per lecture, and a typical segment is about half of a page. One segment roughly corresponds to the content of a slide. This time the segmentation was obtained from the lecturer herself. The lecturer went through the transcripts of lecture recordings and segmented the lectures with the objective of making the segments correspond to presentation slides for the lectures.

Due to the low recording quality, we were unable to obtain the ASR transcripts for this class. Therefore, we only use manual transcriptions of these lectures.

Synthetic Corpus Also as part of our analysis, we used the synthetic corpus created by Choi (2000) which is commonly used in the evaluation of segmentation algorithms. This corpus consists of a set of concatenated segments randomly sampled from the Brown corpus. The length of the segments in this corpus ranges from three to eleven sentences. It is important to note that the lexical transitions in these concatenated texts are very sharp, since the segments come from texts written in widely varying language styles on completely different topics.

5.3 Evaluation Metric

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 on an equal basis with near misses.

Both of these metrics are defined with respect to the average segment length of texts and exhibit high variability on real data. We follow Choi (2000) and compute the mean segment length used in determining the parameter k on each reference text separately.

We also plot the Receiver Operating Characteristic (ROC) curve to gauge performance at a finer level of discrimination (Swets, 1988). The ROC plot is the plot of the true positive rate against the false positive rate for various settings of a decision criterion. In our case, the true positive rate is the fraction of boundaries correctly classified, and the false positive rate is the fraction of non-boundary positions incorrectly classified as boundaries. In computing the true and false positive rates, we vary the threshold distance to the true boundary within which a hypothesized boundary is considered correct. Larger areas under the ROC curve of a classifier indicate better discriminative performance.

5.4 Human Segmentation Study

Spoken lectures are very different in style from other corpora used in human segmentation studies (Hearst, 1994; Galley et al., 2003). We are interested in analyzing human performance on a corpus of lecture transcripts with much longer texts and a less clear-cut concept of a sub-topic. We define a segment to be a sub-topic that signals a prominent shift in subject matter. Disregarding this sub-topic change would impair the high-level understanding of the structure and the content of the lecture.

As part of our human segmentation analysis, we asked three annotators to segment the Physics lecture corpus. These annotators had taken the class in the past and were familiar with the subject matter under consideration. We wrote a detailed instruction manual for the task, with annotation guidelines for the most part following the model used by Gruenstein et al. (2005). The annotators were instructed to segment at a level of granularity

[2] A speaker-dependent model of the lecturer was trained on 38 hours of lectures from other courses using the SUMMIT segment-based Speech Recognizer (Glass, 2003).

	O	A	B	C
MEAN SEG. COUNT	6.6	8.9	18.4	13.8
MEAN SEG. LENGTH	69.4	51.5	24.9	33.2
SEG. LENGTH DEV.	39.6	37.4	34.5	39.4

Table 2: Annotator Segmentation Statistics for the first ten Physics lectures.

REF/HYP	O	A	B	C
O	0	**0.243**	0.418	0.312
A	0.219	0	0.400	0.355
B	0.314	0.337	0	0.332
C	0.260	0.296	0.370	0

Table 3: P_k annotation agreement between different pairs of annotators.

that would identify most of the prominent topical transitions necessary for a summary of the lecture.

The annotators used the NOMOS annotation software toolkit, developed for meeting segmentation (Gruenstein et al., 2005). They were provided with recorded audio of the lectures and the corresponding text transcriptions. We intentionally did not provide the subjects with the target number of boundaries, since we wanted to see if the annotators would converge on a common segmentation granularity.

Table 2 presents the annotator segmentation statistics. We see two classes of segmentation granularities. The original reference (O) and annotator A segmented at a coarse level with an average of 6.6 and 8.9 segments per lecture, respectively. Annotators B and C operated at much finer levels of discrimination with 18.4 and 13.8 segments per lecture on average. We conclude that multiple levels of granularity are acceptable in spoken lecture segmentation. This is expected given the length of the lectures and varying human judgments in selecting relevant topical content.

Following previous studies, we quantify the level of annotator agreement with the P_k measure (Gruenstein et al., 2005).[3] Table 3 shows the annotator agreement scores between different pairs of annotators. P_k measures ranged from 0.24 and 0.42. We observe greater consistency at similar levels of granularity, and less so across the two classes. Note that annotator A operated at a level of granularity consistent with the original reference segmentation. Hence, the 0.24 P_k measure score serves as the benchmark with which we can compare the results attained by segmentation algorithms on the Physics lecture data.

