Dimensionality Reduction for Using High-Order n-grams in SVM-Based Phonotactic Language Recognition

Mikel Penagarikano, Amparo Varona, Luis Javier Rodriguez-Fuentes, German Bordel

GTTS, Department of Electricity and Electronics
University of the Basque Country, Spain

mikel.penagarikano@ehu.es

Abstract

SVM-based phonotactic language recognition is state-of-the-art technology. However, due to computational bounds, phonotactic information is usually limited to low-order phone n-grams (up to $n = 3$). In a previous work, we proposed a feature selection algorithm, based on n-gram frequencies, which allowed us work successfully with high-order n-grams on the NIST 2007 LRE database. In this work, we use two feature projection methods for dimensionality reduction of feature spaces including up to 4-grams: Principal Component Analysis (PCA) and Random Projection. These methods allow us to attain competitive performance even for small feature sets (e.g. of size 500). Systems were built by means of open software (BUT phone decoders, *HTK*, *SRILM*, *LIBLINEAR* and *FoCal*) and experiments were carried out on the NIST 2009 LRE database. Best performance was attained by using the feature selection algorithm to get around 11500 features: 1.93% EER and $C_{LLR} = 0.413$. When considering smaller sets of features, PCA provided best performance. For instance, using PCA to get a 500-dimensional feature subspace yielded 2.15% EER and $C_{LLR} = 0.457$ (25% improvement with regard to using feature selection).

Index Terms: Phonotactic Language Recognition, SVM, High-Order n-grams, Feature Selection, Principal Component Analysis, Random Projection

1. Introduction

For Spoken Language Recognition (SLR) tasks, two main complementary approaches are typically used [1]: *low level* acoustic modeling and *high level* phonotactic modeling. To model the target language, *low level* acoustic systems take information from the spectral characteristics of the audio signal, whereas *high level* phonotactic systems use sequences of phones produced by Parallel Phone Recognizers (PPR).

In this paper, we focus on the currently most common phonotactic approach: counts of phone n-grams are used to build feature vectors which feed a discriminative classifier based on Support Vector Machines (SVM) [2]. In general, N phone decoders are applied to the input utterance, yielding N phone decodings. The output of each decoder i ($i \in [1, N]$) is scored for each target language j ($j \in [1, L]$), by applying the SVM model $\lambda(i, j)$ (estimated using the outputs of the phone decoder i for a training database, taking j as the target language). Scores for the subsystem i are calibrated, typically by means of a Gaussian backend. Finally, $N \times L$ calibrated scores are fused applying linear logistic regression, to get L final scores for which a minimum expected cost Bayes decision is taken, according to application-dependent language priors and costs (see [3] for details).

The performance of each phone recognizer can be increased significantly by computing the statistics from phone lattices instead of 1-best phone strings [4], since lattices provide richer and more robust information. Another way to increase system performance is the use of high-order n-gram counts, which are expected to contain more discriminant (more language-specific) information. Since, the number of n-grams grows exponentially as n increases, dimensionality reduction techniques must be applied to get SVM vectors of a reasonable size. Usually feature selection methods are applied [5]: (1) feature selection based on frequency (low frequency n-grams are discarded); and (2) discriminative feature selection, which takes into account the rank of feature weights in the SVM vectors (least discriminant n-grams are discarded) [5] [6]. In both cases, optimal selection requires building *complete* vectors (i.e. vector containing all the components). Again, the huge amount of n-grams for $n \geq 4$ makes it computationally unfeasible a brute-force approach to selecting n-grams. In [5], a kind of suboptimal expansion has been proposed: starting from a relatively small trigram SVM system, a 4-gram SVM system is built by using an alternating wrapper/filter method. In the wrapper step, the most discriminant/frequent trigrams are selected. Then, in the filter step, the subset of 4-grams is generated by appending/prepending each phone in the phone set to each selected trigram.

In a recent work [7], we have proposed a new n-gram selection algorithm that allows the use of high-order n-grams to improve the performance of a baseline system based on trigram SVMs. The algorithm requires one single parameter: M, the desired number of features, and works by dynamically updating a ranked list of the most frequent units (from unigrams to n-grams), retaining only those units whose counts are higher than a given threshold. Finally, after processing all the training data, the M most frequent units are used.

