SVM Modeling of "SNERF-Grams" for Speaker Recognition

Elizabeth Shriberg[1,2], Luciana Ferrer[1], Anand Venkataraman[1], Sachin Kajarekar[1]

SRI International[1]
International Computer Science Institute[2]
{ees,lferrer,anand,sachin}@speech.sri.com

Abstract

We describe a new approach to modeling idiosyncratic prosodic behavior for automatic speaker recognition. The approach computes prosodic features by syllable (syllable-based nonuniform extraction region features, or "SNERFs"), and models the syllable-feature sequences ("SNERF-grams") using support vector machines (SVMs). We evaluate performance on development data for a system submitted to the NIST 2004 Speaker Recognition Evaluation. Results show that SNERF-grams provide significant performance gains when combined with a state-of-the-art baseline system, as well as with both prosodic and word-based noncepstral systems.

1. Introduction

Recent research in modeling long-range stylistic features for speaker recognition has shown remarkable success [1,2,3,4,5]. So much so, that it has become increasingly challenging to come up with approaches that improve on the status quo—particularly when training data for target speakers is limited. We describe a new approach to modeling idiosyncratic stylistic prosodic behaviors for automatic speaker recognition. The approach computes prosodic features associated with each syllable (syllable-based nonuniform extraction region features, or "SNERFs"), and then models those syllable-feature sequences ("SNERF-grams") using support vector machines (SVMs).

We evaluate the approach on development data for a system submitted to the NIST 2004 Speaker Recognition Evaluation. The task is a speaker verification task, in which one side of a short telephone conversation is provided for training the speaker model, and another conversation side is provided for testing.

2. Method

2.1. Speech Data

We used 2564 5-minute conversation sides from the Fisher corpus of two-party telephone conversations on various topics. We divided the set into three subsets containing no shared speakers, as shown in Table 1.

As indicated in Table 1, we used only half the original conversation side length in our development sets; this is to better match the different data set in the NIST 2004 evaluation, which contains shorter test lengths.

2.2. Baseline System

Our baseline cepstral Gaussian mixture model (GMM) system [6] uses a 300-3300 Hz bandwidth front end consisting of 19 MEL filters. It computes 13 cepstral coefficients (C1-C13) with cepstral mean subtraction, and their delta and double delta coefficients, producing a 39-dimensional feature vector. The feature vectors are modeled by a 2048-component GMM. The background GMM is trained using gender- and handset-(electret, carbon and cell phone) balanced data. Target GMMs are adapted from the background GMM using MAP adaptation of the means of the Gaussian components. For channel normalization, the feature transformation described in [7] is applied using gender- and handset-dependent models that are adapted from the background model. Verification is performed using 5-best Gaussian components per frame, selected with respect to the background model scores.

2.3. SNERF-Gram System

2.3.1. Automatic speech recognition

Conversation sides were segmented and decoded using SRI's five times real-time conversational speech recognizer, which has a word error rate of approximately 20% on the 2003 Fisher evaluation test set. Acoustic models were trained on data from the Switchboard and CallHome corpora. The language model was trained on additional data from broadcast news transcripts and online Web data. Note that no part of the recognition system was trained or tuned on the Fisher data itself.

	Background Model	Dev set 1	Dev set 2
Conversation sides	1128	734	702
Unique speakers	1128	249	249
Imposter trials	-	13130	9153
True speaker trials	-	1508	1328
Ave. orig. side length (min.)	~5	~5	~5
Ave. side length used (min.)	~5	~2.5	~2.5

Table 1: Statistics on Fisher data sets

*2.3.2. SNERF-*gram *features*

To obtain estimated syllable regions, we syllabified the output of the speech recognizer using 'tsylb2', a program that uses a set of human-created rules that operate on the best-matched dictionary pronunciation for each word. For each resulting syllable region, we obtain phone-level alignment information from the speech recognizer, and then extract a large number of features related to the duration, pitch, and energy in the syllable. The duration features are obtained from recognizer alignments. Pitch is estimated using the *get_f0* function in ESPS/Waves, and then post-processed using an approach adapted from [8]. The post-processing median-filters the pitch, then fits linear splines, and produces the posterior probability of pitch halving and pitch doubling for each frame

using a log-normal tied-mixture model of pitch. The model also estimates speaker pitch range parameters used for normalization. Energy features are obtained using the RMS energy values from ESPS/Waves, and post-processed to fit one spline for each segment obtained from the pitch stylization.

We then create a number of duration, pitch, and energy features, as follows. The motivation for computing features that are highly correlated (differ for example only in normalization, binning, or N-gram length, as described below) is that we will be able to view the usefulness of a specific feature in the SVM modeling. Since the SVM model does not suffer (too greatly) by having a large set of features, this allows us to inspect the feature weights and discover which normalizations, binnings, and so on, are best for which types of features.

