NONPARAMETRIC FEATURE NORMALIZATION FOR SVM-BASED SPEAKER VERIFICATION

Andreas Stolcke[1] *Sachin Kajarekar*[1] *Luciana Ferrer*[2]

[1]Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA
[2]Department of Electrical Engineering, Stanford University, Stanford, CA, USA
{stolcke,sachin,lferrer}@speech.sri.com

ABSTRACT

We investigate several feature normalization and scaling approaches for use in speaker verification based on support vector machines. We are particularly interested in methods that are "knowledge-free" and work for a variety of features, leading us to investigate MLLR transforms, phone N-grams, prosodic sequences, and word N-gram features. Normalization methods studied include mean/variance normalization, TFLLR and TFLOG scaling, and a simple nonparametric approach: rank-normalization. We find that rank-normalization is uniformly competitive with other methods, and improves upon them in many cases.

Index Terms— Speaker verification, SVM modeling, feature normalization, kernel design.

1. INTRODUCTION

A great deal of progress and innovation in speaker recognition has been brought about by the use of support vector machines (SVMs) as speaker models. Through the ingenious design of features and kernels, SVMs have been applied to speaker modeling for a wide range of phenomena, from low-level cepstral observations to high-level prosodic and lexical patterns. The typical steps involved in SVM speaker modeling are

1. Raw feature extraction: This step computes the fundamental observations associated with a speech sample, such as frame-level cepstral features, phone N-grams or word N-grams [1].
2. Feature reduction: This step reduces the variable-length sequence of observations associated with a speech sample to a fixed-length vector. For example, cepstral observations are condensed into averages of polynomials over the cepstral coefficients [2], or N-grams are represented by the vector of relative frequencies for a fixed subset of (frequent) N-grams [3].
3. Feature normalization: Here the components of the fixed feature vector are scaled or warped so as to enable more effective modeling of speaker differences.
4. Standard kernel computation: The reduced, normalized feature vector is combined with a standard kernel function, such as linear, quadratic, or exponential, as is available with most SVM implementations.

The boundaries between these steps are not defined a priori, but have emerged as useful generalizations from common practice. For example, one can always argue about where to draw the line between raw and reduced features, or between normalization and kernel, because ultimately the entire process defines the kernel function used by the SVM. However, to the extent that many different systems share common approaches to feature extraction, reduction, and normalization, it is useful to study the above steps individually.

In this paper, we focus on feature normalization, precisely because a common repertoire of methods has been proposed in the literature and applied to a range of features. Also, we will simplify the study by fixing the standard kernel function to be the linear vector inner product, since it has empirically been shown to give best or near-best results with a great many feature sets commonly used in speaker recognition (after suitable feature design and normalization, of course). Intuitively, feature normalization is needed because SVMs (unlike some other classifier architectures) are not invariant to the scale of their input feature spaces: multiplying a feature dimension by a fixed constant gives that dimension more weight in the value of the SVM objective function and, therefore, in the choice of the decision boundary. Therefore, in the absence of prior knowledge, one should choose a normalization method that leaves all feature dimensions in a comparable range.

In the remainder of the paper, we will examine several normalization methods proposed in the literature that all have this intuitive property, and apply them to a set of speaker models spanning the range from acoustic to stylistic, or "low-level" to "high-level." Section 2 summarizes the data and systems used in our study. Section 3 describes the normalization methods examined. Section 4 presents and discusses the results obtained, followed by some brief conclusions and suggestions for future work in Section 5.

2. MODELS AND DATA

The following speaker modeling approaches were used to generate features for our study.

2.1. MLLR transform features

The raw features used by this approach are the PLP-cepstrum vectors used by the second pass of the SRI speech recognizer. Features are reduced to fixed-length vectors by computing maximum likelihood linear regression (MLLR) adaptation transforms for each conversation side, and then using the transform coefficients as derived features [4, 5]. The system computes eight 39×40-dimensional affine transforms, corresponding to eight phone classes, for the Gaussian means of the speech recognition models. This process is carried out twice, using a male and a female recognition model, respectively, regardless of the speaker gender, thereby yielding a vector of $2 \times 8 \times 39 \times 40 = 24,960$ feature components.