In this work, we propose and evaluate a hybrid approach that combines feature selection and feature projection for dimensionality reduction. First, the selection algorithm proposed in [7] is applied to get an optimal medium-size set of features, an then a feature projection method is applied to get smaller sets of features with low performance degradation. In [8], Principal Component Analysis (PCA) was successfully applied as a feature transformation method on the NIST 2009 LRE database. Random Projection has been also proposed for other applications, such as face recognition and information retrieval in text documents [9], the original high-dimensional data being projected onto a lower-dimensional random subspace. In this work, both PCA and Random Projection have been tested for dimensionality reduction after feature selection.

This work has been supported by the University of the Basque Country under grant GIU10/18, by the Government of the Basque Country under program SAIOTEK (project S-PE10UN87) and by the Spanish MICINN under Plan Nacional de I+D+i (project TIN2009-07446, partially financed by FEDER funds).

The rest of the paper is organized as follows. Section 2 presents the experimental setup and the baseline phonotactic language recognition system used in this work. Section 3 describes the proposed feature projection methods. Results obtained in language recognition experiments on the NIST 2009 LRE database are presented in Section 4. Finally, conclusions are summarized in Section 5.

2. Experimental Setup

2.1. Phone lattices

An energy-based voice activity detector was applied in first place, which split and removed long-duration non-speech segments from signals. Then, the Temporal Patterns Neural Network (TRAPs/NN) phone decoders, developed by the Brno University of Technology (BUT) for Czech (CZ), Hungarian (HU) and Russian (RU) [10], were applied to perform phone tokenization. Before processing phone sequences, non-phonetic units: *int* (intermittent noise), *pau* (short pause) and *spk* (non-speech speaker noise) were mapped to a single non-phonetic unit. After that, the number of units was 43 for Czech, 59 for Hungarian and 50 for Russian. Since each BUT decoder runs its own acoustic front-end, channel compensation and noise reduction for all the systems presented in this paper relied on the acoustic front-end provided by BUT decoders.

Posterior probabilities from BUT recognizers were used as input to the HVite decoder from HTK [11] to produce phone lattices, which encode multiple phonetic hypotheses with acoustic likelihoods. Finally, the *lattice-tool* from *SRILM* [12] was used to produce the expected counts of phone n-grams.

2.2. Ranked Sparse Representation

In this work, up to 4-grams were considered. Therefore, using the raw SVM feature space became almost unfeasible, due to its huge dimension: the number of possible 4-grams could be up to 59^4. A sparse representation was used instead, which involved only the most frequent features. That is, instead of using a full space representation, features were ranked according to their counts on the training dataset, using a feature selection algorithm based on frequency [7], and only those features (unigrams + bigrams + ... + n-grams) with the M highest counts were considered.

The selection algorithm works by periodically updating a ranked list (a hash table indexed by features and storing counts) of the most frequent units, so it doesn't need to index all the possible n-grams, but just a relatively small subset of them. The process involves accumulating counts until their sum gets higher than K, and updating the ranked list of units by retaining only those counts higher than a given threshold τ. At each update, all the counts lower than τ are implicitly set to zero. This means that the selection process is *suboptimal*, since many counts are discarded at each update. The algorithm outputs the M leading items of the ranked list; note that K and τ must be tuned so that enough number of alive counts (at least, M) are kept at each update. In this work, we heuristically fixed K=10^6 and τ=0.1 to get about $M = 100000$.

Though M (the number of selected features) can take a high value, given an input utterance, most features have null counts and are not explicitly included in the representation (recall that a sparse representation is used), so the actual size of the SVM feature vector is far less than M. Note also that the longer an utterance is, the more dense the feature vector is, and therefore, test utterances (which are around 30 seconds long) produce more sparse feature vectors than training utterances (see Table 1 for details).

2.3. SVM modeling

The SLR systems developed in this work follow the SVM phonotactic approach. SVM vectors consist of counts of features representing the phonotactics of an input utterance, in particular phone n-grams up to $n = 4$ weighted as in [5]. A Crammer and Singer solver for multiclass SVMs with linear kernels has been applied, by means of LIBLINEAR [13], which has been modified by adding some lines of code to compute regression values. Finally, systems are built by fusing the scores of three calibrated SVM-based phonotactic subsystems (one per BUT decoder). The *FoCal* toolkit has been used for calibration and fusion (see [3] for details).

2.4. Train, development and evaluation datasets

Train and development data were limited to those distributed by NIST to all 2009 LRE participants [14]: (1) conversational telephone speech from previous LREs: the Call-Friend Corpus, the OHSU Corpus provided by NIST for LRE05 and the development corpus provided by NIST for the 2007 LRE; and (2) narrow band (telephone channel) speech segments from *Voice Of America* (VOA) broadcast news recordings (provided by NIST for the 2009 LRE).

