SEGMENTATION AND MODELING IN SEGMENT-BASED RECOGNITION[1]

Jane W. Chang and James R. Glass

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139 USA
http://www.sls.lcs.mit.edu
{jwc, jrg}@sls.lcs.mit.edu

ABSTRACT

Recently, we have developed a probabilistic framework for segment-based speech recognition that represents the speech signal as a network of segments and associated feature vectors [2]. Although in general, each path through the network does not traverse all segments, we argued that each path must account for all feature vectors in the network. We then demonstrated an efficient search algorithm that uses a single additional model to account for segments that are not traversed. In this paper, we present two new extensions to our framework. First, we replace our acoustic segmentation algorithm with "segmentation by recognition," a probabilistic algorithm that can combine multiple contextual constraints towards hypothesizing only the most likely segments. Second, we generalize our framework to "near-miss modeling" and describe a search algorithm that can efficiently use multiple models to enforce contextual constraints across all segments in a network. We report experiments in phonetic recognition on the TIMIT corpus in which we achieve a diphone context-dependent error rate of 26.6% on the NIST core test set over 39 classes. This is a 12.8% reduction in error rate from our best previously reported result.

1. INTRODUCTION

Unlike recognizers that use an acoustic representation based on a temporal sequence of frames, the SUMMIT speech recognizer developed by our group uses a more general representation based on a temporal network of segments, where each segment is associated with a fixed-dimensional feature vector [2]. Segment-based representation enables the extraction of information from the speech signal based on hypothesized segment start and end times. However, before we can exploit this ability in recognition, we must address issues in segmentation, modeling and search.

One requirement of segment-based recognition is to explicitly hypothesize segment start and end times. Since the number of possible segments grows as the square of the number of frames, it is computationally expensive to model and search all segments. In order to reduce computation, we hypothesize a segment network and only model and search the segments in the network. However, deletion and insertion errors in segmentation are irreparable and place an upper bound on recognition performance. In general, computation and performance trade off, and the smaller the number of segments, the larger the number of errors. These tradeoffs have confounded the evaluation of segment-based approaches, as losses in segmentation may be larger than gains in modeling.

We have been using a segmentation algorithm based on local acoustic change [2]. This algorithm hypothesizes a reasonable number of segments with a reasonable number of errors. However, segmentation depends on many factors that are difficult to capture by local acoustic measures alone. For example, although transitions between vowels and consonants may correspond to large acoustic discontinuities, vowel-vowel transitions may not. In this paper, we present "segmentation by recognition," a probabilistic framework that hypothesizes segments through the process of recognition. This framework can combine multiple constraints, such as acoustic context and language models, towards hypothesizing only the most probable segments.

Another requirement of segment-based recognition is to explicitly model and search the entire segment network. A segment network provides alternative segmentations, each of which defines a unique subset of contiguous segments that span the network. Each path through the network traverses only one segmentation and therefore only one subset of segments. However, probabilistically, each path must account for the entire set of feature vectors in the network. Since in general, the number of segmentations grows exponentially as the number of segments, it is computationally expensive to re-process the entire network once for each segmentation.

We have recently developed an efficient search algorithm based on the introduction of a "not" model for segments that are not in a segmentation [2]. For segments at the phonetic level, we also refer to this additional model as the "anti-phone." The "not" model can normalize each path to account for all segments in the network. However, the many segments that are not in a segmentation represent distinct segments of the speech signal that are difficult to capture by a single class. For example, although "not" segments through vocalic regions may share acoustic similarities, they may not resemble consonantal segments. In this paper, we generalize "not" modeling to "near-miss modeling" and use multiple classes to model segments that are not in a segmentation as "near-misses" of segments that are. In addition, we present a search algorithm that can efficiently enforce these contextual constraints across all segments in a network.

In the following sections, we further describe the ideas of "segmentation by recognition" and "near-miss modeling." We also describe a set of experiments in phonetic recognition on the TIMIT corpus and show significant improvements to our previously reported results. Overall, we achieve an error rate of 26.6% on the NIST core test set for 39 classes.