For duration features, we use five different regions in the syllable: onset, nucleus, coda, onset+nucleus, nucleus+coda, and the full syllable. We obtain the duration for that region, and normalize it using three different ways of computing normalization statistics based on background model data. We use instances of the same phone sequence anywhere, instances of the same phone sequence in the same position, and instances of the same triphone anywhere. We use four different types of normalization: no normalization, division by the distribution mean, Z-score normalization ((value-mean)/st.dev), and percentile.

For pitch features, we use two different regions: voiced frames in the syllable, and voiced frames ignoring any frames deemed to be halved or doubled by the pitch post-processing described earlier. The pitch output in these regions is then used in one of three forms: raw, median-filtered, or stylized using the linear spline approach mentioned earlier. For each of these pitch value sequences, we compute a large set of features: maximum pitch, mean pitch, minimum pitch, maximum – minimum pitch, number of frames that are rising/falling/doubled/halved/voiced, length of the first/last slope, number of changes from fall to rise, value of first/last/average slope, and the maximum positive/negative slope. These features are normalized by five different approaches: no normalization, divide by mean, subtract mean, Z-score normalization, and percentile value.

For energy features, we use four different regions: the nucleus, the nucleus minus any unvoiced frames, the whole syllable, and the whole syllable minus any unvoiced frames. These raw energy values are then used to compute features in a manner similar to that just described for pitch features.

Because we use count-based features in the SVM modeling, we discretize the features described above. Since we do not know *a priori* where to place thresholds for binning the data, we try a small number of different total bin counts. In each case, we discretize evenly on the rank distribution of values for the particular feature, so that resulting bins contain roughly equal amounts of data. We use four different bin counts per feature: 2, 3, 5, and 10 bins.

Each resulting feature is then also modeled in three ways: unigram (current syllable only), bigram (current syllable and previous syllable or pause), and trigram (current syllable and previous two syllables or pauses). Pauses present an interesting case in this approach. Although they do not contain pitch or energy information, we do not want to ignore them. They provide useful conditioning information when present in the longer N-grams, and provide the priors for pause occurrence when used as unigrams. We thus needed to come up with a binning approach for pauses. We used one set of hand-chosen threshold values (6, 15, and 30 frames) to divide pauses into four different lengths. This approach was used across all features (note that each feature may combine with a pause for N-gram lengths larger than 1).

The resulting number of different observed N-grams (where an N-gram is a sequence of specific bin values of a specific feature) is large—on the order of 200,000. For each N-gram, we count the number of appearances of that N-gram and normalize that count by the total syllables in the conversation side. The resulting values are provided to the SVM.

2.3.3. SVM modeling

An SVM [9] is used to separate true and imposter speakers. Each training or test conversation side is assumed to provide a single point in the hyperspace, whose coordinates are given by the feature vector described in the previous section. For practical reasons, we do not use the complete set of features. Instead, we select the 10,000 most frequent N-grams occurring in the background model training data.

During training, each true speaker vector is assigned to the class "+1", and each imposter is assigned to the class "-1". The score assigned to any particular test trial is then calculated as the Euclidean distance from the separating hyperplane to the point that represents the particular trial, with negative values indicating imposters. We used the SVMLite toolkit [10] by Thorsten Joachims to induce SVMs and classify instances. We used a linear kernel and imposed a bias of 500 against misclassification of positive examples, a number that we obtained empirically through experimentation. The scores obtained in this manner were then normalized using TNORM [11].

An important advantage of this approach (over, for example, GMMs) is that if one uses a linear kernel, the induced SVM provides a way to infer the contribution of individual features to overall performance. By examining the hyperplane, one can determine the importance of a feature by looking at the angle it forms with the hyperplane. The more discriminative the feature, the more orthogonal the angle with the hyperplane. Intuitively we may say that the rate at which one approaches the hyperplane and crosses over to the other side by moving along the axis of any particular feature is directly representative of the importance of that feature in classifying the sample. Note that for this to be true, features must be normalized so that they have comparable standard deviations. Accordingly, we use the cosine of the angle between the normal to the hyperplane and the axis of interest as a measure of the importance of each feature in the classification task.

In future work we plan to use this measure to aid feature selection, rather than basing selection only on feature frequency as we do now. We also expect that information on feature usage will lead to the discovery of better features. Perhaps most important: the study and interpretation of such feature weights should lead to progress in our fundamental scientific understanding of speaking behavior.