A set of 64 languages/dialects was defined. Each of them was mapped either to a target language of the NIST 2009 LRE or to Out-Of-Set (OOS). For example, Mainland and Taiwan from the NIST 2007 LRE and Mandarin from VOA were all mapped to Mandarin, whereas Arabic was mapped to OOS. Persian and Farsi were mapped to the same language, as was properly pointed in [15]. For the languages appearing in VOA recordings, the longest speech segment out of each file was posted to the train dataset, with a minimum of 225 segments per language. The number of segments extracted per file was relaxed (augmented) for those languages with few files in VOA. The train dataset consisted of 43278 segments, which amounted to 2286 hours. For development, speech segments lasting around 30 seconds (between 25 and 35 seconds) were randomly extracted, using no more than 2 segments per file, and a minimum of 225 segments per language. The development dataset consisted of 13269 segments, which amounted to around 110 hours. Evaluation was carried out on the NIST 2009 LRE evaluation corpus, specifically on the 30-second, closed-set condition (primary evaluation task).

2.5. Evaluation measures

In this work, systems will be compared in terms of Equal Error Rate (EER), which, along with DET curves, is the most common way of comparing the performance of language recognition systems, but also in terms of the so called C_{LLR} [16], an alternative performance measure used in NIST evaluations.

3. Feature Dimensionality Reduction

When using high order n-grams in SVM-based Phonotactic Language Recognition, the dimensionality of the feature vectors may be intractable in terms of memory and time requirements. Dimensionality reduction can be carried out by selecting the most relevant features, based on their relative frequency or even on more complex discriminative properties [5, 6]. On the other hand, by applying PCA, the source feature space can

be projected onto a lower-dimensional orthogonal subspace that captures as much of the variation as possible [8]. Somehow related to PCA, Random Projection is a computationally low-cost method that projects features onto a lower-dimensional orthogonal random subspace without introducing a significant distortion [9].

However, when dealing with high-dimensionality spaces even a simple frequency ranking could be unfeasible, as has been noted in Section 2.2. Therefore, as a first step, a frequency based selection may be carried out in order to reduce the number of features to a tractable value. In this work, such value was set to 100000. Table 1 shows the accumulated posterior probability (in %) –computed on the set of n-grams of the train dataset–, as a function of the number of features retained by the feature selection algorithm.

Starting from the 100000 most frequent features, dimensionality reduction onto feature subspaces of 10000 to 500 dimensions was performed by means of frequency-based feature selection, PCA and Random Projection. Details are given in the following sections.

3.1. Frequency-based feature selection

The same ranking used to select the 100000 most frequent features was used to further reduce the feature set. Unlike the projection approaches (addressed below), which produce dense vectors, the feature selection algorithm produces sparse vectors. Since the SVM software uses a sparse data representation, the key property (in terms of time and memory requirements) is not the dimension of the feature space, but the actual size of feature vectors (i.e. how many features have non-null counts on average). Therefore, the selection parameter M was set to values for which the average vector size matched one of the target subspace dimensions used in the projection approaches. Table 1 shows the average size of feature vectors for the train and test datasets and different number of features. Note that test segments produced more sparse feature vectors because they were shorter than train segments.

3.2. Principal Component Analysis

PCA projection was done as proposed in [8]. A language balanced random subset of the train dataset (22500 segments out of 43278) was used. PCA was performed on the SVM input space (consisting of weighted expected n-gram counts), and principal components were computed using the randomized algorithm for PCA as defined in [17] (Section 4.3 - *A modified algorithm*).

3.3. Random projection

Random projection is a well-known method used in dimensionality reduction. It is based on the Johnson-Lindenstrauss lemma [18], which states that if a set of points in a high-dimensional Euclidean space is projected onto a lower dimensional random subspace, pairwise distances are nearly preserved. Instead of finding an orthogonal random projection matrix, it is possible to simply use a random matrix, because in a high-dimensional space, vectors having random directions might be sufficiently close to orthogonal [9]. The dense gaussian random matrix R can be further simplified by a sparse random matrix with elements drawn from the Achlioptas distribution [19]:

$$r_{ij} = \sqrt{3} \cdot \begin{cases} +1 & with\, probability\ 1/6 \\ 0 & with\, probability\ 2/3 \\ -1 & with\, probability\ 1/6 \end{cases} \quad \begin{array}{l} 1 \le i \le dim \\ 1 \le j \le dim_{reduced} \end{array}$$

Preliminary experiments showed that performance degradation when using random projection was drastically reduced (from 11% EER down to 4% EER when reducing feature dimensionality to 500) if the random vectors used to build the matrix R were weighted with the background average feature values, $\widetilde{r}_{ij} = r_{ij} \cdot \overline{f_i}$.

