Discriminative Topic Segmentation of Text and Speech

Mehryar Mohri
Courant Institute and Google
mohri@cs.nyu.edu

Pedro Moreno
Google
pedro@google.com

Eugene Weinstein
Google
weinstein@google.com

Abstract

We explore automated discovery of topically-coherent segments in speech or text sequences. We give two new discriminative topic segmentation algorithms which employ a new measure of text similarity based on word co-occurrence. Both algorithms function by finding extrema in the similarity signal over the text, with the latter algorithm using a compact support-vector based description of a window of text or speech observations in word similarity space to overcome noise introduced by speech recognition errors and off-topic content. In experiments over speech and text news streams, we show that these algorithms outperform previous methods. We observe that topic segmentation of speech recognizer output is a more difficult problem than that of text streams; however, we demonstrate that by using a lattice of competing hypotheses rather than just the one-best hypothesis as input to the segmentation algorithm, the performance of the algorithm can be improved.

1 Introduction

Natural language streams, such as news broadcasts and telephone conversations, are marked with the presence of underlying topics that influence the statistics of the speech produced. Learning to identify the topic underlying a given segment of speech or text, or to detect boundaries between topics is beneficial in a number of ways. We describe our work on topic segmentation, defined here as automatic division of a stream of text or speech into topic-coherent segments. Topic segmentation is an important task in text and speech processing, with many potential applications. For example, knowledge of the topic-wise segmentation can be used to enable effective presentation of the underlying word stream and to improve speech transcription quality through the use of a speech recognizer with a topic-dependent language model (Lane *et al.*, 2005). It can also be used to improve navigation of audio and video collections, by enabling the consideration of topic-coherent segment closeness as a feature when creating links between items. Finally, in real-time speech recognition applications, topicality information can be used to improve the dialogue path selected by the system (Riccardi *et al.*, 1997).

We address specifically the case when the input to the topic segmentation algorithm is a speech audio sequence. Speech-to-text transcription of audio streams is a process inherently marked with errors and uncertainty, which results in difficulties for algorithms trying to discover topical structure. We create novel algorithms for topic segmentation that use word co-occurrence statistics to evaluate topic-coherence between pairs of adjacent windows over the speech or text stream and hypothesize segment boundaries at extrema in the similarity signal. These algorithms make local decisions about topic-coherence, rather than requiring the entire speech or text document to be analyzed globally in order to produce a segmentation. Hence, they are well suited for use in tasks where it is important to process the input speech or text in an online way, such as for streaming news broadcasts or telephone conversations.

To improve algorithm robustness in the presence of off-topic content in a generally topic-coherent observation stream, we employ the use of a compact support-vector based description of a window of text or speech observation learned discriminatively in word similarity space. In empirical trials, we demonstrate that this approach results in a more accurate topic segmentation algorithm that also outperforms a modern generative learning technique, the hidden topic Markov model (HTMM) (Gruber *et al.*, 2007), in the topic segmentation task. We further demonstrate that incorporating competing speech recognition hypotheses, rather then only the one-best, into the topic segmentation algorithm can result in an improvement in segmentation quality.

Appearing in Proceedings of the 13^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 9 of JMLR: W&CP 9.

1.1 Previous Work

Topic models or topic labeling algorithms assign a topic label sequence to a stream of text or speech. Since a topic assignment to a stream of text or speech also implies a topic-wise segmentation of the stream, these algorithms are also topic segmentation algorithms. Much of the recent work on topic analysis has been focused on generative models, in which a text sequence is explained by a latent sequence of topic labels. Let $V = \{w_1, w_2, \ldots, w_n\}$ be the vocabulary of n words. Then an *observation* a is an observed set of text or speech expressed through the empirical frequency, or expected count, $C_a(w_i)$ for each $w_i \in V$. A simple generative formulation of a topic model is

$$z = \arg\max_z \Pr(z|a) = \arg\max_z \Pr(a|z)\Pr(z), \quad (1)$$

where z is the topic label assigned. Under such topic models, the observation sequence is labeled by decoding the maximum *a posteriori* sequence of topics accounting for the observations. In these models, a is treated as a "bag of words," meaning the order of the words in the text or speech stream underlying a is generally not considered. In practice, a can correspond to a sentence, a window over a text, an utterance, or a single word. In Latent Dirichlet Allocation (LDA) (Blei *et al.*, 2003), the formulation of equation 1 is used, but the distributions $\Pr(a|z)$ and $\Pr(z)$ are modeled as multinomial distributions with Dirichlet priors. Hidden Topic Markov Models (HTMMs) (Gruber *et al.*, 2007) use an HMM structure where each state corresponds to a topic z and an underlying topic model (such as LDA or n-gram), as in Blei and Moreno (2001); Yamron *et al.* (1998).