2. SEGMENTATION BY RECOGNITION

In order to extract segment-based features, we must address the problem of segmentation. The goal of segmentation is to hypothesize a small number of segments without introducing a large number of deletion and insertion errors. These goals are conflicting, thus making segmentation a difficult problem. In addition, segmentation is often based on local acoustic measures without the benefit of higher level constraints used in

[1]This research was supported by DARPA under contract N66001-94-C-6040, monitored though Naval Command, Control and Ocean Surveillance Center. J. Chang receives support from Lucent Technologies.

recognition. These issues have contributed to the difficulty in pursuing segment-based approaches.

Towards improving this situation, we have developed "segmentation by recognition," a probabilistic framework that can apply more powerful constraints to the problem of segmentation. In segmentation by recognition, we explicitly create a segment network in the process of running a first pass recognizer with a suitable search. For example, we can use a forward pass Viterbi search to consider all possible segmentations of a set of frames and then use a backwards A^* search to create a network. This idea shares similarities with other work in segmentation and search [1, 4].

Segmentation by recognition has several desirable characteristics. First, it is flexible and can use any first pass recognition strategy. This strategy may be simple or complex depending on available computation. For example, we can still use local acoustic measures, or we can combine context-dependent acoustic and language models.

Second, it is accurate and hypothesizes only the most likely segments. This "minimizes" the number of deletion and insertion errors in a given number of segments. By using a powerful first pass recognizer, we can achieve an upper bound in segmentation performance and allow the exploration of segment-based strategies with less concern over segmentation.

Third, it is adaptive and adjusts to all sources of variability, whether from the segment, word, utterance, speaker or environment. While it tends towards a singular segmentation in regions of confidence, it hypothesizes multiple alternatives in regions of uncertainty. This focuses subsequent segment-based computation where it is most needed.

Finally, since segmentation by recognition runs a first pass recognizer, it hypothesizes not only the most likely segments but also their scores and most likely labels. This information can be used in subsequent segment-based recognition. For example, since our recognition framework is general and allows both frame and segment-based features, we can directly re-use scores from first pass recognition in subsequent segment-based recognition [2]. This allows us to combine complementary recognizers that take advantage of different recognition strategies.

Furthermore, we can explore hierarchical strategies in segment-based recognition. For example, since most confusions in phonetic classification occur between phones of the same manner class, we can focus on reducing these confusions by designing a set of segment-based features specifically to discriminate between the phones of each class [3]. In training, we produce multiple sets of models, one for each set of class-dependent features. However, in testing, we only score each segment against a single set of models based on first pass hypotheses. This enables us to use class-dependent features without sacrificing probabilistic integrity or computational efficiency.

3. MODELING

While segmentation allows the extraction of segment-based features, we must address problems in modeling and search in order to use these features in recognition. To explain the issues, we first review our probabilistic framework [2]. The goal of recognition is to find the sequence of words, W^*, that maximizes the *a posteriori* probability of the speech signal which is represented by acoustic features, A:

$$W^* = \arg\max_W P(W|A) = \arg\max_W P(A|W)P(W)$$

To find W^*, $P(A|W)$ and $P(W)$ are estimated by acoustic and language models, respectively.

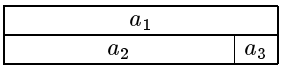

Figure 1: A hypothetical segment network that contains three features, a_1, a_2 and a_3, and two segmentations.

In frame-based recognition, A is a temporal sequence of features. Each segmentation of the speech signal, S, accounts for all frames and therefore all A. As a result, $P(A|W)$ can be efficiently computed.

In contrast, for segment-based recognition, A is a temporal network of features. Each segmentation, S, accounts for only a subset of all segments and therefore only a subset, A_S, of A. In order for a path through S to account for all A, it must also account for $A_{\bar{S}}$, where $A = A_S \cup A_{\bar{S}}$.

For example, Figure 1 shows a hypothetical segment network. For this network, A contains three features, a_1, a_2 and a_3, and two segmentations. A path through the top segmentation, S_{top}, must account for both $A_{S_{top}}$, containing a_1, and $A_{\bar{S}_{top}}$, containing a_2 and a_3.

As a result, for each segmentation, S:

$$P(A|W) = P(A_S A_{\bar{S}}|W)$$

The dependence of $P(A|W)$ on $A_{\bar{S}}$ suggests that the entire segment network must be processed once for each segmentation. However, in general, the number of segmentations grows exponentially as the number of segments, and such processing is computationally daunting.