Table 1. Data sets used

Test set	SRE-05 English		SRE-06 Common Cond.	
Training	1-side	8-side	1-side	8-side
Models	587	463	517	483
Trials	26270	20683	24013	17547

2.2. Phone N-gram features

Here we use the phone recognition-based modeling paradigm of [3] with the lattice-based refinement of [6]. An English open-loop phone recognizer is run on each conversation side, generating lattices. The expected frequencies for unigrams, bigrams, and trigrams are computed (i.e., N-grams are weighted according to their posterior probability of occurrence in the lattice). The 8,483 most frequent phone N-grams from the background data are determined a priori, and a test feature vector consists of the relative frequencies of those N-grams in the conversation side.

2.3. Prosodic sequence features

This system models syllable-based prosodic features (SNERFs) [7]. Raw features are based on estimated F0, energy, and phone duration information extracted over syllables inferred via automatic syllabification of automatic speech recognition (ASR) output; inter-word pauses and their durations are also represented. Prosodic feature sequences are reduced to fixed-length vectors by a particular implementation of the Fisher score [8]. Features modeling sequences of one, two, and three syllables are used. The resulting feature vector has dimension 38,314.

2.4. Word N-gram features

This model captures idiosyncratic word usage, as suggested by [1], but using the same N-gram frequency vector representation as for phone N-grams (see above) [3]. This approach was shown to be superior to likelihood ratios obtained from languages models [9]. The 126k most frequent word N-grams up to length 3 are extracted from the 1-best hypotheses of the ASR system over the background corpus. The test feature vector corresponds to the relative frequencies of those N-gram types in the test conversation side. Note that computation with these high-dimensional feature vectors benefits greatly from the sparse nature of the vectors, i.e., most components of a given vector are typically zero. This will be an important consideration in feature normalization.

2.5. Data

The models and normalization methods (to be described next) were tested on subsets of the NIST 2005 and 2006 speaker recognition evaluation (SRE) data sets. The data was drawn from the LDC Mixer data collection, and was composed of telephone conversations of about 5 minutes in length (about 2.5 minutes of speech per conversation side) [10, 11]. We studied the 4-wire (separate channels) speaker verification task with both 1 and 8 conversation sides as target training data, and 1 conversation side as test data. For SRE-05, all English-language trials of these conditions were included; for SRE-06, only the "common condition" trials, which by definition also used only English language, were used. Table 1 gives an overview of the amount of data and number of trials in each condition.

For background (impostor) training data, we used portions of various telephone speech corpora available from LDC and NIST, including Switchboard-II Phases 2 and 3 (SRE-03), Switchboard-II Phase 5, Fisher Phase 1, and Mixer (SRE-04), comprising a total of 4814 conversation sides. All training and test data was transcribed automatically by various versions of SRI's telephone recognizer [12] to generate the phone-level and word-level transcripts, MLLR transforms, and alignments needed to extract the features used in our various speaker models.

3. NORMALIZATION METHODS

3.1. Mean/variance normalization

One of the most common approaches for feature normalization, especially when close-to-Gaussian distribution is assumed, is subtraction of the population mean and scaling to achieve unit variance:

$$x_i' = \frac{x_i - \bar{x}_i}{\sigma_{x_i}} \tag{1}$$

where x_i is the raw value of the ith feature, $\bar{x}_i$ is the feature mean, σ_{x_i} its standard deviation, and x_i' the normalized feature value.

This method has the intended effect of roughly equating the dynamic ranges along each dimension, although this is true only to the extent that the different feature distributions follow similar shapes. Thus, this normalization scheme might be inappropriate for systems with very heterogeneous features, such as SNERFs, where, for example, a pitch-based feature might be log-normally distributed, whereas a duration feature is more likely to have an exponential or Poisson-shaped distribution.

Note that mean normalization causes sparse feature vectors to become less sparse, resulting in additional computational expense when dealing with large sparse feature spaces, such as used by the word N-gram model. Fortunately, SVMs are invariant to additive translations in the feature space, so a simple variance normalization $x_i' = \frac{x_i}{\sigma_{x_i}}$ is sufficient for our purposes.