Generative topic analysis algorithms such as LDA and HTMM attempt to model the distribution of words in a particular topic, the distribution of topic-to-topic transitions, and/or the global distribution of topic labels. Certainly if one can accurately model the distribution of the underlying topic sequence, one can also easily solve the problem of topic segmentation or any other related problem. However, our goal in this work is simpler – to arrive at the best topic-wise segmentation of a stream of text or speech, and we endeavor to create an algorithm specifically designed for this problem. A number of efforts have been made to create algorithms specifically for the segmentation task. In TextTiling (Hearst, 1997), word counts are computed for a sliding window over the input text. Text similarity is then evaluated between pairs of adjacent windows according to a cosine similarity measure, $\frac{\sum_{i=1}^{n} C_1(w_i)C_2(w_i)}{\sqrt{\sum_{i=1}^{n} C_1(w_i)^2 \sum_{i=1}^{n} C_2(w_i)^2}}$. The segmentation is obtained by thresholding this similarity function. In this approach, words that are naturally more prevalent in the corpus effectively receive a higher weight in the cosine score. One popular way to bypass this limitation is by using the term frequency–inverse document frequency (tf–idf) (Salton and Buckley, 1988) to weight each word's contribution to the similarity score. However, even with tf–idf weighting, considering words in isolation for topic segmentation results in a natural impairment of the algorithms created. Word pair similarity can be used to move beyond this limitation (Kozima, 1993). It is also possible to view the topic segmentation task as a binary classification problem at every possible segment boundary, with maximum entropy models a popular classifier choice (Beeferman *et al.*, 1999; Reynar, 1999).

The window-based approaches just mentioned segment a stream of observations by topic by making local decisions about whether adjacent sets of observations are similar. However, in recent years several approaches have been developed for making globally-optimal decisions about topic segmentation by analyzing the whole document to be segmented. In Utiyama and Isahara (2001), the authors model topic-coherence by the repetition frequency of words within a segment, and posit the task of optimally segmenting a document as finding a minimum-cost path in a weighted graph. Ji and Zha (2003) used cosine distance as a between-sentence similarity measure to obtain an image representing the sentence closeness matrix, and employed image processing algorithms to refine the matrix so that a topical segmentation can be found using a dynamic programming-based search. In Malioutov and Barzilay (2006), the authors also used cosine distance and modeled the segmentation problem as a graph, but proposed an efficient way to find the minimum-cost partition of the graph, which allowed them to take into account long-range topic coherence within a segment. The algorithm presented in the last work can be viewed as an application of clustering to text observations, where the number of partitions K of the text stream must be decided in advance. In addition, all three approaches just mentioned are global approaches, and as we have already mentioned, this mandates that they cannot be used in an online text or speech processing setting, which limits their applicability in real-world tasks. In contrast with these approaches, the algorithms we will present make local decisions about topic-coherence, and are thus broadly applicable in online as well as offline settings.

2 Measuring Topical Similarity

Let the input to a segmentation algorithm be a sequence of observations $T = (x_1, \ldots, x_m)$. We refer to the correct segmentation provided by human judges of topicality or some other oracle as the reference, and that provided by a topic segmentation algorithm to be evaluated as the hypothesis. The most popular topic segmentation quality measure used in past work is known as the Co-occurrence Agreement Probability, or CoAP. CoAP (Beeferman *et al.*, 1999) is broadly defined as $P_D(\mathrm{ref}, \mathrm{hyp}) = \sum_{1 \le i \le j \le m} D(i,j)\,(\delta_{\mathrm{ref}}(i,j) \overline{\oplus} \delta_{\mathrm{hyp}}(i,j))$, where $D(i,j)$

is a distance probability distribution over observations i, j; $\delta_{\text{ref}}(i,j)$ and $\delta_{\text{hyp}}(i,j)$ are indicator functions that are one if observations i and j are in the same topic in the reference and hypothesis segmentations, respectively; and $\overline{\oplus}$ is the exclusive NOR operation ("both or neither"). In practice, the choice of D is almost always the distribution with its mass placed entirely on one distance k. CoAP scoring is then reduced to a single fixed-size sliding window over the observations. CoAP is marked with several limitations, such as that it functions purely by analyzing the segmentation of the observations into topic-coherent chunks without taking into account the content of the chunks labeled as topic-coherent, that it depends on the choice of window size k and that it implicitly requires that the reference and the hypothesis segmentations are obtained by placing boundaries in the same stream of text.

To develop an improved topic segmentation quality measure and novel segmentation algorithms, we seek a general similarity function between segments of text and speech. One rudimentary similarity function is the cosine distance. However, this is based on evaluating the divergence in empirical frequency for a given word between the two segments. For example, if the first segment being considered has many occurrences of "sport", then a segment making no mention of "sport" but mentioning "baseball" frequently might be assigned the same similarity score as a segment not mentioning anything relevant to sports at all. To develop a more robust approach, let $x, y \in V$ be two words. If T is a training corpus, then let $C_T(x,y)$, $C_T(x)$, and $C_T(y)$ be the empirical probabilities (i.e., the counts normalized by the total number of words in T) of x and y appearing together in the same topic-coherent chunk, and that of x and y appearing, in T, respectively. A similarity measure between words is $\text{sim}(x,y) = \frac{C_T(x,y)}{C_T(x)C_T(y)}$, which is just the pointwise mutual information (PMI) (Church and Hanks, 1990) without the logarithm.