3.1. "Not" modeling

Recently, we have described an algorithm that efficiently computes $P(A_S A_{\bar{S}}|W)$ for segment-based recognition by using an additional nonlexical "not" model, $\bar{w}$, to account for all segments that are not in a segmentation and therefore $A_{\bar{S}}$ [2]. Assuming independence between A_S and $A_{\bar{S}}$, we can use the not model, $\bar{w}$, to normalize each segmentation, S, to implicitly account for all segments:

$$P(A_S A_{\bar{S}}|W) = P(A_S|W)P(A_{\bar{S}}|\bar{w})\frac{P(A_S|\bar{w})}{P(A_S|\bar{w})} = K\frac{P(A_S|W)}{P(A_S|\bar{w})}$$

where K is constant for all segmentations. For each S, rather than scoring A_S against the lexical models and $A_{\bar{S}}$ against the "not" model, we can score A_S against all models, including both lexical and "not" models, and thereby avoid scoring $A_{\bar{S}}$.

For example, in Figure 1, when scoring a path through S_{top}, rather than scoring a_1 against the lexical models and a_2 and a_3 against the "not" model, we can score a_1 against all models and avoid scoring a_2 and a_3.

3.2. Near-miss modeling

Although "not" modeling is efficient, it requires mapping all segments that are not in a segmentation to a single class. However, the many segments that are not in a segmentation are as distinct as the segments that are in the segmentation, which we map to multiple classes. In fact, each path contextually constrains the entire network, including all segments that are not in its segmentation. For example, in Figure 1, if a_1 is hypothesized to be an [ɔ], a_2 must represent the start of that [ɔ], while a_3 must represent the end of that [ɔ].

Towards applying this contextual constraint in segment-based recognition, we have generalized the idea of "not" modeling to "near-miss modeling" and use multiple nonlexical classes to

model segments that are not in a segmentation as "near-misses" of segments that are. However, since we cannot use multiple classes to normalize each segmentation to account for all segments, we must re-address the search problem.

In order to efficiently compute $P(A_S A_{\bar{S}}|W)$ for near-miss modeling, we have developed a search algorithm that associates each segment with a near-miss subset drawn from all other segments in the network. Specifically, the near-miss subsets are drawn such that for each segmentation, S, the near-miss subsets of the segments in S are mutually exclusive and their union, $\bar{A}_S$, is $A_{\bar{S}}$. Assuming independence between A_S and $\bar{A}_S$, we can then compute for each segmentation, S:

$$P(A_S A_{\bar{S}}|W) = P(A_S|W)P(\bar{A}_S|\bar{W})$$

where $\bar{W}$ are the nonlexical models associated with the segments that are not in S.

One way of showing that there exist such near-miss subsets for any segment network is based on the following observation: all segments in a network that span a given time must not be in the same segmentation. As a result, for each segment, we can choose any time within the segment and add the segment to the near-miss subsets of all segments that span the chosen time. For each segmentation, S, since only one segment in S spans each time, the near-miss subsets of all segments in S must be mutually exclusive. In addition, since the segments in S span all times, the union of their near-miss subsets must be the set of all segments not in S.

For example, we can add each segment to the near-miss subsets of all segments that span its midpoint. By this algorithm, in Figure 1, we map a_1 to the near-miss subset of a_2 and both a_2 and a_3 to the near-miss subset of a_1. Then, a path through S_{top} must account for $A_{S_{top}}$, containing $a1$, and $\bar{A}_{S_{top}}$, containing $a2$ and $a3$.

There are three important ramifications of these observations. First, since the near-miss subsets are mutually exclusive and collectively exhaustive, near-miss modeling maintains the integrity of our probabilistic framework.

Second, since the near-miss subsets are local and time synchronous, near-miss modeling allows the use of efficient search strategies, such as Viterbi, to enforce context across the entire segment network. Effectively, the score for each segment contains not only its score against the lexical models but also the score of the segments in its near-miss subset against the near-miss models. For example, in Figure 1, the score for a_1 includes the score of a_1 against the lexical models and the score of a_2 and a_3 against the near-miss models.

Finally, near-miss modeling is general and allows all segments in the network to be classed in any manner. In addition, the fact that each segment and its near-misses must share temporal and spectral information suggests interesting modeling strategies. For example, by using the midpoint of each segment to map near-miss subsets, we can "maximize" sharing between lexical and nonlexical models. Then, we can introduce one near-miss model for each lexical model. By this algorithm, in Figure 1, if a_1 is hypothesized to be an [ɜ˞], a_2 and a_3 are constrained to be near-misses of an [ɜ˞].