3.2. TFLLR scaling

Campbell et al. [3] have proposed a kernel for N-gram features that approximates a log likelihood ratio computed from N-gram frequencies (term frequency log likelihood ratio, or TFLLR). The method boils down to a linear inner product kernel applied to a feature vector consisting of the N-gram frequencies with scaling factors applied:

$$x_i' = d_i x_i \tag{2}$$

where x_i is the frequency of the ith N-gram in the speech sample, and d_i is a weight proportional to the square root of the inverse overall frequency f_i of that N-gram:

$$d_i = \sqrt{\frac{1}{f_i}} \tag{3}$$

The intuitive justification for the inverse frequency weighting is that overall rare N-grams carry more information, so their d_i weight will be larger than for ubiquitous N-grams. Numerically, the effect is again one that tends to make the dynamic ranges more similar. This occurs because rare N-grams have small raw feature values, and therefore small absolute variation, but larger scaling factors (though the scaling is attenuated by the square root function).

3.3. TFLOG scaling

In [13], a generalization of TFLLR scaling is proposed that is empirically better suited for very sparse features, such as word N-grams. In this approach, the feature scaling weight is computed as

$$d_i = \min\{C, g(\frac{1}{f_i})\} \tag{4}$$

where C is a constant that sets the maximum feature weight, and g is a squashing function that compresses the dynamic range of the inverse term frequency (replacing the square root in (3)). They find empirically that $g(r) = \log r + 1$ with $C = 10$ gives good results on word N-grams, and call this the TFLOG kernel (or scaling, in our terminology). The benefit of TFLOG is to limit the impact of very rare N-grams, which would otherwise have their noisy frequency estimates amplified by large scale factors.

3.4. Rank normalization

Finally, we investigate a nonparametric approach to normalization that is not based on estimating any distribution parameters. The basic idea is to replace each test feature value by its *rank* in the background data, followed by a normalization to the unit interval. For example, if a value x_i would be ranked 10th out of 100 among the background data values, its normalized values would be $10/100 = 0.1$. If a value does not occur exactly in the background data the rank of the nearest value below it is used, and a value exceeding the maximum observed value is assigned a normalized value of 1. Formally,

$$x_i{}' = \frac{|\{y_i \in B : y_i < x_i\}|}{|B|} \tag{5}$$

where B is the background data set, and $|X|$ denotes the cardinality of set X.

A useful consequence of this method is that zero values in non-negative distributions are mapped to zero, thus preserving sparseness of feature vectors (note that N-gram frequency features are non-negative).

Rank normalization is a priori attractive because of its lack of assumptions about the underlying distributions. The strongest motivation for it comes from two related properties: any feature distribution, to the extent that it matches the background distribution, is warped to a uniform distribution over the interval $[0, 1]$. Conversely, the kernel-induced distance between datapoints

$$D(x, y)^2 = K(x, x) + K(y, y) - 2K(x, y) = ||x - y||^2 \tag{6}$$

(in the case of a linear kernel $K(x, y)$) is such that along any single feature dimension, two points x and y are separated by a distance proportional to the number of background data samples falling between x and y. In other words, the normalization stretches the feature space in areas of high population density and shrinks it in areas of low density. Another consequence is that rank normalization is invariant under any monotonic transformation of the input feature space. For example, it would yield the same results if the input features were first subjected to any of the normalization methods discussed earlier (with the exception of TFLOG for $C < \infty$).

Note that we have previously advocated use of rank normalization for certain SVM models [4, 7]; however, this is the first time we are conducting a systematic study of different normalization methods for a range of speaker models.

Table 2. Results for 1-side training condition. The best result for each model and condition is highlighted.

