Improved language recognition using mixture components statistics

3 authors, including:

Abualsoud Hanani
Birzeit University
9 PUBLICATIONS **61** CITATIONS

SEE PROFILE

Martin J Russell
University of Birmingham
125 PUBLICATIONS **1,360** CITATIONS

SEE PROFILE

Improved Language Recognition using Mixture Components Statistics

Abualsoud Hanani, Michael Carey and Martin Russell

Department of Electronic, Electrical and Computer Engineering,
University of Birmingham, UK
Aah648@bham.ac.uk, m.carey@bham.ac.uk, m.j.russell@bham.ac.uk

Abstract

One successful approach to language recognition is to focus on the most discriminative high level features of languages, such as phones and words. In this paper, we applied a similar approach to acoustic features using a single GMM-tokenizer followed by discriminatively trained language models. A feature selection technique based on the Support Vector Machine (SVM) is used to model higher order n-grams. Three different ways to build this tokenizer are explored and compared using discriminative uni-gram and generative GMM-UBM. A discriminative uni-gram using very large GMM tokenizer with 24,576 components yields an EER of 1.66%, rising to 0.71% when fused with other acoustic approaches, on the NIST'03 LRE 30s evaluation.

 Index Terms: Language Recognition, Support Vector Machine, Gaussian Mixture Model, Tokenization**.**

1.   Introduction

Automatic spoken language recognition is the process of classifying an unknown spoken utterance as one of a set of pre-encountered languages. A number of approaches to the language recognition problem are described in the literature. The most successful to-date are those based on the phonotactics and acoustic of the languages [1].

   The core of the acoustic systems is a Gaussian Mixture Model (GMM), which can be used with other techniques such as Support Vector Machines (SVM), Joint-Factor Analysis (JFA), and channel compensation.

   A typical phonotactic-based approach was described in the classic paper by Zissman [2]. It uses a parallel phone recognizer followed by a language model (PRLM) for each language. Multiple phone recognizers are used as a front-end to generate an estimate of the phone sequence of the utterance to be identified. The language models are used to estimate the probability of this phone sequence being generated by a speaker of a particular language. Although this approach performed well on the National Institute of Standards and Technology (NIST) language recognition evaluation (LRE), it has some disadvantages. Training of phone recognizers is computationally expensive and laborious, because it needs orthographically or phonetically transcribed training data. This can be difficult to obtain for some languages, especially minority languages.

   The crucial component of phone-based systems is the language model (LM) which captures language specific information from the phone/word sequence produced by the phone recognizer. The usual approach to building LMs is statistical n-gram modeling, which can be applied to sequences of different level of speech segments, e.g. phones and words.

In this paper we propose discriminative training of LMs using a SVM. This has been successfully used in acoustic-based language recognition systems, and is able to outperform the usual n-gram approach [3]. The SVM is trained on the n-gram statistics of phone/lattice sequences per utterance.

   In other approaches, such as in [4], the phone recognizer is replaced by a Gaussian Mixture Model (GMM) as a tokenizer. The GMM tokenizer processes each acoustic frame and generates a sequence of indices of N-best GMM components. The advantage of the GMM tokenizer is that it does not require phonetically transcribed data, and it can be trained on the same acoustic data that is used to train the acoustic-based language recognition system. This is computationally less intensive than the phone recognizer and fast-scoring techniques using the background model can be used.

   We use a single Universal Background Model (UBM) as a tokenizer for all languages. The language models are trained on N-gram statistics using SVM. To emphasise the discriminative components, a weighting technique is applied. In [5] the Inverse Document Frequency (IDF) used in Information Retrieval (IR), and the Log-Likelihood Ratio (LLR) weighting are applied at the phone and word levels. The LLR weighting technique outperformed IDF.

   We suggest that the most discriminative N-gram components are common in one language and rare in others. Increasing the order of GMM allows these features to occupy separate components. We achieve this with low computational cost and less training data by replacing the traditional UBM with a Multi-Language Model (MLM), which is a concatenation of multiple language-dependant GMMs.
 GMM-UBM and GMM-SVM systems are used as a baseline and to fuse their results with the proposed systems.

   The rest of this paper is organized as follow: Section 2 describes the baseline systems and the proposed systems with the weighting and feature selection techniques. Together with the Corpus and the evaluation criteria, the experimental systems are described in section 3. Our results are presented and discussed in section 4 with conclusions in section 5.