4. EXPERIMENTS

In order to evaluate segmentation by recognition and near-miss modeling, we have conducted phonetic recognition experiments on the TIMIT corpus. For all experiments, we use the NIST 462 speaker training set and 24 speaker core test set and an independent 50 speaker development set. We report error rates on the test set for the 39 phonetic classes commonly used for evaluation [2, 5, 7].

In the following subsections, we describe three recognizers we have used in our experiments. All recognizers initially represent the speech signal using 10 Mel-scale cepstral coefficients (MFCCs) every 10 ms. Each recognizer then further extracts frame or segment-based features. However, regardless of the specific features that are extracted, all features are modeled by mixture of diagonal Gaussian models.

4.1. Context-dependent frames

The first recognizer is frame-based and uses diphone context-dependent acoustic models and a trigram language model. This recognizer is similar to an HMM recognizer that uses one state per phone. All possible segmentations are allowed, and each frame is either a transition between two phones or an internal self-loop of a phone. For features, we extract three averages of MFCCs over varying temporal durations before and after each frame for a total of 60 dimensions per frame. This diphone context-dependent frame-based recognizer achieves a phonetic recognition error rate of 27.7%.

4.2. Context-independent Segments

The second recognizer is segment-based and uses context-independent acoustic models and a bigram language model. For features, we extract average MFCCs over segment thirds, derivative MFCCs over segment boundaries, and log segment duration for a total of 51 dimensions per segment. In addition to the 61 phonetic labels in TIMIT, we model a single "not" class. Using our acoustic segmentation algorithm, this recognizer achieves an error rate of 38.7%. This error rate is higher than the context-independent error rate reported in our previous paper because we are modeling fewer dimensions per segment.

To evaluate segmentation by recognition, we hypothesize segment networks in the process of running the frame-based recognizer described above with a backwards A* search. We have qualitatively compared these networks with the networks hypothesized by our acoustic segmentation algorithm and observed that the probabilistic networks are more adaptive, both within and across utterances. In addition, we have measured time alignment against the manual transcription using a Viterbi algorithm and found the probabilistic networks are significantly more accurate.

We can more quantitatively evaluate our probabilistic segmentation framework by using it in place of our acoustic segmentation for segment-based recognition. Although performance improves as we increase the size of the networks, the improvements become smaller and smaller. For the remaining experiments in this paper, we have used the frame-based recognizer to hypothesize networks that contain approximately half as many segments as hypothesized by the acoustic segmentation algorithm, which is approximately four times the number of segments in the manual transcription. Substituting these networks in the context-independent recognizer described above reduces the error rate to 34.3%.

To evaluate near-miss modeling, we model 61 near-miss classes, one corresponding to each phonetic label. Adding these models to the context-independent recognizer running segmentation by recognition further reduces the error rate to 31.1%.

We have also used the labels from first pass recognition in a hierarchical strategy that uses different features for vowels and consonants derived by simple class-dependent principal components rotations. The hierarchical models obtain an error rate of 30.9%.

Description	Error (%)	Δ (%)
Acoustic Seg + CI Segments + Bigram	38.7	-
+ Segmentation By Recognition	34.3	11.3
+ Near-Miss Modeling	31.1	9.3
+ CD Segments + Trigram	28.4	6.8
+ CD Frames	26.6	6.3

Table 1: Phonetic recognition error rates on the TIMIT core test set over 39 classes.

4.3. Context-dependent Segments

The third recognizer is a segment-based recognizer that uses segmentation by recognition, diphone context-dependent acoustic models and a trigram language model. Based on experiments using the development set, we have chosen to model right context. For each segment, we extract average MFCCs over thirds and log duration. In addition, for each right context, we extract three averages over the same temporal durations as used in the frame-based recognizer. This yields 61 dimensions per segment.

When using a single "not" model, this diphone context-dependent segment-based recognizer achieves an error rate of 29.0%. When using the 61 "near-miss" classes described above, the error rate reduces to 28.4%.

Finally, we have directly re-used the scores from first pass recognition in subsequent segment-based recognition. The combined diphone context-dependent frame and segment-based recognizer achieves a phonetic recognition error rate of 26.6%.

