

A Text-Constrained Prosodic System for Speaker Verification

Elizabeth Shriberg¹, Luciana Ferrer²

¹SRI International, Menlo Park, CA, USA

²Stanford University, Stanford, CA, USA

ees@speech.sri.com, lferrer@speech.sri.com

Abstract

We describe four improvements to a prosody SVM system, including a new method based on text- and part-of-speech-constrained prosodic features. The improved system shows remarkably good performance on NIST SRE06 data, reducing the error rate of an MLLR system by as much as 23% after combination. In addition, an N -best system analysis using eight systems reveals that the prosody SVM is the third and second most important system for 1- and 8-side training conditions, respectively—providing more complementary information than other state-of-the-art cepstral systems. We conclude that as cepstral systems continue to improve, it should become only more important to develop systems based on higher-level features.

Index Terms: speaker recognition, prosody, part of speech (POS), text-constrained, kernel, intersession variability, SVM

1. Introduction

Higher-level features that use linguistic or longer-range information have garnered significant interest in speaker recognition as a means by which to augment systems based on frame-level cepstral features (see [1] for a review). This is particularly true for contexts in which one has available at least a few minutes of speech for training and test data, since higher-level features operate at longer time spans and are consequently more sparse. Higher-level features also offer the possibility of robustness to channel variation, and can help provide metadata about a recording beyond speaker identity.

This paper presents a prosodic support vector machine (SVM) system that performs surprisingly well—both alone and in combination with multiple state-of-the-art cepstral and higher-level systems. The system is the result of four improvements over an earlier system described in [2]. Three improvements come from recently developed modeling techniques, including feature parameterization [3], an SVM smoothing kernel [4], and intersession variability compensation [5]. The fourth improvement, which has not previously been published, comes from combining the system with a new system that shares the same prosodic features as used in the unconstrained system, but that conditions features based on text and part-of-speech (POS) constraints.

Section 2 describes the system and improvements, with most attention to the new approach using text and POS constraints. Section 3 provides results on NIST SRE06 evaluation data for (1) the prosodic system after the four improvements, (2) the contribution of the improved system to performance of our best single system, and (3) the importance of the prosody system to combination when multiple systems are available. Section 4 summarizes findings and concludes with a general implication for future work.

2. Method

2.1. Baseline prosodic system

In the original prosodic system, as well as in the improved versions herein, syllables are inferred from automatic speech recognition (ASR) output via automatic syllabification. ASR output consists of 1-best word hypotheses and phone-level alignment information produced by SRI's speech recognition system. None of the test or background data were used in tuning of the recognition system. The system is a fast, two-stage version of SRI's conversational telephone speech (CTS) system, as originally developed for the 2003 DARPA Rich Transcription evaluation [6] and later modified for the NIST 2004 speaker recognition evaluation [7].

A set of prosodic features is extracted from each syllable. Prosodic features include features based on pitch, energy, and duration information. Duration features use phone-level alignment information from ASR, and reflect the duration of either the whole syllable, or of subsyllable regions (onset, rhyme, or coda). To make duration features text independent, we normalize them based on phone identity using mean and variance statistics from the background data. Pitch and energy features use information from a piecewise linear fitting to obtain features such as linear pitch slopes, and to help eliminate octave errors. Examples of pitch features include the mean, minimum, maximum, and slope within a syllable. Energy features use frame-level RMS energy values and comprise measures similar to that for pitch features.

We model these features using support vector machines (SVMs), which has proven to be an exceptionally good choice in speaker verification systems. In order to model the variable-length sequences of prosodic features with an SVM, we first need to define a kernel that can measure the *similarity* between two sequences, or analogously, a transformation that can map the sequences into fixed-length vectors. Our approach to date has been to use as transformation some set of values that represent the empirical distribution of the prosodic features observed in the sample, assuming independence of the features within one syllable; transforms are obtained independently for each feature and later concatenated into the final vector. In practice, independence is also assumed across time in order to find the parameters that describe the empirical distribution of each feature. Nevertheless, we make up for this assumption by also modeling the distribution of sequences of 2, 3 and 4 consecutive syllables (or pauses). We call these sequences N-grams.