4. Results

The NIST 2009 LRE test set was used to evaluate systems implementing the proposed approaches, more specifically the 30-second, closed-set condition (core condition). Note again that we call *system* to the fusion of three subsystems, each corresponding to a TRAPS/NN phone decoder (for Czech, Hungarian and Russian).

Table 1 shows EER and C_{LLR} performance using the *frequency-based feature selection* algorithm for different values of M (number of selected features). M ranged from 27732 down to 526, for the average size of feature vectors matched the target dimensions used in PCA and Random Projection (10000 down to 500).

Table 1: Average size of feature vectors, accumulated posterior probability (%), % EER and C_{LLR} on the NIST 2009 LRE closed-set 30s evaluation set, for various 4-gram SVM systems working on feature sets obtained with the *frequency-based feature selection* algorithm.

	Avg Vector Size		**AccProb**		
M	**train**	**test**	**(%)**	**%EER**	$\mathbf{C_{LLR}}$
100000	17876	5670	95	1.98	0.421
27732	10000	3773	86	1.97	0.416
18685	8000	3233	83	1.95	0.412
11697	6000	2648	79	1.93	0.413
6417	4000	1997	73	1.96	0.427
4319	3000	1631	70	2.03	0.432
2556	2000	1302	65	2.25	0.468
1125	1000	726	58	2.42	0.511
526	500	416	51	2.92	0.612

In contrast to previous results ([8] and [6]), where performance was significantly compromised when frequency-based feature selection was applied, we did not observe any degradation but a slight improvement when up to 6417 (the most frequent) features were kept (which is equivalent to projecting down to 4000 dimensions). Note that in [8] feature vectors contained the square roots of expected n-gram counts, and in [6] n-gram probabilities were used, whereas we used *weighted* n-gram probabilities (see [5]). In our experiments, best performance was achieved when using the 11697 most frequent features (which is equivalent to projecting down to 6000 dimensions). Such a simple and effective selection algorithm reduces memory and time requirements to affordable values with no added complexity.

Table 2 shows performance results using both PCA and random projection to project the source feature space down to 10000-500 dimensions. PCA outperformed the frequency-based feature selection method only for dimensions below 3000. Random projection led to worse (yet comparable) performance.

Though PCA outperformed the feature selection approach, both PCA computation (even when using the randomized algorithm) and projection are computationally expensive in high dimensional source spaces. Random projection can be done on the fly, but results were not satisfactory, and many other random

Table 2: EER and C_{LLR} on the NIST 2009 LRE closed-set 30s evaluation set, for various 4-gram SVM systems working on feature sets obtained with Principal Component Analysis and Random Projection.

	PCA		Random	
Size	**%EER**	$\mathbf{C_{LLR}}$	**%EER**	$\mathbf{C_{LLR}}$
10000	1.96	0.421	2.33	0.497
8000	1.89	0.414	2.35	0.502
6000	1.97	0.424	2.40	0.509
4000	2.00	0.424	2.53	0.532
3000	1.97	0.430	2.65	0.557
2000	2.02	0.430	2.83	0.597
1000	2.12	0.449	3.25	0.669
500	2.19	0.470	4.09	0.801

projection alternatives should be explored in the future. The cost of PCA can be drastically reduced by reducing the dimensionality of the source feature space, e.g. by applying feature selection. Table 3 shows performance when using the 11697 most frequent features as source space, and projecting it down to 4000-500 dimensions using both PCA and random projection. Both methods attained slightly better performance than that attained when starting from the original 100000-dimensional feature space.

Table 3: EER and C_{LLR} for various 4-gram SVM systems working on initial feature selection set of the 11697 most frequent features and posterior PCA and random projection.

	PCA		Random	
Size	**%EER**	$\mathbf{C_{LLR}}$	**%EER**	$\mathbf{C_{LLR}}$
4000	1,95	0,411	2,48	0,529
3000	1,95	0,416	2,65	0,553
2000	2,00	0,425	2,82	0,585
1000	2,03	0,435	3,16	0,658
500	2,15	0,457	3,77	0,761

5. Conclusions

In this work, different dimensionality reduction methods for high-dimensional (high-order n-gram) SVM-based Phonotactic Language Recognition have been studied. Results indicate that frequency-based feature selection is the most effective method (in terms of computational cost and performance) for dimensionality reduction. If extremely low dimensionality was desired, PCA should be used. In such a case, the computational cost of PCA estimation and projection can be reduced by first applying frequency-based feature selection on the source space. Random projections are an interesting and computationally simple alternative to PCA, but the results obtained in this work suggest that further research is needed to make an effective use and get performance gains from such projections or similar alternatives.

6. References

[1] P. Torres-Carrasquillo, E. Singer, T. Gleason, A. McCree, D. Reynolds, F. Richardson, and D. Sturim, "The MITLL NIST LRE 2009 language recognition system," in *Proc. of ICASSP 2010*, 2010, pp. 4994–4997.

[2] W. M. Campbell, J. P. Campbell, D. A. Reynolds, E. Singer, and P. A. Torres-Carrasquillo, "Support vector machines for speaker and language recognition," *Computer Speech and Language*, vol. 20, no. 2–3, pp. 210–229, 2006.

[3] N. Brümmer and D. van Leeuwen, "On calibration of language recognition scores," in *Proceedings of Odyssey - The Speaker and Language Recognition Workshop*, 2006, pp. 1–8.

[4] W. M. Campbell, F. Richardson, and D. A. Reynolds, "Language recognition with word lattices and support vector machines," in *ICASSP*, Honolulu, HI, 2007, pp. 15–20.

[5] F. Richardson and W. Campbell, "Language recognition with discriminative keyword selection," in *Proceedings of ICASSP*, 2008, pp. 4145–4148.

[6] R. Tong, B. Ma, H. Li, and E. S. Chng, "Selecting phonotactic features for language recognition," in *Proceedings of Interspeech*, September 2010, pp. 737–740.

[7] M. Penagarikano, A. Varona, L. Rodríguez-Fuentes, and G. Bordel, "A dynamic approach to the selection of high-order n-grams in phonotactic language recognition," in *Proceedings of ICASSP*, Prage, Czech Republic, 2011.

[8] T. Mikolov, O. Plchot, O. Glembek, P. Matejka, L. Burget, and J. Cernocky, "PCA-based feature extraction for phonotactic language recognition," in *Proc. Odyssey 2010 - The Speaker and Language Recognition Workshop*, 2010, pp. 251–255.

[9] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '01, New York, NY, USA, 2001, pp. 245–250.

[10] P. Schwarz, "Phoneme recognition based on long temporal context," Ph.D. dissertation, Faculty of Information Technology BUT, http://www.fit.vutbr.cz, Brno, CZ, 2008.

[11] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Lui, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. Woodland, *The HTK Book (for HTK Version 3.4)*, Cambridge, UK, 2006.

[12] A. Stolcke, "SRILM - an extensible language modeling toolkit," in *Proceedings of ICSLP*, November 2002, pp. 257–286.

[13] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008, software available at http://www.csie.ntu.edu.tw/∼cjlin/liblinear.

[14] A.Martin and C. Greenberg, "The 2009 nist language recognition evaluation," in *Proceedings of Odyssey 2010: The Speaker and Language Recognition Workshop*, Brno, Czech Republic, 2010, pp. 165–171.

[15] Z. Jancik, O. Plchot, N. Brummer, L. Burget, O. Glembek, V. Hubeika, M. Karafiat, P. Matejka, T. Mikolov, A. Strasheim, and J. Cernocky, "Data selection and calibration issues in automatic language recognition - investigation with but-agnitio nist lre 2009 system," in *Proc. Odyssey 2010 - The Speaker and Language Recognition Workshop*, 2010, pp. 215–221.

[16] N. Brümmer and J. A. du Preez, "Application-independent evaluation of speaker detection," *Computer Speech & Language*, vol. 20, no. 2–3, pp. 230–275, 2006.

[17] V. Rokhlin, A. Szlam, and M. Tygert, "A randomized algorithm for principal component analysis," *SIAM J. Matrix Anal. Appl.*, vol. 31, pp. 1100–1124, August 2009.

[18] W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," in *Conference in modern analysis and probability (New Haven, Conn., 1982)*, ser. Contemporary Mathematics. American Mathematical Society, 1984, vol. 26, pp. 189–206.

[19] D. Achlioptas, "Database-friendly random projections," in *Proceedings of the twentieth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, ser. PODS '01. New York, NY, USA: ACM, 2001, pp. 274–281.