EDGE CUTOFF						
	10	25	50	100	200	NONE
PHYSICS (MANUAL)						
PK	0.394	0.373	0.341	**0.295**	0.311	0.330
WD	0.404	0.383	0.352	**0.308**	0.329	0.350
PHYSICS (ASR)						
PK	0.440	0.371	0.343	0.330	**0.322**	0.359
WD	0.456	0.383	0.356	0.343	**0.342**	0.398
AI						
PK	0.480	0.422	0.408	0.416	**0.393**	0.397
WD	0.493	0.435	0.420	0.440	**0.424**	0.432
CHOI						
PK	0.222	**0.202**	0.213	0.216	0.208	0.208
WD	0.234	**0.222**	0.233	0.238	0.230	0.230

Table 4: Edges between nodes separated beyond a certain threshold distance are removed.

As an additional point of reference we note that the uniform and random baseline segmentations attain 0.469 and 0.493 P_k measure, respectively, on the Physics lecture set.

6 Experimental Results

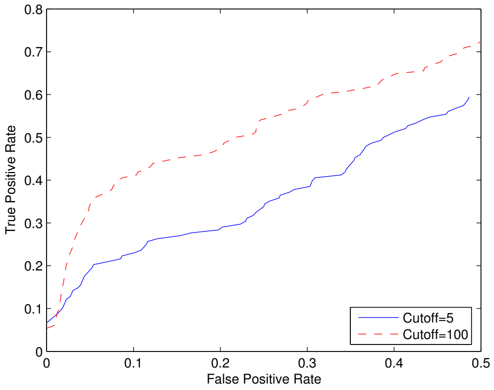

Figure 3: ROC plot for the Minimum Cut Segmenter on thirty Physics Lectures, with edge cutoffs set at five and hundred sentences.

Benefits of global analysis We first determine the impact of long-range pairwise similarity dependencies on segmentation performance. Our

[3] Kappa measure would not be the appropriate measure in this case, because it is not sensitive to near misses, and we cannot make the required independence assumption on the placement of boundaries.

	Choi	UI	MinCut
Physics (Manual)			
PK	0.372	0.310	**0.298**
WD	0.385	0.323	**0.311**
Physics (ASR transcripts)			
PK	0.361	0.352	**0.322**
WD	0.376	0.364	**0.340**
AI			
PK	0.445	**0.374**	0.383
WD	0.478	0.420	**0.417**
Choi			
PK	0.110	**0.105**	0.212
WD	0.121	**0.116**	0.234

Table 5: Performance analysis of different algorithms using the P_k and WindowDiff measures, with three lectures heldout for development.

key hypothesis is that considering long-distance lexical relations contributes to the effectiveness of the algorithm. To test this hypothesis, we discard edges between nodes that are more than a certain number of sentences apart. We test the system on a range of data sets, including the Physics and AI lectures and the synthetic corpus created by Choi (2000). We also include segmentation results on Physics ASR transcripts.

The results in Table 4 confirm our hypothesis — taking into account non-local lexical dependencies helps across different domains. On manually transcribed Physics lecture data, for example, the algorithm yields 0.394 P_k measure when taking into account edges separated by up to ten sentences. When dependencies up to a hundred sentences are considered, the algorithm yields a 25% reduction in P_k measure. Figure 3 shows the ROC plot for the segmentation of the Physics lecture data with different cutoff parameters, again demonstrating clear gains attained by employing long-range dependencies. As Table 4 shows, the improvement is consistent across all lecture datasets. We note, however, that after some point increasing the threshold degrades performance, because it introduces too many spurious dependencies (see the last column of Table 4). The speaker will occasionally return to a topic described at the beginning of the lecture, and this will bias the algorithm to put the segment boundary closer to the end of the lecture.

Long-range dependencies do not improve the performance on the synthetic dataset. This is expected since the segments in the synthetic dataset are randomly selected from widely-varying documents in the Brown corpus, even spanning different genres of written language. So, effectively, there are no genuine long-range dependencies that can be exploited by the algorithm.

Comparison with local dependency models We compare our system with the state-of-the-art similarity-based segmentation system developed by Choi (2000). We use the publicly available implementation of the system and optimize the system on a range of mask-sizes and different parameter settings described in (Choi, 2000) on a held-out development set of three lectures. To control for segmentation granularity, we specify the number of segments in the reference ("O") segmentation for both our system and the baseline. Table 5 shows that the Minimum Cut algorithm consistently outperforms the similarity-based baseline on all the lecture datasets. We attribute this gain to the presence of more attenuated topic transitions in spoken language. Since spoken language is more spontaneous and less structured than written language, the speaker needs to keep the listener abreast of the changes in topic content by introducing subtle cues and references to prior topics in the course of topical transitions. Non-local dependencies help to elucidate shifts in focus, because the strength of a particular transition is measured with respect to other local and long-distance contextual discourse relationships.