In the original system the domain of each feature was divided into bins to produce roughly equal amounts of data in each bin. The distribution of each feature was obtained from a large set of held-out speakers. N-grams were modeled by concatenating the symbols for 2, 3 and 4 consecutive bins. Since this approach resulted in too many symbols, the list of symbols

was pruned based on the frequency of the symbols in the held-out data. The transformation for each sample then consisted of the frequencies of each of the symbols in the pruned list as found in the sample. The resulting features, which we called *hard-bin* features, were finally normalized to a uniform distribution between 0 and 1. We then trained an SVM to perform regression on the labels (-1 and +1) with a bias of 500 against misclassification of positive examples. Further details are given in [2].

2.2. Modeling improvements

Soft-bin transform. In [3] we introduced a new method to transform the prosodic features to obtain a fixed-length vector that can be well modeled by SVMs. As in the previous method, the idea is to find a set of values that describes the empirical distribution observed in the sequence, but in this case the parameters are the weights of a set of Gaussian mixture models (GMMs). For each feature and each of its N-grams, a GMM is trained on held-out data using a VQ or EM algorithm. Given a sample, the transform is found as the vector of posterior probabilities (or weights) for each of the Gaussians in the original GMM. We call the resulting transformed features *soft-bin* features. In this paper we will show results only for the case of VQ training of the GMMs. As in the approach using hard bins, features are finally normalized to a uniform distribution.

Smoothing kernel. A further improvement is achieved by considering that parameters (Gaussian weights) from neighboring Gaussians are likely to have similar importance for classification. In [4] we showed that imposing this constraint through the addition of a new regularization term in the SVM objective function is equivalent to using a new *smoothing* kernel on the features.

Intersession variability compensation. Several methods have been developed to compensate for within-speaker variability attributable to different recording instances, or “sessions”. Recently, nuisance attribute projection (NAP) has been proposed as a method to compensate for intersession variability when using SVMs as speaker models [5]. A matrix $P = I - UU^T$ is created that projects features onto a subspace that is hoped to be more resistant to intersession variability. Although developed for cepstral-based systems, we applied the same technique to our prosodic system. We use principal component analysis of the within-class covariance matrix to determine the intersession variability subspace [8]. The columns of matrix U then correspond to the first N principal components, where N is tuned on a development set.

2.3. Text- and POS-constrained features

We introduce a new system where prosodic features remain the same as in the system described above, but constraints are used to filter the conversation side to retain contexts hypothesized to have greater across-speaker than within-speaker prosodic variability than average speech locations. Each context is defined by a list of one or more constraints, where a constraint consists of a specific word or word sequence, a specific POS tag (or sequence of tags), or a word+POS tag pair (or sequence). Individual constraints within a context match in number of syllables. Note that unlike the case of text-constrained cepstral models, the contexts used here contain multiple words varying in phonetic content, but hypothesized to have similar prosody. POS information is obtained by a simple HMM-based tagger trained on Penn Treebank-3 data, and implemented using SRI Language Model (SRILM) tools. Tags were post-processed us-

ing hand-constructed word lists to distinguish important classes confounded by the tagger, in particular to separate interjections (tag = UH) into backchannels, filled pauses, and discourse markers. Contexts were suggested by a psycholinguist but are obviously unlimited and an open area for future work. The system makes use of 18 contexts that performed well when a system using only those contexts was run on development data.

We find that some of the most useful constraints are based on parts of speech that reflect discourse categories rather than grammatical categories, including backchannels, discourse markers, and disfluencies. Contexts include subsets of other contexts, as it turned out to be useful to include both contexts. The subset is likely to be less variable in prosody; the larger set includes more cases and thus can increase robustness. For example, fillers and discourse markers share some prosodic properties, and we included both fillers alone and the combined class. Of syntactic-based constraints, personal pronouns were particularly useful, both alone and when modeled as a sequence with the following syllable. It is not yet clear why this class is useful, whereas other individual classes (e.g., nouns, verbs, and adjectives) were less so. Some possibilities are that personal pronouns reflect more speaker-specific prosodic properties because of location (near phrase onsets) or semantics (involves content from the first person perspective), but this requires further study.

Pause conditioning was found to be clearly important, particularly for words and POS classes that occur frequently in nonpause contexts. Pauses can reflect either syntactic/semantic boundaries, disfluencies, or turn boundaries (locations during which the other speaker is talking). In all cases (albeit for different reasons), the pause context is likely to help because speech is more isolated from the local prosodic context, which itself must contribute to variability.

These features are transformed and modeled using the same techniques used for the unconstrained version of the system. In this case, transformations are found for the complete sequence matching a certain context (in the same way in which N-gram transforms are found in the case of unconstrained features).