We will evaluate the total similarity of a pair of observations a and b as $K(a,b) = \sum_{w_1 \in a, w_2 \in b} C_a(w_1)\ C_b(w_2)\ \text{sim}(w_1, w_2)$. Let A and B be the column vectors of empirical word frequencies such that $A_i = C_a(w_i)$ and $B_i = C_b(w_i)$ for $i = 1, \ldots, n$. Let $\mathbf{K}$ be the matrix such that $\mathbf{K}_{i,j} = \text{sim}(w_i, w_j)$. The similarity score can then be written as a matrix operation, $K(a,b) = A^\top \mathbf{K} B$. We normalize to ensure that the score is in the range $[0,1]$ and that for any input, the self-similarity is 1,

$$K_{norm}(a,b) = \frac{A^\top \mathbf{K} B}{\sqrt{(A^\top \mathbf{K} A)(B^\top \mathbf{K} B)}}. \tag{2}$$

Proposition 1. *K_{norm} is a positive-definite symmetric (PDS) kernel.*

Proof. In the following, the empirical frequencies and expectations are computed as before over a training corpus T. For notational simplicity we omit the subscript T. Let 1_{w_i} be the indicator function of the event "w_i occurred." Then

$$\begin{aligned} \mathbf{K}_{ij} &= \frac{C(w_i, w_j)}{C(w_i)C(w_j)} = \frac{\mathbf{E}[1_{w_i} 1_{w_j}]}{\mathbf{E}[1_{w_i}]\,\mathbf{E}[1_{w_j}]} \\ &= \mathbf{E}\left[\frac{1_{w_i}}{\mathbf{E}[1_{w_i}]}\frac{1_{w_j}}{\mathbf{E}[1_{w_j}]}\right]. \end{aligned} \tag{3}$$

Clearly K is symmetric. Recall that for two random variables X and Y, we have $\text{Cov}(X,Y) = \mathbf{E}[XY] - \mathbf{E}[X]\,\mathbf{E}[Y]$ and observe that for all $i, \mathbf{E}\left[1_{w_i}/\,\mathbf{E}[1_{w_i}]\right] = 1$. Thus we have

$$\text{Cov}\left(\frac{1_{w_i}}{\mathbf{E}[1_{w_i}]}, \frac{1_{w_j}}{\mathbf{E}[1_{w_j}]}\right) = \mathbf{E}\left[\frac{1_{w_i}}{\mathbf{E}[1_{w_i}]}\frac{1_{w_j}}{\mathbf{E}[1_{w_j}]}\right] - 1 \tag{4}$$

Next, recall that any covariance matrix is positive semidefinite. Applying this fact to the covariance matrix of equation 4, we get

$$\sum_{i,j=1}^{m} c_i c_j\, \mathbf{E}\left[\frac{1_{w_i}}{\mathbf{E}[1_{w_i}]}\frac{1_{w_j}}{\mathbf{E}[1_{w_j}]}\right] - \sum_{i,j=1}^{m} c_i c_j \geq 0. \tag{5}$$

Now, let $\mathbf{1}$ and C denote column vectors of size m such that $\mathbf{1}_i = 1$ and $C_i = c_i$ for $i = 1, \ldots, m$. Then,

$$\begin{aligned} \sum_{i,j=1}^{m} c_i c_j &= \text{Tr}(CC^\top \mathbf{1}\mathbf{1}^\top) = \text{Tr}(C^\top \mathbf{1}\mathbf{1}^\top C) \\ &= \text{Tr}((C^\top \mathbf{1})^2) \geq 0. \end{aligned} \tag{6}$$

Combining equations 5 and 6, we get

$$\sum_{i,j=1}^{m} c_i c_j\, \mathbf{E}\left[\frac{1_{w_i}}{\mathbf{E}[1_{w_i}]}\frac{1_{w_j}}{\mathbf{E}[1_{w_j}]}\right] \geq \sum_{i,j=1}^{m} c_i c_j \geq 0. \tag{7}$$

This shows that the $\mathbf{K}$ is positive semidefinite. If $K(a,b) = A^\top \mathbf{K} B$, where A is a column vector of counts, $A = (C_a(w_1), \ldots, C_a(w_N))^\top$, and similarly with B, then $K(a,b) =< \mathbf{K}^{1/2} A, \mathbf{K}^{1/2} B >$. Hence K is a PDS kernel. Normalization preserves PDS, so K_{norm} is also a PDS kernel. □

This property of K_{norm} enables us to map word observations into a similarity feature space for the support vector based topic segmentation algorithm we will present in Section 3.1. However, we have also in previous work used this general measure of similarity for text to create a topic segmentation quality measure that we call the Topic Closeness Measure (TCM) (Mohri *et al.*, 2009; Weinstein, 2009). TCM overcomes the limitations of CoAP discussed above, and as we shall see in Section 5, correlates strongly with CoAP in empirical trials.

Let k and l be the number of segments in the reference and hypothesis segmentation, respectively. Additionally, let $r_1, \ldots, r_k$ and $h_1, \ldots, h_l$ be the segments in the reference and hypothesis segmentation, respectively. $Q(i,j)$

quantifies the overlap between the two segments i, j. In this work, $Q(i, j)$ is the indicator variable that is one when reference segment i overlaps with hypothesis segment j, and zero otherwise. However, various other functions can be used for Q, such as the duration of the overlap or the number of overlapping sentences or utterances. Similarly, other similarity scoring functions can be incorporated in place of K_{norm}. The Topic Closeness Measure (TCM) between the reference segmentation R and the hypothesis segmentation H is defined as

$$\mathrm{TCM}(R, H) = \frac{\sum_{i=1}^{k} \sum_{j=1}^{l} Q(i, j) K_{norm}(r_i, h_j)}{\sum_{i=1}^{k} \sum_{j=1}^{l} Q(i, j)}. \quad (8)$$

3 New Topic Segmentation Algorithms

Let the input be a sequence of observations $T = (x_1, \ldots, x_m)$. Then a topic segmentation algorithm must decide the set b of topic boundaries in T, that is the set of indices i such that x_i and x_{i+1} belong to different topics. For $i \in \{\delta, \ldots, m\}$, we will refer to a *window* of observations of size δ ending at i as the set $w_i = \{x_{i-\delta+1}, \ldots, x_i\}$. The windowing of an observation stream is illustrated in Figure 1(a). Let $s_i = s(w_i, w_{i+\delta})$ be either a similarity score or a distance between w_i and $w_{i+\delta}$. If s_i is a similarity score, we can hypothesize a segment boundary where the similarity signal dips below a global threshold θ to define the boundary set $b = \{i : s_i < \theta\}$. Because the range of s on either side of a true boundary might vary, a more robust segmentation algorithm is to look for range extrema in s. This is accomplished by passing a window of size δ over s and hypothesizing boundaries where minima or maxima occur, depending on whether s is a similarity or distance score. Let $s_i = K_{norm}(w_i, w_{i+\delta})$. Further, to denote the minimum and maximum of a range of s values, let $\mathrm{rmax}(s, i, j) = \max(s_i, \ldots, s_j)$ and $\mathrm{rmin}(s, i, j) = \min(s_i, \ldots, s_j)$. Then we obtain a segmentation algorithm by detecting range minima in s, and applying the absolute threshold θ to each range minimum, $b = \{i : s_i < \theta \wedge s_i = \mathrm{rmin}(s, i - \lfloor \delta/2 \rfloor, i + \lfloor \delta/2 \rfloor)\}$. This simple search for range extrema, combined with the use of K_{norm} to evaluate similarity, results in a novel segmentation algorithm, to which we will refer as the similarity-based segmentation algorithm.

3.1 Support Vector Topic Segmentation Algorithm

One disadvantage of using the verbatim empirical word distribution to compare windows of observations, as in the above algorithm, is that speech recognition errors or various spoken language phenomena might result in off-topic content within a generally topic-coherent stream of observations, resulting in a reduction in segmentation quality. To combat this drawback, we seek a description of the text or speech being considered that is able to discriminate between the observations belonging to the true distribution and noise or outlier observations, but without attempting to learn the distribution. The sphere-based descriptors of Tax and Duin (1999) attempt to find a sphere in feature space that encloses the true data from a given class but excludes outliers within that class and data from other classes. An alternative approach of Schölkopf *et al.* (1999) posits this task as separating data from the origin in feature space, a problem that is equivalent to the spheres problem for many kernels, including the one used in this work. This problem is often referred to as one-class classification, and because the problem formulation resembles that of support vector machines (SVM) (Cortes and Vapnik, 1995; Vapnik, 1998), often as the one-class SVM.

More formally, given a set of observations $x_1, \ldots, x_m \in \mathcal{X}$, our task is to find a ball or sphere that, by enclosing the observations in feature space, represents a compact description of the data. We assume the existence of a mapping of observations into a feature space, $\Phi \colon \mathcal{X} \mapsto F$. This results in the existence of a kernel operating on a pair of observations, $K(x, y) = \Phi(x) \cdot \Phi(y)$. A sphere in feature space is then parametrized by a center $c \in F$ and radius $R \in \mathbb{R}$. We allow each observation x_i to lie outside the sphere by a distance ξ_i, at the cost of incurring a penalty in the objective function. The optimization problem written in the form of Schölkopf *et al.* (1999) is

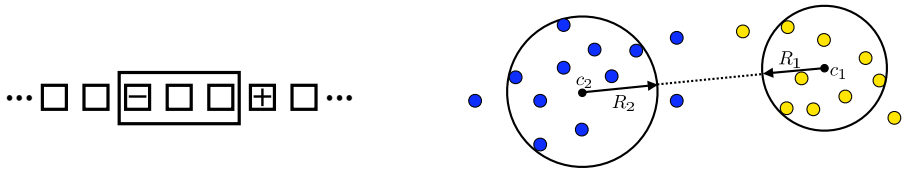
$$\begin{aligned} \min_{R \in \mathbb{R}, \boldsymbol{\xi} \in \mathbb{R}^m, c \in F} \quad & R^2 + \frac{1}{\nu m} \sum_i \xi_i \\ \text{subject to} \quad & \|\Phi(x_i) - c\|^2 \leq R^2 + \xi_i,\ \xi_i \geq 0, \\ & i \in [1, m]. \end{aligned} \quad (9)$$