2.4. Other Noncepstral Systems

The true test of the utility of the SNERF system is to see whether it provides complementary information beyond that already modeled by a number of successful systems developed

in past work. We have previously developed three types of systems, aimed at capturing long-range features for speaker verification. All systems use the same speech recognition output as described earlier for the SNERF-gram system.

2.4.1. Word N-grams SVM system

The word N-gram based SVM system consists of a linear-kernel SVM where the coordinates of the point are determined by the relative frequencies of word N-grams in the conversation sides. All orders of N-grams from 1 to 3 are chosen as potential candidates for input space dimensions. The particular N-grams to model are chosen based on their frequency: all sequences that appear more than once in the background model set are included in the feature vector. This way, around 150,000 N-grams are included, but in this case this size is not problematic because only a few of them occur on each conversation side. As for the SNERF-gram system, we used a linear kernel with a bias of 500 against misclassification of positive examples. These scores are also TNORMed.

2.4.2. Duration systems

This system [2] models a speaker's idiosyncratic temporal patterns in the pronunciation of individual words, phones, and states, inspired by previous work on similar features for conversational speech recognition. Three different models are created: (1) word models that contain the sequence of phone durations in the word; (2) phone models that contain the duration of context-independent phones; and (3) state-in-phone models that contain the sequence of HMM state durations in the phones. Speaker models obtained through MAP adaptation of means and weights of a background model are used to score test conversations. This score is normalized by the score obtained using the background model on the same test sample. The score is further normalized using TNORM.

2.4.3. GMM-based NERF system

This system models another class of nonuniform extraction region features (NERFs). In this case, the region is not a syllable but rather stretches from one pause to the next pause (using a pause duration threshold of 500 ms). One feature vector comprising various F0, energy and duration features is extracted per region. Features are modeled using GMMs after whitening. Scores are then obtained as for the duration system. The details of this approach are explained in [4].

2.5. System Combination

All systems are combined using LNKNet software. A neural network with no hidden layer and sigmoid output nonlinearity is used as a classifier. The combiner is trained on dev set 2 and applied to dev set 1 to obtain the final scores.

3. Results and Discussion

We report results using two error metrics: equal error rate (EER) and minimum detection cost function (min DCF, used by NIST). The equal error rate is the probability of miss detections (MD) when this value is equal to the probability of false alarms (FA). DCF is computed using four quantities: 1) MD probability, 2) FA probability, 3) cost for misclassification (C_{MD} and C_{FA}), and 4) target prior (P_{tgt}) with the following formula:

$$DCF = C_{MD}\, P_{tgt}\, P_{MD} + C_{FA}\, (1\text{-}P_{tgt})\, P_{FA}$$

In the results presented here, parameters are set to: C_{MD}=10, C_{FA}=1 and P_{tgt}=0.01.

3.1. Two-way Combinations with Baseline

To provide an idea of how much complementary information the SNERF system provides beyond that from the baseline system, relative to the other noncepstral systems developed in prior work, we ran all two-way combinations of noncepstral systems with the baseline. Results are shown in Table 2. In all results shown in this paper, the three duration systems are combined into one system (which, as would be expected, performs better than any single duration system alone).

	EER (%)	Min DCF ($x10^2$)
Baseline only	8.95	2.83
Baseline + 3 Duration systems	7.03	2.13
Baseline + SNERF-grams	7.36	2.31
Baseline + Word N-grams	7.82	2.21
Baseline + P2P NERFs	8.02	2.51

Table 2: Equal error rate and detection cost function for all two-way combinations of baseline system with noncepstral systems

All noncepstral systems improve performance over the baseline (all are statistically significant). The best system in combination is the three-part duration system developed in earlier work. The SNERF-gram system is next best in terms of EER, but only third best in DCF. Since the SNERF-gram system includes duration information, the next obvious question is whether or not it will provide additional (complementary) information when combined with the duration system.

3.2. Three-way Combinations with Baseline and SNERFs

To investigate the question of information added beyond that in the duration system, we ran a three-way combination with the baseline, the three-part duration system, and the SNERF-gram system. We also ran the combination for the word and NERF systems, respectively. As shown in Table 3 (cf. Table 2), the SNERF-gram system combines well with other systems. In particular, it provides a gain significant at the .01 level when combined with the duration system.

Baseline + SNERF-grams +	EER (%)	Min DCF ($x10^2$)
Duration	6.50	1.89
Word N-grams	7.16	1.90
P2P NERF	7.49	2.29

Table 3: Equal error rate and detection cost function for three-way combinations

3.3. Multisystem Combination

To assess the contribution of the SNERFs system when all

other systems are present, we ran two further combinations, in which all noncepstral systems were included with and without SNERFs. Results are shown in Table 4. As indicated, the SNERF-gram system continues to provide a highly significant gain (again at the .01 level) even when all other systems are present. Detection error tradeoff curves are shown for these systems in Figure 1.