2.   System Description

2.1. GMM-UBM with Inter-Session Compensation (ISC)

Two gender-dependent UBMs are trained with the maximum likelihood EM-algorithm using utterances from all of the languages. A language dependent model is obtained by MAP adaptation (means only with relevance factor) of the UBM using the language specific enrollment conversations. The result is two UBM models and two language dependent models for each language. The inter-session variability within a language, such as inter-channel and inter-speaker variability,

is estimated using the technique described in[6]. Then, both the UBM means and the language dependent GMM means are shifted to the estimated nuisance direction of each testing utterance before scoring.

2.2. SVM on GMM Supervectors (GMM-SVM)

The SVM is a two-class discrimination technique which involves finding a hyperplane for effective separation of the two classes; target and background.

In our GMM-SVM system, each utterance is used to estimate the parameters of a GMM by MAP adaptation of the UBM. The GMM mean vectors are concatenated into one 'supervector'. Hence each speech utterance is mapped from the cepstral feature vector sequence domain to the supervector domain (very high dimensional space), where the languages are assumed to be more linearly separable. This process also normalizes the length of the utterances. The supervectors are used to train the SVM.

Since language recognition is a multi-class problem, a 'one-against-the-rest' strategy is used. The language specific supervectors are used as the 'targets' and supervectors of the remaining languages as 'background', giving one SVM model for each language. The test supervectors are scored against the SVM models. Positive scores indicate that the test utterance belongs to the target language, and negative scores that the utterance belongs to one of the non-target languages. Thus, the output scores of the SVM can be interpreted as log-likelihood scores and log-likelihood normalization (LLR) or 'max' log-likelihood normalization can be used.

The linear KL-divergence approximated kernel, which was used in [7] is used for this system.

Campbell and Karam [8] illustrated the connection between SVM scoring and GMM scoring, and showed that the latter yielded better results. Weighted averages of the target and background support vectors are 'pushed back' to the GMM domain and used in GMM scoring. Our system differs from that described in [8] in two ways: First, only GMM means are adapted to form the supervectors, because this outperforms adapting the means and covariances, and inter-session compensation is applied to both target and non-target 'pushed' GMM models, as described for the GMM-UBM system in 2.1.

2.3. GMM Tokenization with Discriminative Language Modeling

2.3.1. Language Modeling with SVM

The successful application of the SVM to GMM supervectors has encouraged workers to apply it to language modeling (e.g. [3]). The N-gram components of the sequence of tokens generated from an utterance U can be represented as a D-dimensional vector p where, D is the number of all N-grams (in our case GMM components), C_j is the jth N-gram and the probability p_j of C_j is estimated using counts of N-grams,

$$p_j = \frac{Count(C_j)}{\sum_i Count(C_i)} \qquad (1)$$

where the sum in (1) is performed over all N-grams and $Count(C_j)$ is the number of times the N-gram C_j occurs in the produced sequence of tokens.

Assuming, p^{tar} and p^{bkg} are probability vectors of the target and background languages respectively, the usual N-gram classifier will have a hyperplane h which separates the target and the background vectors, where

$$h_i = log(p^{tar}_i/p^{bkg}_i) \qquad (2)$$

The N-gram classifier can be represented as a linear classifier where the SVM can be applied to find a better separating hyperplane h by using different kinds of kernel functions. The most commonly used SVM kernels are the Gaussian and the polynomial. The simple linear dot-product kernel is used in this system because other kernels gave no improvement.

2.3.2. Weighting

It has been shown that using a SVM for language modeling outperforms the traditional N-gram approach [3]. Before we apply the SVM, the probabilities of the N-grams are estimated for each utterance rather than for each language as in the usual N-gram method. Then, these probabilities are weighted to emphasize the most discriminative components (i.e. those which occur frequently in one language and infrequently in others). The N-gram components which are common in most languages, such as silence or common phones, contain little discriminative information and are de-emphasized. Numerous weighting techniques are available for this purpose, such as the Inverse Document Frequency (IDF) from Information Retrieval (IR), 'Usefulness' from Topic Spotting and Identification, and the Log-Likelihood Ratio (LLR) weighting technique proposed in [5]. The author in [5] applied IDF and LLR weighting to sequence of high level features, such as phones, lattices and words. The LLR weighting worked best. The LLR weighting w_j for component C_j is given by:

$$w_j = g_j\left(\frac{1}{p(C_j/all)}\right) \qquad (3)$$

where g_j is a function used to smooth and compress the dynamic range (for example, $g_j(x)=\sqrt{x}$, or $g_j(x)=log(x)+1$). $p(C_j/all)$ is the probability of N-gram component C_j across all languages. The components which have zero occupancy in all languages are removed since they do not carry any useful information. A benefit of discarding these non-visited components is that it reduces the feature dimension dramatically, particularly for the high order N-gram system as the dimension of the N-gram increases exponentially (M^n) with GMM model order (M).

Those N-gram components which have a very small probability have a very high weighting, allowing a minority of components to dominate the scores. To prevent this, a minimum threshold T_1 on the weighting w_j was applied.

According to Zipf's law, the rank-frequency distribution of words in a typical document follows a decaying exponential. The high ranking words with high probability are not useful for discrimination because they appear in most of the documents. Conversely, the low-rank words are too rare to gather useful statistical information. The area of interest is somewhere in the middle. This motivates us to apply a second, maximum, threshold T_2 on the weighting vector to de-emphasise the common components. The values of T_1 and T_2 were determined empirically on the development data set.

2.3.3. Feature Selection

In addition to the weighting and thresholds described in section 2.3.2, a feature selection technique is needed to minimize the number of N-gram components by keeping only those which are most discriminative. This is particularly necessary in high order N-gram systems because the dimension is increased exponentially. Consequently, reducing the number of N-gram components decreases the computational cost and the required amount of memory.

A powerful iterative feature selection algorithm based on the SVM is proposed by Guyon, et.al [9]. This is applied to phone-based language recognition with discriminative keyword selection in [10], where more details can be found.

A similar algorithm is used on the bi-gram data in our work.

2.3.4. Multi-Language Model (MLM)

We hypothesize that increasing the number of UBM components will cause language-specific information to be represented in separate components. In *N*-gram systems these components contain the most discriminative information. This is the motivation for our high order 'Multi-Language Model' (MLM). For a conventional EM-trained UBM, increasing the model order necessitates more training data and computation. These problems are alleviated in a MLM.

The MLM is a concatenation of language-dependent GMMs, each trained separately on language specific training data using EM. The resulting GMMs are combined to form a single large MLM. Pairs of MLM components for which the KL-divergence distance [7] is below some threshold are combined into a single component. The MLM gives more space to represent language specific information. Each language GMM can be of different order, depending on the available enrolment data. Training a MLM also requires less computational than training a comparable UBM.

If $\lambda_l = \{\omega_l, \mu_l, \Sigma_l\}$ are the parameters of the GMM for language *l*, the probability density function of the GMM is:

$$P(o_t \mid \lambda_l) = \sum_{\forall i} \omega_l^i N(o_t; \mu_l^i; \Sigma_l^i) \tag{4}$$

where, ω^i_l, μ^i_l and Σ^i_l are weight, mean and diagonal covariance of the ith of the GMM for language *l*, respectively. o_t is the observation vector at time *t*. To form the MLM, the parameters of the language-dependent GMMs are simply concatenated together to form one large model.

$$\lambda_{MLM} = \left\{ \frac{[\omega_1, \dots \omega_L]}{L}, [\mu_1; \dots \mu_L], [\Sigma_1; \dots \Sigma_L] \right\} \tag{5}$$

where, λ_{MLM} is the set of MLM model parameters and *L* is the number of language-dependent models. To date we have built gender dependent MLMs of order up to 24,576.

2.3.5. Fusion and Calibration

For fusion, the output scores of four systems are stacked in feature vectors of dimension $N_s \times N_l$ (number of systems times number of language-dependent models). The systems are the GMM-UBM system from 2.1, SVM with GMM supervectors (GMM-SVM) from section 2.2, and the SVM uni-gram and bi-gram systems described in section 2.3. The output feature vectors are then fused (and calibrated) using multi-class linear logistic regression. The fusion parameters were trained on the NIST 1996 evaluation set "lid96e1". Brummer's *Focal multi-class toolkit[i]* was used to estimate the fusion parameter.

3. Corpus and evaluation criteria

3.1. Corpus

The training data used in all of our experiments consists of the train set and the development set of the Callfriend corpus [11]. Each set contains twenty half-hour two-sided telephone conversations for each of the fifteen languages and dialects. The twelve languages are; English, Arabic, Farsi, French, German, Hindi, Japanese, Korean, Mandarin, Spanish, Tamil and Vietnamese. English, Spanish and Mandarin have two dialects. The NIST 1996 evaluation set (lid96e1) is used to train the back-end for calibration and fusion. The NIST 2003 evaluation set (lid03e1) is only used for evaluation. This set contains test utterances of 30s, 10s and 3s segments, but only 30s segments are used in our evaluation. This evaluation sub-set consists of 1280 utterances; 80 for each language come from the Callfriend corpus, 160 for English and 80 for Japanese come from Callhome corpus. Since we focus on closed-set language recognition, Russian utterances were excluded from the evaluation.

3.2. Evaluation Criteria

We based our experiments on the NIST 2003 Language Recognition Evaluation (LRE) closed-set task. The Equal Error Rate (EER) percentage was used to measure the recognition performance. EER is obtained by pooling the scores of all language dependent models. This criterion is biased to the models which have more test data, because it ignores the priors of the language models. To overcome this issue, the average EER percentage is used to measure the performance. In the 2007 and 2009 LRE plans, NIST required a pair-wise language evaluation for all target/non-target language pairs, with the average cost performance C_{avg} to describe the performance of the over-all system. In this paper, 'pooled' EER percentage, average EER, and average cost C_{avg} are used to measure the performance.

3.3. System

In all of the experiments in this paper, acoustic feature vectors were based on nineteen cepstral coefficients derived from the power output of nineteen quadrature pairs of linear phase FIR filters. The Mel Frequency Cepstral Coefficients (MFCC), including C_0, were concatenated with normalized Shifted-Delta Cepstra (SDC) coefficients with a 7-3-3-7 configuration [12], giving a total of 68 features per frame at a frame rate of 100 frame/sec. RASTA filtration is applied to the power spectra, and feature normalization (mean & variance) is applied to the final feature vectors to reduce the channel effect.

Two gender-dependent UBM models, each with 4096 mixture components, were trained using all of the training data with 4 EM iterations updating all parameters; means, diagonal covariances and mixture weights. 24 Language dependent GMMs were MAP-adapted from the UBMs using language specific data (system 'GMM-UBM' in table1).

The UBM means were also MAP adapted using each single-side conversation of each language, generating the GMM supervectors which were used to train the SVM as described in section 2.2 (system 'GMM-SVM' in table 1) and to estimate the eigenchannel matrix *U* which was used for the inter-session compensation.

Two gender-dependent background models were used as GMM-tokenizer for the *N*-gram system described in 2.3: an MLM with 24,576 components for the uni-gram system (system 'uni-gram' in table 1), and a UBM with 4096 components for the bi-gram system (system 'bi-gram' in table 1). Language models were trained on the whole training data with LLR weighting using the SVM-KM MATLAB toolbox[ii]. The 'fused' result was obtained by fusing the outputs of the four systems.

To study the difference between the traditional UBM and the MLMs, a 6144 component background model is built in three ways: A traditional UBM model (table 2, column 2), a concatenation of 12 language dependent 512-component GMMs built separately for each language (table 2, column 2), and a concatenation of 12 language dependent 512-component

GMMs MAP adapted from a 512-component UBM model (table 2, column 4). Each background model was used in two different systems: A GMM-UBM system, and a discriminative uni-gram system (rows 2 and 3 in table 2, respectively).

The experiments were only practically feasible because computations were accelerated by an Nvidia Geforce GTX 260 graphics processing unit (GPU), comprising 216 floating-point processors and 1.76GB of RAM together with an Nvidia Tesla processor of similar performance. Programming was carried out in *MATLAB, GPUmat* and *CUDA*.

4. Results and Discussion

The performances of the first four systems described above are shown in table 1. All the systems used a gender-dependent UBM with 4096 components except the uni-gram system which used an MLM with 24,576 components. By fusing the four systems together, the performance improves to 0.71% EER and 0.0098 C_{avg}. This suggests that the systems are complimentary, focusing on different parts of the acoustic space of the languages.

	'Pooled' EER	Avg EER	C_{avg} *100
GMM-UBM	3.4	3.65	4.11
GMM-SVM	0.82	0.92	3.7
Uni-gram	1.66	2.02	3.02
Bi-gram	3.51	3.96	6.95
fused	0.71	0.83	0.98

Table 1: *Performance of the four systems on the NIST 2003 LRE 30s closed-set for 12 languages.*

Table 2 shows the performance of two different systems; traditional GMM-UBM and SVM-trained Uni-gram. These two systems used the same size of background model, but build using three different methods as described in 3.3. Each of the three models has of 6144 components (512*12).

System/Background	UBM	MLM	MLM-Adapt
UBM-GMM	5.36	5.77	6.56
Uni-gram	2.7	2.38	3.63

Table 2: *Performance [EER%] of GMM-UBM and uni-gram systems using background model built in three different ways.*

It is clear from the results that the MLM background model is advantageous for the uni-gram system but not for the probabilistic GMM-UBM system. A possible explanation is that the areas of interest of the two systems in acoustic space are different. The discriminative *N*-gram systems focus on the language-specific 'boundaries' of the background model, where use of a component is indicative of a particular language. By contrast, the probabilistic GMM-UBM system relies on differences in the probabilities from components of the language specific GMMs which arise from MAP adaptation of the same components from the cross-language 'middle' of the background model. The smaller traditional UBM appears to result in more reliable and robust Gaussian probabilities, but has fewer language-specific components that can be exploited by the unigram model, whereas the larger MLM method has enough space to accommodate the language specific components that the unigram model requires. Thus, the MLM is more biased towards the language specific components than the traditional UBM. This is useful for the discriminative approaches but not for the generative approaches.

5. Conclusion

It has been shown that methods normally applied to sequences of high-level units such as phones or words can be successfully applied to sequences of GMM components. A unigram system works surprisingly well, provided that discriminative weighting is applied to the uni-gram probabilities. The Multiple Language Model (MLM) has been proposed as an alternative to a conventional UBM. The MLM appears to have more language-specific components than a UBM, and for this reason works particularly well as the basis of a uni-gram system (and potentially as the basis of an n-gram system), but less well in a conventional probabilistic GMM-UBM system. The best performance is obtained by fusing the outputs of a conventional 4098 component GMM-UBM-SVM system with those of a discriminatively weighted uni-gram system based on a 24,576 components MLM. This results in an EER of 0.71% on the NIST03 LRE 30s test set.

6. References

1. P. Torres-Carrasquillo, E. Singer, W.M. Campbell, T. Gleason, A. McCree, D.A. Reynolds, F. Richardson, W. Shen, D. Sturim, , *"The MITLL NIST LRE 2007 Language Recognition System"*, in *InterSpeech'08*. 2008.
2. M. A. Zissman, *"Comparison of four approaches to automatic language identification of telephone speech".* IEEE Trans. of Sp and Audio Proc., 1996. **4**(1): p. 31-44.
3. L. Zhai, M.S., X. Yang and H. Gish,, *"Discriminatively trained Language Models Using Support Vector Machines for Language Identification,".* Odyssey 2006, 2006.
4. X. Yang, M.S., *"N-Best Tokenization in a GMM-SVM Language Identification System".* ICASSP'07, 2007. **4**: p. 1005-1008.
5. W.M.Campbell, J.P.Campbell, T. P. Gleason, D.A. Reynolds, Wade Shen, *"Speaker Verification Using Support Vector Machines and High-Level Features,"* Audio, Speech, and Language Processing, IEEE Transactions on,, 2007. **15**(7): p. 2085-2094.
6. C. Vair, D. Colibro, F. Castaldo, E. Dalmasso, P.Laface. *"Channel Factors Compensation in Model and Feature Domain for Speaker Recognition".* in *Speaker and Language Recognition Workshop, IEEE Odyssey 06*. 2006.
7. R. Dehak, N. Dehak,P. Kenny,P. Dumouchel, *"Linear and non linear kernel GMM supervector machines for speaker verification".* INTERSPEECH-07, 2007: p. 302-305.
8. W.M. Campbell, Z. N. Karam, *"A Framework for Discriminative SVM/GMM Systems for Language Recognition".* INTERSPEECH-2009, 2009: p. 2195-2198.
9. I. Guyon, J.W., S. Barnhill, and V. Vapnik, *"Gene selection for cancer classificationusign support vector machines,".* Machine Learning,, 2002. **46**(1-3): p. 389-422.
10. W. M. Campbell, F. S. Richardson,, *"Language recognition with discriminative keyword selection,"* in *ICASSP'08*. 2008: Las Vegas, NV p. 4145-4148.
11. *CallFriend Corpus: http://www.ldc.upenn.edu/Catalog*.
12. P. A. Torres-Carrasquillo, E.S., M.A. