A NOVEL DTW-BASED DISTANCE MEASURE FOR SPEAKER SEGMENTATION

Alex Park and James R. Glass

MIT Computer Science and Artificial Intelligence Laboratory
32 Vassar Street, Cambridge, MA 02139, USA
{malex, jrg}@csail.mit.edu

ABSTRACT

We present a novel distance measure for comparing two speech segments that is based on the segmental DTW algorithm. Our approach is based on the idea of finding word-level speech patterns that are repeated by the same speaker. Using this distance measure, we develop a speaker segmentation procedure and apply it to the task of segmenting multi-speaker lectures. We demonstrate that our approach is able to generate segmentations that correlate well to independently generated human segmentations.

1. INTRODUCTION

The problem of speaker segmentation, also known as speaker change detection, has been well examined by researchers in recent years. Although there are many reasons why one might want to segment an audio stream by its constituent speakers, two major reasons stand out. First, for audio documents, speaker changes are often considered natural points around which to structure the document for navigation by listeners. In broadcast news, for example, speaker changes typically coincide with story changes or transitions. Audio recordings of meetings, presentations, and panel discussions are also examples where organizing audio segments by speaker identity can provide useful navigational cues to listeners.

A second motivation for speaker segmentation relates to automatic transcription of speech. In many scenarios, the performance of automatic speech recognition can benefit greatly from speaker adaptation, whether supervised or unsupervised. Speaker segmentation, while not a strict pre-requisite for speaker adapation, is important for performing adaptation on multi-speaker data, as it can provide the recognizer with homogenous speaker data.

Unsupervised approaches to speaker segmentation typically consist of two components: a distance metric for comparing two segments of speech, and a method for determining change points in the audio stream using the distance metric. Most current approaches to speaker change detection use either the Bayesian Information Criterion (BIC) or log likelihood ratios for comparing two speech segments [1, 2]. The main difference between many segmentation approaches is the manner in which the distance measures are used to produce a segmentation. Some take the distance between two halves of a growing window, while others take two halves of a fixed-size analysis window that slides through time. In most cases, a change point is hypothesized when the distance exceeds a certain threshold.

Support for this research was provided in part by the National Science Foundation under grant #IIS-0415865.

2. SEGMENTAL DTW

The focus of this paper is a novel distance metric for comparing two speech segments that is based on a local variant of dynamic time warping (DTW) that we call segmental DTW. Our approach is based on the idea of finding word-level speech patterns that are repeated by the same speaker. A similar approach was applied to the problem of speaker verification in [3]. In previous work, the segmental DTW algorithm was used as the first stage in a clustering algorithm for performing word discovery [4].

Segmental DTW is a variant of traditional dynamic time warping which searches for *multiple* local alignments of two input utterances, $\mathcal{X}$ and $\mathcal{Y}$, and their associated distance matrix D. The algorithm works by dividing the distance matrix into a set of diagonal bands of width R and searching for the best warp path within each band. The diagonal bands serve multiple purposes. First, they constrain the degree of warping so that two sub-utterances are not overly temporally distorted during alignment. Second, they allow for multiple alignments, as each band corresponds to another potential path with different start and end points.

Following path discovery, each path is trimmed by finding the least average subsequence of the path with minimum length L [5]. The minimum length criterion is used to prevent spurious matches between short segments within each utterance. Since each cell $D(i, j)$ corresponds to the distortion between frames i and j, the least average subsequence represents the portion of the aligned path which exhibits good alignment.

At this stage we are left with a family of warp path fragments $\Phi(\mathcal{X}, \mathcal{Y})$, as shown in Figure 1. For performing speaker segmentation, the distance measure with which we propose to compare two utterances $\mathcal{X}$ and $\mathcal{Y}$ is defined as

$$\mathcal{D}_s(\mathcal{X}, \mathcal{Y}) \triangleq \min_{\varphi \in \Phi(\mathcal{X}, \mathcal{Y})} d_\varphi(\mathcal{X}, \mathcal{Y}), \quad (1)$$

where $d_\varphi(\mathcal{X}, \mathcal{Y})$ is the average distortion of the warp path fragment, φ. That is, the distance is the distortion of the minimum distortion alignment path fragment between the two segments.