Our system does not outperform Choi's algorithm on the synthetic data. This again can be attributed to the discrepancy in distributional properties of the synthetic corpus which lacks coherence in its thematic shifts and the lecture corpus of spontaneous speech with smooth distributional variations. We also note that we did not try to adjust our model to optimize its performance on the synthetic data. The smoothing method developed for lecture segmentation may not be appropriate for short segments ranging from three to eleven sentences that constitute the synthetic set.

We also compared our method with another state-of-the-art algorithm which does not explicitly rely on pairwise similarity analysis. This algorithm (Utiyama and Isahara, 2001) (UI) computes the optimal segmentation by estimating changes in the language model predictions over different partitions. We used the publicly available implementation of the system that does not require parameter tuning on a heldout development set.

Again, our method achieves favorable performance on a range of lecture data sets (See Table 5), and both algorithms attain results close to the range of human agreement scores. The attractive feature of our algorithm, however, is robustness to recognition errors — testing it on the ASR transcripts caused only 7.8% relative increase in P_k measure (from 0.298 to 0.322), compared to a 13.5% relative increase for the UI system. We attribute this feature to the fact that the model is less dependent on individual recognition errors, which have a detrimental effect on the local segment language modeling probability estimates for the UI system. The block-level similarity function is not as sensitive to individual word errors, because the partition volume normalization factor dampens their overall effect on the derived models.

7 Conclusions

In this paper we studied the impact of long-range dependencies on the accuracy of text segmentation. We modeled text segmentation as a graph-partitioning task aiming to simultaneously optimize the total similarity within each segment and dissimilarity across various segments. We showed that global analysis of lexical distribution improves the segmentation accuracy and is robust in the presence of recognition errors. Combining global analysis with advanced methods for smoothing (Ji and Zha, 2003) and weighting could further boost the segmentation performance.

Our current implementation does not automatically determine the granularity of a resulting segmentation. This issue has been explored in the past (Ji and Zha, 2003; Utiyama and Isahara, 2001), and we will explore the existing strategies in our framework. We believe that the algorithm has to produce segmentations for various levels of granularity, depending on the needs of the application that employs it.

Our ultimate goal is to automatically generate tables of content for lectures. We plan to investigate strategies for generating titles that will succinctly describe the content of each segment. We will explore how the interaction between the generation and segmentation components can improve the performance of such a system as a whole.

8 Acknowledgements

The authors acknowledge the support of 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 Masao Utiyama for providing us with an implementation of his segmentation system and Alex Gruenstein for assisting us with the NOMOS toolkit. We are grateful to David Karger for an illuminating discussion on the Minimum Cut algorithm. We also would like to acknowledge the MIT NLP and Speech Groups, the three annotators, and the three anonymous reviewers for valuable comments, suggestions, and help.

References

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

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

F. Y. Y. Choi. 2000. Advances in domain independent linear text segmentation. In *Proceedings of the NAACL*, 26–33.

K. W. Church. 1993. Char_align: A program for aligning parallel texts at the character level. In *Proceedings of the ACL*, 1–8.

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

J. R. Glass. 2003. A probabilistic framework for segment-based speech recognition. *Computer Speech and Language*, 17(2–3):137–152.

A. Gruenstein, J. Niekrasz, M. Purver. 2005. Meeting structure annotation: Data and tools. In *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, 117–127.

M. A. K. Halliday, R. Hasan. 1976. *Cohesion in English*. Longman, London.

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

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

A. Kehagias, P. Fragkou, V. Petridis. 2003. Linear text segmentation using a dynamic programming algorithm. In *Proceedings of the EACL*, 171–178.

E. Leeuwis, M. Federico, M. Cettolo. 2003. Language modeling and transcription of the ted corpus lectures. In *Proceedings of ICASSP*, 232–235.

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

M. F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.

J. Reynar. 1998. *Topic segmentation: Algorithms and applications*. Ph.D. thesis, University of Pennsylvania.

G. Salton, C. Buckley. 1988. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523.

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

J. Swets. 1988. Measuring the accuracy of diagnostic systems. *Science*, 240(4857):1285–1293.

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