	SRE-05		SRE-06	
Normalization	EER	DCF	EER	DCF
MLLR transforms				
None	6.15	.222	5.29	.246
Variance	5.34	.175	3.94	.188
Rank	**5.22**	**.161**	**3.61**	**.177**
SNERF prosodic features				
None	15.57	.612	14.19	.625
Variance	13.96	.571	14.08	.622
Rank	**13.88**	**.544**	**13.65**	**.601**
Phone N-grams				
None	14.64	.536	12.30	.558
Variance	12.62	.457	10.84	.519
TFLLR	12.66	.468	10.73	.512
Rank	**12.18**	**.428**	**10.30**	**.444**
Word N-grams				
None	24.76	.854	22.98	.837
Variance	32.04	.982	31.07	.983
TFLOG, $C = 10$	23.10	.824	21.79	.806
TFLOG, $C = \infty$	23.14	.820	**21.63**	**.800**
Rank	**22.49**	**.799**	23.19	**.800**

4. RESULTS

We tested the various normalization methods across the range of SVM speaker verification systems described above. Results are given in terms of both equal error rate (EER) and minimum decision cost function times 10 (DCF). Tables 2 and 3 give results for the 1-side and 8-side training conditions, respectively. Note that for N-gram-based systems we examined the weighting normalizations proposed by Campbell et al. as our baselines, whereas for the MLLR and prosodic features that role is played by variance normalization. Also note that TFLLR and TFLOG apply only to count-based features, i.e., they are not applicable to MLLR transforms and SNERF features.

Results show that all systems benefit from normalization, as expected. Except for word N-grams, rank normalization delivers results that are better than other methods. For phone N-grams, variance normalization perform about the same, and is not much worse than rank normalization. Somewhat surprisingly, TFLLR is not as good as either variance or rank normalization for phone N-grams, in spite of having been originally introduced for this type of model [3]. Rank normalization also gives the best, or close to the best results, for 1-side training with word N-grams, but TFLOG seems to be consistently better for 8-side training. This seems to indicate that TFLOG is better in principle, but that rank normalization is more robust to the less reliable frequency estimates obtained from 1-conversation-side training. Unlike Campbell et al. [14], we find that TFLOG with $C = 10$ is not better than $C = \infty$, which could be due to any number of differences, such as the recognition systems used. Also note that variance normalization does a reasonable job for all models, except for word N-grams, where both frequency and variance estimates are likely to be very unreliable for rare N-grams, and small variances have an amplifying effect on the noisy frequency estimates.

Finally, we note that the best-performing model, the MLLR-SVM, shows the largest relative error reductions from normalization. This result is at odds with a recent report showing rank normalization

Table 3. Results for 8-side training condition. The best result for each model and condition is highlighted.

	SRE-05		SRE-06	
Normalization	EER	DCF	EER	DCF
MLLR transforms				
None	3.04	.083	2.44	.091
Variance	2.56	.066	**1.99**	**.073**
Rank	**2.46**	**.0.62**	2.19	.074
SNERF prosodic features				
None	n/a	n/a	6.16	.287
Variance	n/a	n/a	5.31	.275
Rank	**5.74**	**.207**	**4.91**	**.241**
Phone N-grams				
None	6.47	.235	5.82	.282
Variance	5.16	.170	4.83	.224
TFLLR	5.16	.180	5.02	.231
Rank	**4.97**	**.151**	**4.48**	**.208**
Word N-grams				
None	9.89	.444	9.95	.476
Variance	18.33	.807	18.81	.845
TFLOG, $C = 10$	8.93	.381	**8.96**	**.422**
TFLOG, $C = \infty$	**8.78**	.377	**8.96**	**.422**
Rank	9.51	**.355**	10.00	.424

to be worse than no normalization for MLLR-SVM modeling [15]. However, there were many differences in the way transforms were estimated in the two cases, which could account for the discrepancy.

5. CONCLUSIONS AND FUTURE DIRECTIONS

We have argued that nonparametric normalization of SVM features based on ranking of feature values provides an a priori attractive and intuitive approach in the absence of specific knowledge about the feature distributions. Experimentally, rank normalization is competitive, and in most cases superior to other commonly used methods, for a range of SVM-based speaker recognition systems.

Much work remains to be done in the area of feature normalization. For one thing, we have not yet systematically examined the effects of feature compensation (such as nuisance attribute projection) or score normalization (such as TNORM) on the choice of feature normalization. Also, it is intriguing to consider the possibility of multidimensional variants of rank normalization, i.e., methods that warp the joint distribution of features toward uniform, and not just the marginal distributions along individual feature dimensions. The idea is promising given that, in the realm of parametric approaches, covariance normalization has been shown superior to diagonal variance normalization for SVMs [16].