When two utterances that share both a word and a speaker in common, the alignment path is likely to match the common word and result in a low distortion. On the other hand, if the utterances are spoken by different speakers, the distortion is likely to be much higher, even if the utterances share a word in common, simply because of variation in speaker characteristics in speaking that particular word. Of course, we cannot guarantee that every pair of adjacent utterances with a common speaker will also share a speech pattern on the order of a word. However, by processing blocks of utterances, the likelihood of finding such a repeating pattern is increased.

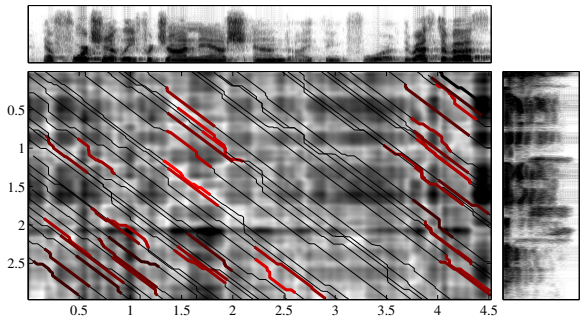

Fig. 1. The segmental DTW procedure for two utterances $\mathcal{X}$ and $\mathcal{Y}$. The family of warp path fragments, $\Phi(\mathcal{X}, \mathcal{Y})$, are shown as colored segments of the constrained diagonal warp paths. Fragments are color coded according to their average distortion, with brighter values corresponding to lower distortion.

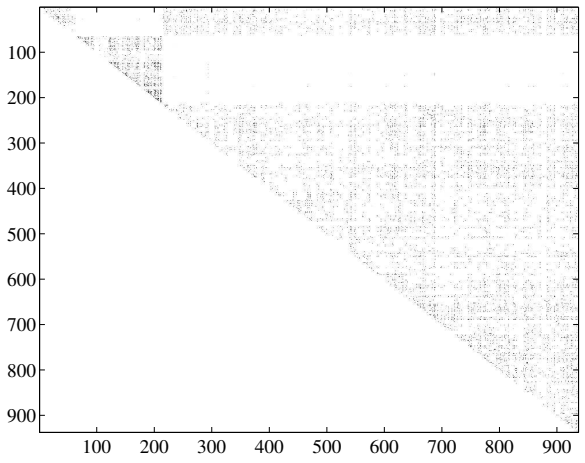

Fig. 2. Utterance level similarity matrix for a physics lecture consisting of two speakers and three segments. The darkness of a cell (i, j) indicates the similarity of utterance i and utterance j using the minimum distortion alignment path fragment.

Though our segmental DTW distance measure is relatively straightforward to describe, it differs from conventional solutions in two important ways. First, speech segments are not considered as "bags of frames", where each frame is processed independently of the other frames in the segment. Instead, the alignment path fragments require frames to be considered in the context of other frames - as part of a sequence, rather than a representative token on its own. The second way in which our approach differs from traditional distance measures, is that utterances are compared on the basis of their most similar token (in this case, a sequence of frames), rather than by averaging all tokens from both utterances.

The effectiveness of this approach can be seen qualitatively by considering the utterance level similarity matrix for a physics lecture as shown in Figure 2. For this matrix, utterance distances are converted into the similarities using a fixed threshold, θ.

$$S(\mathcal{X}, \mathcal{Y}) = \begin{cases} 1 - \frac{D(\mathcal{X},\mathcal{Y})}{\theta} & \text{if } D(\mathcal{X}, \mathcal{Y}) < \theta \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

Even without access to the true speaker change points for the lecture shown in Figure 2, the similarity matrix exhibits a distinct block structure that makes it relatively trivial to visually identify the speaker change points. In the next section we address how to move from this visual representation into one that is more amenable to automatic segmentation.

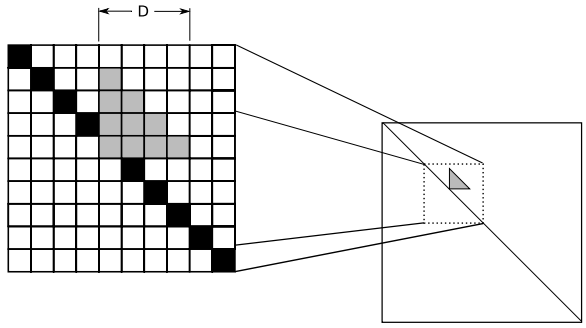

Fig. 3. The diagonal region used to compute the segmentation profile. In this example, $D = 4$.

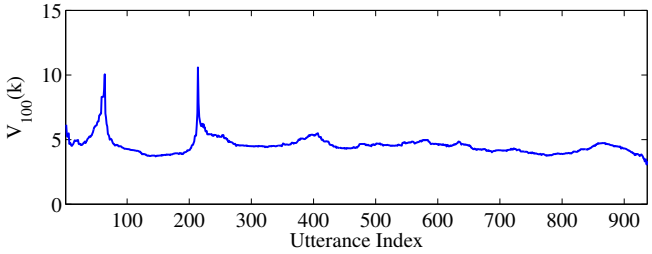

Fig. 4. The log dissimilarity profile for the physics lecture from Figure 2. The width of the diagonal band considered is $D = 100$ utterances.

2.1. Building a segmentation profile

In this section, we propose a method for producing a *segmentation profile* from the similarity matrix shown in the previous section. A segmentation profile is simply a time varying measure of how likely an utterance is to be a speaker change point, i.e. a discontinuity in the similarity matrix, A. Based on this, we propose to track the normalized sum of the cells under a triangular region that slides along the main diagonal of the similarity matrix. This method can be summarized by the diagram in Figure 3. The dissimilarity of segment k to its adjacent segments can be expressed as a function of the normalized sum of nearby cells in the similarity matrix, A.

$$V_D(k) = -\log\Big(\frac{1}{F_D(k)} \sum_{i=k-D+1}^{k} \sum_{j=k}^{i+D-1} A(i,j)\Big) \quad (3)$$

where $F_D(k)$ is a normalizing term that represents the number of cells being added. Figure 4 illustrates a dissimilarity profile for the physics lecture from Figure 2, using a diagonal band, D, of 100 utterances.

As the triangular window slides down the main diagonal of the similarity matrix, the sum of the cells in the window will reach local minima at discontinuities in the matrix. The intuition behind

this approach can be readily seen by considering Figure 2 again. For points in the lecture that are very similar to adjacent regions, more of the cells in the triangular region centered at that particular point tend to be very dark. Conversely, points in the lecture that are dissimilar to adjacent regions have fewer dark points.

Aside from its intuitive appeal, the choice to to consider a small triangular region is motivated by computational reasons as well. With this method, for any particular utterance index, k, it is only necessary to compute similarity scores for pairs of utterances that are within D of k. It follows that the overall running time will be $O(D^2N)$, making the computation linear in the length of the audio stream.

As illustrated in Figure 4, the dissimilarity profile appears to have distinct peaks that coincide with the reference boundaries. Unfortunately, intrinsic variation in the profiles result in many spurious peaks, so simply picking the peaks in the profile would overgenerate the number of potential segment boundaries. We instead utilize an alternative processing technique which focuses on finding *distinct* peaks.

2.2. Finding Distinct Peaks

We note two characteristics of distinct dissimilarity peaks that differ from spurious peaks. First, the distinct peaks happen to persist when filtered with smoothing windows of different widths. Second, they they are significantly higher than neighboring values. The former characteristic can be exploited by performing so-called scale-space filtering on the dissimilarity profile. Scale-space filtering is a technique widely used in pattern recognition and computer vision which generates multiple resolution representations of a signal by filtering with Gaussian windows of different variances, σ. We denote the various smoothed versions of the signal as

$$V_\sigma(k) = G_\sigma(k) * V(k), \tag{4}$$

where $G_\sigma(k)$ is a Gaussian window with variance σ. Progressively larger values of σ reduce the number of peaks in the smoothed signal. We then take the peaks from the smoothed profile and recover the peaks from the original profile by backtracing along the profiles with progressively larger values of σ. This process is shown in Figure 5. At this stage, we are left with peaks that "survive" the scale-space filtering process.