The objective function attempts to keep the size of the sphere small, while reducing the total amount by which outlier observations violate the sphere constraint. The parameter ν controls the tradeoff between these two goals. Using standard optimization techniques, we can write the Lagrangian of this optimization problem using Lagrangian variables $\alpha_i \geq 0, i \in [0, m]$. Solving for c, we obtain $c = \sum_i \alpha_i \Phi(x_i)$. Substituting this back into the primal problem of equation 9, we obtain the dual problem, in which the kernel K takes the place of dot products between training observations,

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & \textstyle\sum_{i,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j) - \sum_{i=1}^{m} \alpha_i K(x_i, x_i) \\ \text{subject to} \quad & 0 \leq \alpha_i \leq \tfrac{1}{\nu m}, \quad \textstyle\sum_{i=1}^{m} \alpha_i = 1. \end{aligned} \quad (10)$$

By substitution into the equation of a sphere in feature space, the classifier then takes the form

$$\begin{aligned} f(x) \quad = \quad & \mathrm{sgn}\left(R^2 - \sum_{i,j=1}^{m} \alpha_i \alpha_j K(x_i, x_j) \right. \\ & \left. + 2 \sum_{i=1}^{m} \alpha_i K(x_i, x) - K(x, x) \right) \end{aligned} \quad (11)$$



(a) (b)

Figure 1: (a) An illustration of windowing a stream of observations. Each square represents an observation, and the rectangle represents the current position of the window. To advance the window one position, the window is updated to add the observation marked with $+$ and to remove that marked with $-$. (b) An illustration of two sets of observations being compared in feature space based on their sphere descriptors. The dashed line indicates the shortest distance between the two spheres.

The resulting data description is a combination of the support observations $\{x_i \colon \alpha_i \neq 0\}$. The radius can be recovered by finding the value that yields $f(x_{SV}) = 0$, where x_{SV} is any support observation. As pointed out in Schölkopf *et al.* (1999), for any kernel K such that $K(x, x)$ is a constant, the sphere problem described above has the same solution as the separation from the origin problem. We have $K_{norm}(x, x) = 1$, thus the condition for equivalence is met and thus our geometric description can be viewed as an instance of either problem.

The sphere data description yields a natural geometric formulation for comparing two sets of observation streams. To accomplish this, we calculate the geometric shortest distance in feature space between the two spheres representing them. This comparison is illustrated in Figure 1(b). Assume that we are comparing two windows of m observations (in the case of topic segmentation, two windows of m utterances or sentences), w_1 and w_2. Let $x_{1,1}, \ldots, x_{m_1,1}$ and $x_{1,2}, \ldots, x_{m_2,2}$ be the word frequency counts, and let $\alpha_{1,1}, \ldots, \alpha_{m_1,1}$ and $\alpha_{1,2}, \ldots, \alpha_{m_2,2}$ be the dual coefficients resulting from solving the optimization problem of equation 10, for w_1 and w_2, respectively. The resulting support vector descriptions are represented by spheres (c_1, R_1) and (c_2, R_2). Then, the distance between the centers of the spheres is the Euclidean distance in the feature space. Since the mapping $\Phi(\cdot)$ is implicitly expressed through the kernel $K(\cdot, \cdot)$, the distance is also computed by using the kernel, as

$$\begin{aligned} \|c_1 - c_2\|^2 &= \sum_{i,j=1}^{m_1} \alpha_{i,1}\alpha_{j,1}K(x_{i,1}, x_{j,1}) \\ &\quad + \sum_{i,j=1}^{m_2} \alpha_{i,2}\alpha_{j,2}K(x_{i,2}, x_{j,2}) \\ &\quad - 2\sum_{i=1}^{m_1}\sum_{j=1}^{m_2} \alpha_{i,1}\alpha_{j,2}K(x_{i,1}, x_{j,2}). \end{aligned} \tag{12}$$

The shortest distance between the spheres is simply obtained by subtracting the radii to obtain $\mathrm{dist}(w_1, w_2) = \|c_1 - c_2\| - (R_1 + R_2)$. Note that it is possible for the sphere descriptors to overlap, and in fact in practice this is frequently the case for adjacent windows of observations. Hence, the quantity $\mathrm{dist}(w_1, w_2)$ does not always represent a geometric margin, but nevertheless it can be viewed as an an algebraic measure of separation even if it is negative.

Distances between a pair of observations in the Hilbert space defined by K_{norm} represent the divergence in similarity between word pairs across two observations computed according to our co-occurrence based word similarity score. Since K_{norm} is a PDS kernel, the convexity of the optimization problems above is guaranteed. To construct our final discriminative topic segmentation algorithm, we simply set $s_i = \mathrm{dist}(w_1, w_2)$, and hypothesize segment boundaries in the observation at range maxima in the s signal, $b = \{i : s_i > \theta \wedge s_i = \mathrm{rmax}(s, i - \lfloor \delta/2 \rfloor, i + \lfloor \delta/2 \rfloor)\}$.

4 Lattice-based Topic Analysis

There is a significant literature on topic analysis of spoken language (e.g., Blei and Moreno (2001); Yamron *et al.* (1998)). However, the majority of these works use only the one-best recognition hypothesis as input to a topic labeling and/or segmentation algorithm. Since modern recognizers use the Viterbi sub-optimal beam search to approximate the most likely word sequence, there is always a beam of almost-as-likely hypotheses being considered. As a result, it is possible to produce a list, or more compactly, a graph, of the top hypotheses along with their likelihoods. Such a graph is known as a *lattice*. Lattices can be represented with finite automata, which enables compactness, lookup efficiency, and easy implementation of necessary lattice manipulations with general automata algorithms (Mohri, 1997). A recent work demonstrated an improvement using word and phoneme lattices for assigning topic labels to pre-segmented utterances in isolation (Hazen and Margolis, 2008). We focus exclusively on word lattices.

In our topic segmentation and labeling algorithms, we use two information sources derived from lattices, expected counts and confidence scores, as follows. The input to all

the algorithms we described is a sequence of bag-of-word observations. Let a be a set of observations and $f_a(w)$ the set of word weights to be computed. If no lattice information is available then a is represented by a bag-of-words of its one-best hypothesis, $(w_1, \ldots, w_l)$. Then $f_a(w) = \sum_{i=1}^{l} 1_{w_i=w}$. If a word lattice is available, we can compute for each word in the lattice a total posterior probability, or expected count, accumulated over all the paths that contain that word. If V is the vocabulary the expected count of the word w according to a stochastic lattice automaton A, i.e., that with the weights of all the paths summing to one, is $f_a(w) = \sum_{u \in V^*} |u|_w [\![A]\!](u)$, where $|u|_w$ is the number of occurrences of word w in string u and $[\![A]\!](u)$ is the probability associated by A to string u. The set of expected counts associated with all the words found in the lattice can be computed efficiently (Mohri, 2003). We also compute word-level confidence scores for the one-best hypothesis using a logistic regression classifier. The classifier takes two features as input, the word expected counts just mentioned and a likelihood ratio between the standard recognizer with full context-dependent acoustic models and a simple recognizer with context-independent models. If each word w_i has an associated confidence $c(w_i)$, then $f_a(w) = \sum_{i=1}^{l} c(w_i) 1_{w_i=w}$. Finally, the word weights $f_a(w)$ are normalized to produce the counts that serve as input to the segmentation algorithm, $C_a(w) = \frac{f_a(w)}{\sum_{w \in V} f_a(w)}$.

5 Experiments

Our experimental work has been in the context of the English portion of the TDT corpus of broadcast news speech and newspaper articles (Kong and Graff, 2005). The speech corpus consisted of 447 news show recordings of 30-60 minutes per show, for a total corpus size of around 311 hours, with human-labeled story boundaries treated here as topic boundaries. For the experiments involving speech data, we used 41 and 69 shows containing 957 and $1{,}674$ stories, for development and testing, respectively. The 337 remaining shows containing $6{,}310$ stories were used for training. For those experiments using text data, a total of $1{,}314$ training news streams were used, including the human transcripts of the 337 news shows already mentioned as well as 977 non-speech news streams, with a total of $15{,}110$ non-speech stories. The task in our empirical evaluation was to automatically reconstruct the story/topic boundaries from a stream of speech utterance transcriptions or text sentences. The HTMM was trained with 20 topics and with hyperparameters α and β set as in (Steyvers and Griffiths, 2007).

To evaluate our co-occurrence based similarity score empirically, we computed K_{norm} between all pairs of test and development stories with human topic labels such as "Earthquake in El Salvador." With 291 stories, there were $3{,}166$ same-topic story pairs and $39{,}172$ different-topic pairs in our experiment. The average pairwise similarity between different-topic story pairs was 0.2558 and that between same-topic story pairs was 0.7138, or around 2.8 times greater. This indicates that our text similarity measure is a good indicator of topical similarity between two segments of text or speech. We also explored the correlation between K_{norm} and true segmentation boundaries by processing each show's transcription by sliding a window of $\delta = 6$ sentences along the text and accumulating the word frequencies within each window. For each sentence t, let w_t be the window ending at sentence t. Figure 2 displays the plot of $K_{norm}(w_t, w_{t+\delta})$ for a representative show. As this figure illustrates, true topic boundaries are extremely well correlated with range minima in the similarity score. Similar trends are observed with other shows in the corpus.

5.1 Topic Segmentation Results

For the speech experiments, the audio for each show was first automatically segmented into utterances, while removing most non-speech audio, such as music and silence (Alberti *et al.*, 2009). Each utterance was transcribed using Google's large-vocabulary continuous speech recognizer (the baseline system of Alberti *et al.* (2009)). The word error rate of this recognizer on the standard 1997 Broadcast News evaluation set was 17.7%. The vocabulary for the HTMM algorithm consisted of a subset of $8{,}821$ words. This was constructed by starting with the set of words seen in the recognizer transcription of the training data, applying Porter stemming (Porter, 1980), removing a stoplist of function and other words not likely to indicate any topic, and keeping only those words occurring more than five times. The HTMM was trained with an Expectation Maximization (EM) algorithm initialized with random values for model parameters. To minimize the possibility of a particular randomization overfitting the test data, we ran 20 trials of model training and testing and picked the model that had the best segmentation quality on the development data set.

For the algorithms based on K_{norm}, the parameter set included the threshold θ and the window size δ for both algorithms. The sphere-distance based algorithm was additionally parametrized by the regularization tradeoff parameter ν. For both algorithms, we performed parameter selection on the development data set. Since these two algorithms rely on the use of the co-occurrence based word similarity score, their training consists of calculating word and word pair frequencies over a corpus of text segmented into topic-coherent chunks. The input to the training stage of all the segmentation algorithms were human transcriptions of the speech news broadcasts as well as non-speech news sources, as detailed above.

The experiment results are given in Tables 1 and 2. The former gives segmentation quality scores for degenerate

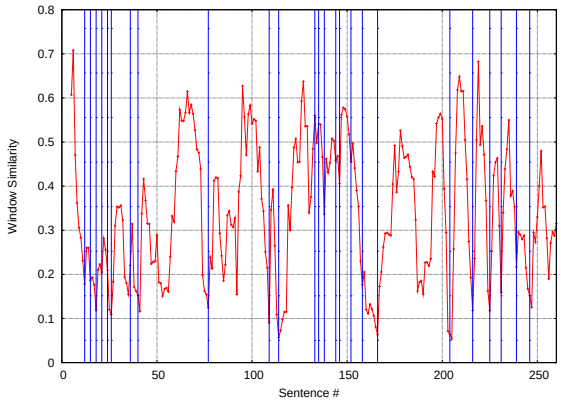


Figure 2: The distance $K_{norm}(w_t, w_{t+\delta})$ for a representative show. The vertical lines are true story boundaries. A line at sentence t denotes that sentence $t+1$ is from a new story.

segmentations with boundaries decided by a fair coin toss (Random), all possible boundaries (Full), and no boundaries at all (None). The results for the similarity-based segmentation algorithm and the sphere-distance based algorithm of Section 3.1 are denoted as Sim and SV, respectively. We tested on the reference text (Text), as well as three different varieties of speech transcriptions, transcriptions only (One-best), and speech transcriptions weighted with lattice expected counts (Counts) and confidence scores (Confidence).

5.2 Discussion

In the following, all comparisons are made in terms of relative error improvement. TCM error and CoAP error are both defined as $1 - \text{TCM}$ and $1 - \text{CoAP}$, respectively. In our experiments, segmentations produced by all three algorithms significantly outperform degenerate segmentations by both measures. The similarity-based segmentation algorithm does not show a consistent improvement over HTMM in the cases where the inputs are derived from speech recognition data. However, for text test data, this algorithm outperforms HTMM. This result verifies empirically that the variability in the word distribution introduced by speech recognition errors is a challenge for similarity-based segmentation algorithms. While the performance of the similarity-based algorithm degrades in the presence of speech recognition errors, the SV algorithm outperforms HTMM and Sim significantly across all test conditions by both measures. For example, compared to HTMM, segmentation error is reduced by 29.3% and 18.6% when we test on text data and by 10.3% and 11.7% when we test on one-best speech data, with CoAP and TCM, respectively. Improvements are additionally demonstrated by using lattice information in segmentation algorithms. For example, for HTMM, lattice counts yield a 2.3% relative improvement, in TCM error compared to the one-best baseline, 1.4% in terms of CoAP. Confidence scores also yield improvements with both measures, 7.0% relative by TCM and 9.4% by CoAP. The SV algorithm also achieves improvements when confidence scores are used, of 1.9% and 2.4% by CoAP and TCM over the one-best baseline.

Input Type	CoAP	TCM
Text Random	50.4%	58.4%
Text Full	50.4%	51.8%
Text None	49.6%	56.2%
One-best Random	50.8%	48.8%
One-best Full	51.0%	43.0%
One-best None	49.1%	52.9%

Table 1: CoAP and TCM measured on degenerate segmentations.

Input Type	Algorithm	Quality Measure	
		CoAP	TCM
Text	HTMM	66.9%	72.6%
	Sim	72.0%	75.0%
	SV	**76.6%**	**77.7%**
One-best	HTMM	65.0%	61.5%
	Sim	60.4%	62.8%
	SV	**68.6%**	**66.0%**
Counts	HTMM	65.5%	62.4%
	Sim	59.4%	63.4%
	SV	**68.5%**	**66.5%**
Confidence	HTMM	68.3%	64.2%
	Sim	59.7%	63.8%
	SV	**69.2%**	**66.8%**

Table 2: Topic segmentation quality.

As we have already mentioned, the algorithms described make local decisions about topic coherence. However, it is worth noting that if the entire document to be segmented is available in advance, global information can be incorporated into our algorithm as well. While our algorithm functions by thresholding locally the distance between adjacent sphere descriptors of windowed observations, we could create a global algorithm by, for example, placing the topic boundaries at those locations with the largest n gaps between adjacent windows of observations, or by using one of the graph-based segmentation methods mentioned in Section 1.1.