6. ACKNOWLEDGMENTS

This work was funded in part by a Department of Defense KDD Award via NSF IIS-0544682, as well as through a development contract with Sandia National Laboratories. The views herein are those of the authors and do not necessarily represent the views of the funding agencies.

7. REFERENCES

[1] G. Doddington, "Speaker recognition based on idiolectal differences between speakers", in P. Dalsgaard, B. Lindberg, H. Benner, and Z. Tan, editors, *Proc. EUROSPEECH*, pp. 2521–2524, Aalborg, Denmark, Sep. 2001.

[2] W. M. Campbell, "Generalized linear discriminant sequence kernels for speaker recognition", *in Proc. ICASSP*, vol. 1, pp. 161–164, Orlando, FL, May 2002.

[3] W. M. Campbell, J. P. Campbell, D. A. Reynolds, D. A. Jones, and T. R. Leek, "Phonetic speaker recognition with support vector machines", in S. Thrun, L. Saul, and B. Schölkopf, editors, *Advances in Neural Information Processing Systems 16*, pp. 1377–1384, Cambridge, MA, 2004. MIT Press.

[4] 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, Sep. 2005.

[5] A. Stolcke, S. S. Kajarekar, L. Ferrer, and E. Shriberg, "Speaker recognition with session variability normalization based on MLLR adaptation transforms", *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, pp. 1987–1998, Sep. 2007.

[6] A. O. Hatch, B. Peskin, and A. Stolcke, "Improved phonetic speaker recognition using lattice decoding", *in Proc. ICASSP*, vol. 1, pp. 169–172, Philadelphia, Mar. 2005.

[7] 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, Special Issue on Quantitative Prosody Modelling for Natural Speech Description and Generation.

[8] L. Ferrer, E. Shriberg, S. Kajarekar, and K. Sonmez, "Parameterization of prosodic feature distributions for SVM modeling in speaker recognition", *in Proc. ICASSP*, vol. 4, pp. 233–236, Honolulu, Apr. 2007.

[9] S. 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, Mar. 2005.

[10] A. Martin, D. Miller, M. Przybocki, J. Campbell, and H. Nakasone, "Conversational telephone speech corpus collection for the NIST speaker recognition evaluation 2004", *in Proceedings 4th International Conference on Language Resources and Evaluation*, pp. 587–590, Lisbon, May 2004.

[11] C. Cieri, L. Corson, D. Graff, and K. Walker, "Resources for new research directions in speaker recognition: The Mixer 3, 4 and 5 corpora", *in Proc. Interspeech*, pp. 950–954, Antwerp, Aug. 2007.

[12] A. Stolcke, B. Chen, H. Franco, V. R. R. Gadde, M. Graciarena, M.-Y. Hwang, K. Kirchhoff, A. Mandal, N. Morgan, X. Lin, T. Ng, M. Ostendorf, K. Sönmez, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng, and Q. Zhu, "Recent innovations in speech-to-text transcription at SRI-ICSI-UW", *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 14, pp. 1729–1744, Sep. 2006, Special Issue on Progress in Rich Transcription.

[13] W. M. Campbell, J. P. Campbell, D. A. Reynolds, D. A. Jones, and T. R. Leek, "High-level speaker verification with support vector machines", *in Proc. ICASSP*, vol. 1, pp. 73–76, Montreal, May 2004.

[14] W. M. Campbell, J. P. Campbell, T. P. Gleason, D. A. Reynolds, and W. Shen, "Speaker verification using support vector machines and high-level features", *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 15, pp. 2085–2094, Sep. 2007.

[15] Z. N. Karam and W. M. Campbell, "A new kernel for SVM MLLR based speaker recognition", *in Proc. Interspeech*, pp. 290–293, Antwerp, Aug. 2007.

[16] A. O. Hatch and A. Stolcke, "Generalized linear kernels for one-versus-all classification: Application to speaker recognition", *in Proc. ICASSP*, vol. 5, pp. 585–588, Toulouse, May 2006.