We next turn to the second characteristic of distinct peaks listed above. In order to exploit the characteristic that distinct peaks rise higher above local neighboring values than non-distinct peaks, we can use the smoothed profile with the lowest value of σ as a form of gain control by subtracting it from the original profile. We can then threshold this difference

$$V(p) - V_\sigma(p'), \tag{5}$$

to produce a set of hypothesized segment boundaries. In Eq. 5, p' is a peak found in the smoothed version of the profile, $V_\sigma(k)$, and p is the corresponding peak found by backtracking to the original profile, $V(k)$. Figure 5 illustrates this thresholding procedure. The threshold used in our experiments was set to a fixed value of 0.2 by using a held out lecture as development data. In general, however, the peak picking algorithm is relatively insensitive to the choice of threshold value because of the peakiness of the profile at actual change points.

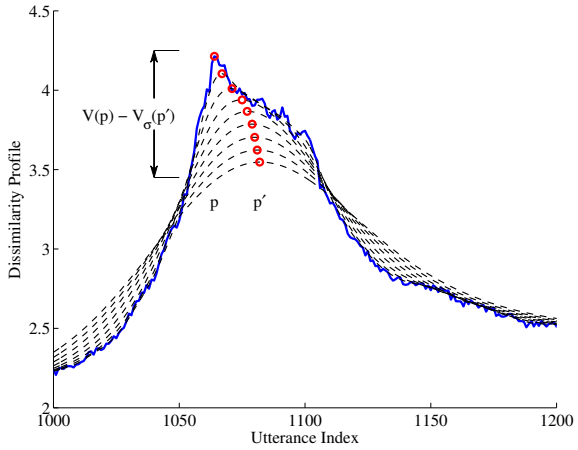

Fig. 5. Illustration of scale space filtering to backtrace peaks in the dissimilarity profile. The solid blue line is the original profile, and the dashed black lines are smoothed versions of the profile for different values of σ. The red circles represent the peak found at the lowest value of σ, backtraced to find the peak on the original profile. The labeled quantity, $V(p) - V_\sigma(p')$, is the value which is thresholded in order to determine how distinctive a peak is in relation to its neighboring values.

3. DATA DESCRIPTION

The data used in our segmentation experiments was a subset of the lectures in the MIT World lecture corpus. Unlike the lectures examined in the previous chapters, we specifically selected lectures consisting of speech from multiple presenters speaking for a significant period of time (several minutes or more). At the time of this writing, these lectures were all publicly available on the MIT World website [6]. The subject material contained within the lectures is wide-ranging; topics include hurricane relief response, medical technology, weapons proliferation policy, and quantum mechanics. In all, the data represents more than ten hours of speech contributed by at least 25 different speakers. A summary of the lectures is given in Table 1.

One of the useful aspects of this data is the availability of high level reference segmentations provided by organizers of the MIT World site. Each lecture is accompanied by a summary page which includes an index of major landmarks in the lecture as judged by a human listener. For the most part, these landmarks typically correspond to speaker changes, but not all speaker changes are included. For instance, minor speaker turns occuring as part of a Q & A session are not individually labeled, but are grouped together as a separate section. Since these landmarks are intended to help guide listeners to important times in the lecture, we focus on the task of automatically finding these landmarks and not just arbitrary speaker changes. In some ways, using these landmarks as our target references is less ambitious than traditional speaker change detection tasks as it does not require exhaustively finding all speaker changes. On the other hand, determining how to select the particular change points that constitute important boundaries to human listeners is also a nontrivial task.

Title	# Speakers	Length
1. Transforming Healthcare	5	1:31
2. Human Machine Relationships	5	1:32
3. Transforming the Next Century	3	1:07
4. Weapons of Mass Confusion	7	2:01
5. Response to Hurricane Katrina	4	1:58
6. Future of Flight	5	1:44

Table 1. A summary of the 6 lectures used in this paper. The number of speakers listed for each lecture is taken from the MIT World web site, and is a lower bound, as most of the discussions also include questions from the audience.

4. RESULTS AND DISCUSSION

The results of our segmentation procedure are summarized in Table 2. Rather than use a floating threshold which yields a detection error tradeoff curve, we instead use a fixed threshold and evaluate the precision and recall of the resulting segmentation. Our reason for doing this is because for actual deployment, an actual segmentation is desirable, meaning the utility of the algorithm is strongly tied to the threshold selected.