2.4. Task and data

Results are reported on the NIST 2006 SRE evaluation data [9]. Results are for the primary subset (the “Common Condition”), which consists of English-only conversations. Test data consist of one conversation side. We report results for both 1- and 8-side training. The 1-side training set comprises 3,209 conversation sides, 517 models, and 24,013 trials; 8-side training contains 6,556 sides, 483 models, and 17,547 trials. Background training data consist of 1,553 conversation sides from separate data collections (Switchboard-II, Fisher, and some SRE2004). Background data did not share any speakers with the data in the test sets. The data used to find the bins in the hard-bin system and the GMMs in the soft-bin systems, as well as data used to compute the intersession variability matrix U , was drawn from 2004 evaluation data.

3. Results and Discussion

3.1. Prosody model and combination with MLLR

Table 1 provides results for the unconditioned, conditioned, and combined systems using prosodic feature sequences; the result after applying intersession variability compensation to the combined system is also given. As shown, the new feature modeling techniques give significant improvements, especially for

the 1-side condition. The conditioned system performs roughly on par with the unconditioned system (as data size increases, the conditioned system is actually better, despite having very sparse features per context), and there is a significant reduction in error rates when systems are combined. The combination is performed by concatenating the feature vectors from both systems and training new SVM models. Furthermore, it is interesting that the fused system benefits from intersession variability compensation, an approach developed for acoustic features rather than for features derived from prosodic patterns like those used here.

Table 1: Results for unconditioned, conditioned, and combined prosodic sequence systems. AllSyl: all-syllable features, Constr: constrained features, HB: hard-bin transform, SB: soft-bin transform, SMK: smoothing kernel used, ISVC: Intersession variability compensation applied.

ID	System	EER / DCFx10 1-side	EER / DCFx10 8-side
1	AllSyl, HB	14.19 / 0.609	5.19 / 0.257
2	AllSyl, SB	13.65 / 0.601	4.91 / 0.241
3	AllSyl, SB + SMK	12.09 / 0.547	4.96 / 0.221
4	Constr, SB + SMK	13.33 / 0.544	4.62 / 0.222
5	3 + 4	11.55 / 0.487	4.28 / 0.181
6	3 + 4 + ISVC	10.36 / 0.461	3.65 / 0.156
7	MLLR + ISVC	4.00 / 0.197	2.14 / 0.073
8	6 + 7	3.72 / 0.167	1.74 / 0.056

Additional results in Table 1 illustrate the benefit of combining the prosody model with our single best system, a maximum likelihood linear regression (MLLR) SVM system [10]. The MLLR system is based on cepstral features, but makes use of information from an ASR system to obtain speaker-specific model transforms. It uses the difference between speaker-adapted Gaussian means and corresponding speaker-independent means as features, expressed as the coefficients of an affine transform (computed with MLLR) that obtains a speaker-dependent model from the speaker-independent model. The scores from the two systems are combined using an SVM with a linear inner product kernel. The combiner is trained using the scores obtained for SRE05 data, which we assumed to be reasonably representative of the SRE06 data. As can be seen, even though the MLLR system achieves superior performance alone, the prosody system provides significant added benefit, particularly in the 8-side training condition where the relative error reduction is 19% and 23% for the EER and DCF, respectively. We note that the improvement over a cepstral GMM (see below) in the 8-side condition is about 40%.

3.2. Importance in multisystem combination

From an applied perspective it is also important to ask whether the prosodic SVM system adds any information over the already remarkably good performance that one can obtain by using multiple systems—particularly multiple cepstral-based systems. The following analysis provides an answer, via the study of a larger set of systems run for the same evaluation data. The systems represent a snapshot of performance at a particular time. They use roughly similar amounts (although different specific subsets) of background data. The better systems (with the exception of the cepstral SVM) use intersession variability compensation.

Low-level systems include a cepstral GMM, a cepstral SVM, and a Gaussian supervector SVM. The cepstral GMM

system models cepstral feature distributions in the form of a mixture of Gaussian densities [11]. The GMM is adapted by reestimating the Gaussian means on the target speaker data. The cepstral SVM computes polynomials of the cepstral features and averages them over the entire conversation [12]. A feature vector consisting of a large number of these polynomial features characterizes the joint distribution of cepstral features. These feature vectors are then modeled by SVMs. The Gaussian supervector SVM system is based on the adapted target speaker GMM mentioned above [13]. Instead of modeling the cepstral features directly, it uses the adapted Gaussian means as features, stringing them into a long “supervector” that is then modeled as an SVM classifier input, similar to the cepstral SVM.