5. DISCUSSION

This paper describes two novel extensions to segment-based recognition which yield significant improvements in phonetic recognition. Table 1 summarizes our experiments. The descriptions are cumulative and show how each extension further improves performance. The overall result of 26.6% is a 31.3% improvement over the baseline result of 38.7%. This is a 12.8% improvement over our best previously reported result of 30.5% and is competitive with the best results reported using other approaches, including neural networks and the more common HMMs [5, 7]. In the near future, we plan to run similar experiments in word recognition.

We believe segmentation by recognition will facilitate working with segment-based approaches. In comparison to the acoustic segmentation algorithm we have been using, segmentation by recognition can improve recognition performance by 11.3% while cutting the segment search space in half. Although the frame-based recognizer used to achieve this performance is computationally expensive, we believe we can significantly reduce first pass computation without overly sacrificing subsequent performance. For example, we have experimented with a first pass landmark-based recognizer that represents the speech signal as a temporal sequence of acoustically informative landmarks. In addition to considering fewer frames, we can model fewer broad classes to further reduce computation.

We believe near-miss modeling is an important extension of the probabilistic framework for segment-based recognition that we have been developing. By taking advantage of the ability to draw suitable near-miss subsets, we can 1) use multiple classes to model all segments in the network, 2) use efficient search algorithms to constrain context across the entire network, and 3) maintain our probabilistic framework.

In comparison to the "not" model we have been using, near-miss modeling can improve performance by 9.3% in context-independent recognition and 2.1% in context-dependent recognition. As the near-miss classes used to achieve this performance are rather simple, we believe we can gain further improvements by exploring strategies for near-miss modeling. For example, we have experimented with the use of context-dependent near-miss models. In addition, we can class based on other measures such as relative duration.

One of the main motivations of our work is the belief that segment-based approaches offer advantages over the more common frame-based approaches. In exploring the relative strengths and weaknesses of these different strategies, our work has taken advantage of a probabilistic framework that allows both frame and segment-based representations.

We have been able to build a diphone context-dependent frame-based recognizer that achieves a phonetic recognition error rate of 27.7%. We believe the use of features that span varying durations has contributed to this result. While the short averages capture transitional information, the long averages capture steady state information. A similar HMM topology that uses multiple states per phone should only do better.

We have also been able to build a diphone context-dependent segment-based recognizer that achieves an error rate of 28.4%. Although this result is not as good as the frame-based result, the segment-based recognizer uses much simpler models trained on approximately half the number of feature vectors and searches over a much smaller space confined to the segment networks. In addition, even with the high performance frame-based recognizer, the addition of the segment-based recognizer reduces error rate by 4.0%. This suggests that segment-based features can capture additional information in the speech signal that is relevant to recognition.

Based on these results, we believe that better feature extraction and modeling can improve performance for both frame and segment-based recognition. In this paper, we have briefly discussed one way to improve feature extraction and modeling using a hierarchical strategy. We plan to pursue this direction in future research [3]. In addition, there are other techniques for segment modeling that may be able to better capture correlations across a segment [6].

Finally, we plan to continue exploring the similarities and differences between frame and segment-based approaches. On the one hand, we believe we can design more complementary recognizers that can reduce computation without sacrificing overall performance. On the other hand, we have seen similarities between the strategies that we may be able to combine into a better speech recognition framework.

6. REFERENCES

[1] J. Cohen. Segmenting speech using dynamic programming. *Journ. ASA*, 69(5):1430–1438, May 1981.

[2] J. Glass, J. Chang, and M. McCandless. A probabilistic framework for feature-based speech recognition. In *Proc. ICSLP*, pages 2277–2280, 1996.

[3] A. Halberstadt and J. Glass. Heterogeneous acoustic measurements for phonetic classification. In *these proceedings*.

[4] L. Hetherington, M. Phillips, J. Glass, and V. Zue. A* word network search for continuous speech recognition. In *Proc. Eurospeech*, pages 1533–1536, 1993.

[5] L. Lamel and J. Gauvain. High performance speaker-independent phone recognition using cdhmm. In *Proc. Eurospeech*, pages 121–124, 1993.

[6] M. Ostendorf and S. Roukos. A stochastic segment model for phoneme-based continuous speech recognition. *Trans. ASSP*, 37(12):1857–1869, Dec. 1989.

[7] A. Robinson. An application of recurrent neural nets to phone probability estimation. *Trans. Neural Networks*, 5(2):298–305, Mar. 1994.