For our evaluation, a hypothesized boundary is marked as correct if it falls within seven utterances of a true boundary. In temporal terms, we observed that the average distance between those hypothesized boundaries marked as correct and reference boundaries was 8.5 seconds. Table 2 shows that with the development-set selected threshold, the overall recall rate is 100.0%, meaning that all of the human annotated boundaries are found by our segmentation procedure. The overall precision rate is 80.0%, meaning that of the 35 boundaries proposed, seven were not on the list of human-proposed boundaries. It should be noted, however, that all of these "false alarms" actually do correspond to speaker change boundaries that are simply not annotated in the reference. Therefore, while these proposed boundaries should still be considered errors in the context of our experiment, they may not necessarily detract from the performance of the system when actually deployed. The automatically generated segmentations are displayed together with the reference segmentations in Figure 6. Aside from the few false alarms, the automatic boundaries correlate well with the reference boundaries, indicating that our proposed procedure may prove useful for providing navigational boundaries for this particular task.

Lecture	# Ref Bounds	# Hyp Bounds	Precision (%)	Recall (%)
1.	5	7	71.4	100.0
2.	5	6	83.3	100.0
3.	2	2	100.0	100.0
4.	7	8	87.5	100.0
5.	4	7	57.1	100.0
6.	5	5	100.0	100.0
Overall	28	35	80.0	100.0

Table 2. Segmentation statistics for the processed lectures. # Ref Boundaries is the number of segmentation boundaries provided by the human annotation. # Hyp Boundaries is the number of segmentation boundaries hypothesized by our segmentation algorithm.

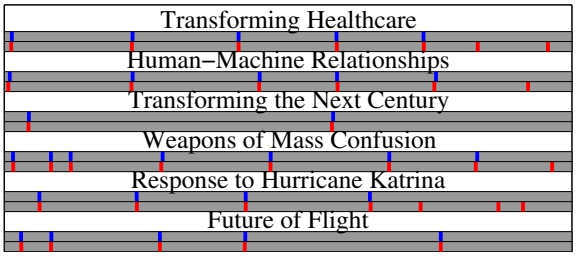

Fig. 6. Comparison of human generated segmentation with automatic segmentation for the 6 processed lectures. For each lecture, the reference boundaries are shown as blue lines in the upper panel, and the automatically generated boundares are shown as red lines in the lower panel.

5. CONCLUSION

The goal of this work has been to illustrate how the speaker specific nature of segmental DTW can be exploited to perform speaker segmentation. To that end, we have implemented and evaluated a segmentation algorithm that is able to find "significant" speaker changes as evaluated by human listeners. Although we recognize that there may be more optimal methods for generating segmentations from a set of inter-utterance distances than the one we describe, our procedure is relatively straightforward and computationally efficient. Moreover, our main interest is not in finding an segmentation algorithm per se, but rather in exploring the potential of segmental DTW as a novel way of comparing utterances. In that context, the results of this chapter are promising, as they demonstrate that segmental DTW can indeed be used as a way to break lengthy audio streams into more manageable segments by their constituent speakers.

6. REFERENCES

[1] S.S. Chen and P.S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proc. DARPA Broadcast News Transcription Workshop*, Lansdowne, VA, February 1998, pp. 127–132.

[2] P. Delacourt and C.J. Wellekens, "DISTBIC: A speaker-based segmentation for audio data indexing," *Speech Communications*, vol. 32, no. 2, pp. 111–126, September 2000.

[3] D. Gillick, S. Stafford, and B. Peskin, "Speaker detection without models," in *Proc. ICASSP*, Philadelphia, 2005, vol. I, pp. 757–760.

[4] A. Park and J. Glass, "Towards unsupervised pattern discovery in speech," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, San Juan, Puerto Rico, 2005.

[5] Y-L. Lin, T. Jiang, and K-M. Chao, "Efficient algorithms for locating the length-constrained heaviest segments with applications to biomolecular sequence analysis," *J. Computer and System Sciences*, vol. 65, no. 3, pp. 570–586, January 2002.

[6] MIT, "MIT World Website: http://mitworld.mit.edu," .