Four higher-level systems require ASR output. They include the MLLR system described earlier, two text-constrained duration GMM systems, and a word N-gram model. A phone-in-word duration GMM models the sequence of phone durations within specific words; a state-in-phone duration GMM models the durations of states (subphone-units) within phone HMMs [14]. Durations for different positions form a feature vector and are modeled in the adapted-GMM framework used for standard cepstral GMM cepstral systems. The word N-gram system [15, 7] models relative frequencies of frequent word unigrams, bigrams, and trigrams using an SVM. The system also encodes the duration (slow/fast) of frequent word types as part of the N-gram frequencies.

Table 2 gives results for all systems, and includes the results from Table 1 for the improved prosody SVM and MLLR systems. As expected, systems based on cepstral or cepstral-derived features show higher accuracy than longer-range systems. However, the prosody SVM system is the best-performing of the noncepstral higher-level systems, and with 8 sides of training data it approaches the performance of the Gaussian supervector SVM.

Table 2: Individual system results.

System	EER/DCFx10 1-side	EER/DCFx10 8-side
MLLR SVM	4.00 / 0.197	2.14 / 0.073
Cepstral GMM	4.75 / 0.216	2.79 / 0.107
Cepstral SVM	5.07 / 0.242	2.33 / 0.093
Gaussian supervector SVM	4.15 / 0.198	3.24 / 0.164
Prosody SVM	10.36 / 0.461	3.65 / 0.156
State-in-phone dur. GMM	16.03 / 0.705	8.07 / 0.423
Phone-in-word dur. GMM	22.24 / 0.874	9.30 / 0.420
Word N-gram SVM	23.46 / 0.815	9.95 / 0.446

Using the systems in Table 2, we ran an all-out combination, using the same combiner described above, for N systems, with N ranging from 2 to 8. We then found best results for each N , based on lowest DCF. Results are shown in Figure 1, for both the 1- and 8-side conditions. A fortunate property of these results is that the ordering of systems is cumulative (with only one local exception at $N=2$ for the 1-side condition). That is, the best result for N systems includes all systems used for $N-1$ systems. We can thus infer system importance from left to right.

The message to take away from the figure is that although cepstral systems are better individually than the prosody system, the prosody system is the third most important system for the 1-side training condition, and the second most important system for the 8-side condition. It is clearly more useful than the duration GMM systems and word N-gram system. What is

	MLLR SVM (H)	Supervisor SVM	Prosody Sequence SVM (H)	Phone-in-Word Duration GMM (H)	State-in-Phone Duration GMM (H)	Word-Duration N-gram SVM (H)	Cepstral GMM	Cepstral SVM	%EER	DCF (x10)
1 Best	■								4.00	0.197
2 Best	■						■		3.13	0.148
3 Best	■	■	■						2.86	0.139
4 Best	■	■	■	■					2.86	0.137
5 Best	■	■	■	■	■				2.80	0.136
6 Best	■	■	■	■	■	■			2.86	0.140
7 Best	■	■	■	■	■	■	■		2.64	0.141
8 Best	■	■	■	■	■	■	■	■	2.59	0.144

	MLLR SVM (H)	Prosody Sequence SVM (H)	Cepstral SVM	Word-Duration N-gram SVM (H)	State-in-Phone Duration GMM (H)	Phone-in-Word Duration GMM (H)	Supervisor SVM	Cepstral GMM	%EER	DCF (x10)
1 Best	■								2.13	0.0728
2 Best	■	■							1.74	0.0561
3 Best	■	■	■						1.59	0.0501
4 Best	■	■	■	■					1.59	0.0488
5 Best	■	■	■	■	■				1.59	0.0478
6 Best	■	■	■	■	■	■			1.54	0.0483
7 Best	■	■	■	■	■	■	■		1.64	0.0480
8 Best	■	■	■	■	■	■	■	■	1.64	0.0485

Figure 1: Results for the best N -way combination of systems, for 1- and 8-side training conditions, respectively. Filled boxes indicate which systems were selected. ‘H’ denotes higher-level systems.

particularly noteworthy is that the prosody SVM provides more complementary information than two of the cepstral systems for 1-side training, and than 3 of the cepstral systems for 8-side training. Since its utility increases with increasing training data (as do longer-range systems in general), it would be interesting to explore combined performance for even longer training conditions, although this is not currently supported by NIST evaluation data resources.