In addition, it should be noted that the algorithms presented are supervised, both since we compute co-occurrence frequencies on a corpus of pre-segmented observations and because we pick the algorithm parameters using accuracy on a development set. Though HTMM in general is unsupervised in that it does not use pre-segmented observations, it is also noteworthy that our use of HTMM can be considered supervised in that we pick the one HTMM out of 20 that gives the best accuracy on the development set. In addition, the Sim algorithm does represent a simpler supervised algorithm than the more sophisticated SV. In fact,

the Sim algorithm can be viewed as the supervised counterpart of cosine-distance based algorithms such as TextTiling. Hence, it is significant that SV yields a substantial improvement over both HTMM and Sim.

6 Conclusion

In this paper, we gave two new topic segmentation algorithms for speech content based on a general measure of topical similarity derived from word co-occurrence statistics. The first algorithm functions by comparing adjacent observation windows according to a similarity measure for words trained on co-occurrence statistics. The second is based on comparing compact geometric descriptions of the adjacent windows in topic similarity feature space. We have demonstrated both algorithms to be empirically effective. The support vector based algorithm significantly and consistently surpasses in quality the segmentation produced by a hidden topic Markov model (HTMM). We have demonstrated that in the presence of uncertainty resulting from the use of a speech recognizer, topic segmentation algorithms can be improved by using recognition hypotheses other than that receiving the highest likelihood.

References

Christopher Alberti, Michiel Bacchiani, Ari Bezman, Ciprian Chelba, Anastassia Drofa, Hank Liao, Pedro Moreno, Ted Power, Arnaud Sahuguet, Maria Shugrina, and Olivier Siohan. An audio indexing system for election video material. In *ICASSP*, Taipei, Taiwan, 2009.

Doug Beeferman, Adam Berger, and John Lafferty. Statistical models for text segmentation. *Machine Learning*, 34(1-3):177–210, 1999.

David M. Blei and Pedro J. Moreno. Topic segmentation with an aspect hidden markov model. In *SIGIR*, pages 343–348. ACM Press, 2001.

David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3, 2003.

Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29, 1990.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

Amit Gruber, Michal Rosen-Zvi, and Yair Weiss. Hidden topic markov models. In *AISTATS*, San Juan, Puerto Rico, 2007.

Timothy J. Hazen and Anna Margolis. Discriminative feature weighting using MCE training for topic identification of spoken audio recordings. In *ICASSP*, pages 4965–4968, Las Vegas, Nevada, 2008.

Marti A. Hearst. Texttiling: segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64, 1997.

Xiang Ji and Hongyuan Zha. Domain-independent text segmentation using anisotropic diffusion and dynamic programming. In *SIGIR*, pages 322–329, 2003.

Junbo Kong and David Graff. TDT4 Multilingual Broadcast News Speech Corpus. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2005S11, 2005.

Hideki Kozima. Text segmentation based on similarity between words. In *ACL*, pages 286–288, Morristown, NJ, USA, 1993. ACL.

Ian R. Lane, Tatsuya Kawahara, Tomoko Matsui, and Satoshi Nakamura. Dialogue speech recognition by combining hierarchical topic classification and language model switching. *IEICE - Transactions on Information and Systems*, E88-D(3):446–454, 2005.

Igor Malioutov and Regina Barzilay. Minimum cut model for spoken lecture segmentation. In *COLING/ACL*, pages 25–32, Sydney, Australia, July 2006.

Mehryar Mohri, Pedro Moreno, and Eugene Weinstein. A new quality measure for topic segmentation of text and speech. In *Interspeech*, Brighton, UK, 2009.

Mehryar Mohri. Finite-state transducers in language and speech processing. *Computational Linguistics*, 23(2):269–311, 1997.

Mehryar Mohri. Learning from uncertain data. In *COLT*, pages 656–670, 2003.

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

Jeffrey C. Reynar. Statistical models for topic segmentation. In *ACL*, pages 357–364, College Park, Maryland, 1999.

G. Riccardi, A. Gorin, A. Ljolje, and M. Riley. A spoken language system for automated call routing. In *ICASSP*, pages 1143–1146, Munich, Germany, 1997.

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

Bernhard Schölkopf, John C. Platt, John Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 1999.

Mark Steyvers and Tom Griffiths. Probabilistic topic models. In Thomas K. Landauer, Danielle S. McNamara, Simon Dennis, and Walter Kintsch, editors, *Handbook of Latent Semantic Analysis*, pages 427–448. Routledge, 2007.

David M. J. Tax and Robert P. W. Duin. Support vector domain description. *Pattern Recognition Letters*, 20(11-13):1191–1199, 1999.

Masao Utiyama and Hitoshi Isahara. A statistical model for domain-independent text segmentation. In *ACL*, pages 491–498, 2001.

Vladimir Vapnik. *Statistical Learning Theory*. Wiley, 1998.

Eugene Weinstein. *Search Problems for Speech and Audio Sequences*. PhD dissertation, New York University, September 2009.

J.P. Yamron, I. Carp, L. Gillick, S. Lowe, and P. van Mulbregt. Event tracking and text segmentation via hidden markov models. In *ASRU*, 1998.