	EER (%)	Min DCF ($x10^2$)
Baseline	8.95	2.83
All systems except SNERFs	7.29	1.96
All systems including SNERFs	6.43	1.68

Table 4: Equal error rate and detection cost function for multiway system combinations

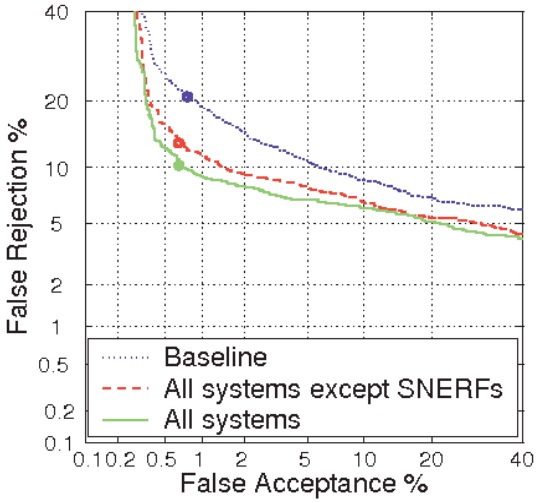

Figure 1: Detection error tradeoff curves for baseline and multiway combinations

3.4. Feature Usage

Using the "cosine of theta" measure described earlier, we can compute the usage of each feature in the model. Note however that the measure is also affected by the manner in which we choose the features to include in the SVM. Since currently we base this decision on frequency, features that occur rarely may be excluded from the model (even if discriminative), because they suffer from high inherent variance. Nevertheless, the measure provides a way to rank the importance of the many types of frequently occurring features retained in the SVM. We find that longer N-grams were generally more useful than shorter N-grams, and that the SNERF-gram system appears to make ample use of pitch and energy information (not just duration). This is consistent with the finding noted earlier that SNERFs can combine well with duration systems.

4. Conclusions

SVM modeling of SNERF-grams appears to be a valuable knowledge source for speaker recognition. Even if training data is limited to a few minutes of speech, SNERF-grams provide significant performance gains when combined with a variety of other systems. Because results reported here have not yet been optimized with respect to feature selection (a process that could be informed by feature weights in the SVM) there is room for further improvement. Taken together, these findings suggest that—despite the moniker—SNERF-grams are nothing to sneeze at.

5. Acknowledgments

We thank SRI colleagues Andreas Stolcke, Jing Zheng, and Kemal Sönmez for speech recognition output, development of the other systems used, and advice and discussion. This work was supported by interagency KDD funding through NSF Award IRI-9619921. The views herein are those of the authors and do not reflect the views of the funding agencies.

6. References

[1] G. Doddington, et al., "Speaker Recognition Based on Idiolectal Differences between Speakers", in *Proc. Eurospeech*, Aalborg, Denmark, Sept. 2001, pp. 2521-2524.

[2] L. Ferrer et al., "Modeling Duration Patterns for Speaker Recognition," in *Proc. Eurospeech*, Geneva, pp.2017-2020, September 2003.

[3] S. Kajarekar et al., "Speaker Recognition Using Prosodic and Lexical Features", in *Proc. IEEE Speech Recognition and Understanding Workshop*, St. Thomas, U. S. Virgin Islands, pp. 19-24, December 2003.

[4] S. Kajarekar, et al., "Modeling NERFs for Speaker Recognition", in *Proc. Odyssey Workshop*, Toledo, Spain, pp. 51-56, May 2004.

[5] D. Reynolds et al., "The SuperSID Project: Exploiting High-level Information for High-accuracy Speaker Recognition", in *Proc. ICASSP*, vol. 4, Hong Kong, pp. 784-787, April 2003.

[6] D. Reynolds, "Speaker Identification and Verification Using Gaussian Mixture Speaker Models", *Speech Communication*, vol. 17, no. 1-2, pp. 91-108, Aug. 1995.

[7] D. Reynolds, "Channel Robust Speaker Verification via Channel Mapping", in *Proc. ICASSP*, vol. 2, pp. 53-56, April 2003.

[8] K. Sönmez, et al., "Modeling Dynamic Prosodic Variation for Speaker Verification," in *Proc. ICSLP*, vol.6, Sydney, Australia, pp. 2631-2634, August 1998.

[9] V. Vapnik, "The Nature of Statistical Learning Theory", Springer, 1995.

[10] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features", in *Proc. of European Conference on Machine Learning*, 1998.

[11] R. Auckenthaler, et al., "Score Normalization for Text-Independent Speaker Verification Systems", *Digital Signal Processing 10*, 2000, pp. 42–54.