4. Conclusions

We have described four improvements to a prosodic SVM system, including soft binning of features, a smoothing kernel for the SVM, application of intersession variability compensation, and a new approach using text-constrained features. Analysis of constraints suggests that idiosyncratic prosodic behavior is best seen during personal pronouns and discourse-related POS classes (backchannels, fillers, and discourse markers), especially at pause boundaries. Results reveal that the improved prosody system performs surprisingly well, reducing the error rate of a cepstral MLLR SVM by as much as 23%. Analysis of system combinations using eight systems shows that the prosodic system is the third and second most important system for 1- and 8-side training, respectively. Although cepstral systems are better individual systems, the prosodic system offers more complementary information than do additional cepstral systems once one or two cepstral systems are present. These results point to a nonobvious strategy for future overall system development: because low-level systems are highly correlated, the

exploration of complementary systems based on higher-level features, such as the prosody system described here, should become only more important as lower-level systems continue to improve.

5. Acknowledgments

We thank Harry Bratt, Sachin Kajarekar, Andreas Stolcke, and Gokhan Tür for contributions. This work was supported by NSF Contract IIS-0544682. The views herein are those of the authors and do not represent the views of the funding agency.

6. References

- [1] E. Shriberg, “Higher-Level Features in Speaker Recognition”, in C. Müller, editor, *Speaker Classification I*, vol. 4343 of *Lecture Notes in Computer Science/AI*. Springer, Berlin, 2007.
- [2] E. Shriberg, L. Ferrer, S. Kajarekar, A. Venkataraman, and A. Stolcke, “Modeling prosodic feature sequences for speaker recognition”, *Speech Communication*, vol. 46, pp. 455–472, 2005.
- [3] L. Ferrer, E. Shriberg, S. Kajarekar, and K. Sonmez, “Parameterization of prosodic feature distributions for SVM modeling in speaker recognition”, in *Proc. ICASSP*, Honolulu, 2007.
- [4] L. Ferrer, K. Sonmez, and E. Shriberg, “Smoothing kernel for spatially related features and its application to speaker verification”, To appear in *Proc. Interspeech*, Antwerp, 2007.
- [5] A. Solomonoff, W. M. Campbell, and I. Boardman, “Advances in channel compensation for SVM speaker recognition”, in *Proc. ICASSP*, vol. 1, pp. 629–632, Philadelphia, 2005.
- [6] A. Stolcke, H. Franco, R. Gadde, M. Graciarena, K. Precoda, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng, Y. Huang, B. Peskin, I. Bulyko, M. Ostendorf, and K. Kirchhoff, “Speech-to-text research at SRI-ICSI-UW”, in *Proc. DARPA RT-03 Workshop*, Boston, 2003.
- [7] S. Kajarekar, L. Ferrer, E. Shriberg, K. Sonmez, A. Stolcke, A. Venkataraman, and J. Zheng, “SRI’s 2004 NIST speaker recognition evaluation system”, in *Proc. ICASSP*, vol. 1, pp. 173–176, Philadelphia, 2005.
- [8] S. Kajarekar and A. Stolcke, “NAP and WCCN: Comparison of approaches using MLLR-SVM speaker verification system”, in *Proc. ICASSP*, Honolulu, 2007.
- [9] A. Martin, “Evaluations of Automatic Speaker Classification Systems”, in C. Müller, editor, *Speaker Classification I*, vol. 4343 of *Lecture Notes in Computer Science/AI*. Springer, Berlin, 2007.
- [10] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman, “MLLR transforms as features in speaker recognition”, in *Proc. Interspeech*, pp. 2425–2428, Lisbon, 2005.
- [11] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, “Speaker verification using adapted Gaussian mixture models”, *Digital Signal Processing*, vol. 10, pp. 181–202, 2000.
- [12] W. M. Campbell, “Generalized linear discriminant sequence kernels for speaker recognition”, in *Proc. ICASSP*, vol. 1, pp. 161–164, Orlando, 2002.
- [13] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, “Support vector machines using GMM supervectors for speaker verification”, *IEEE Signal Processing Letters*, vol. 13, pp. 308–311, May 2006.
- [14] L. Ferrer, H. Bratt, V. R. R. Gadde, S. Kajarekar, E. Shriberg, K. Sonmez, A. Stolcke, and A. Venkataraman, “Modeling duration patterns for speaker recognition”, in *Proc. EUROSPEECH*, pp. 2017–2020, Geneva, 2003.
- [15] G. Doddington, “Speaker recognition based on idiolectal differences between speakers”, in *Proc. EUROSPEECH*, pp. 2521–2524, Aalborg, 2001.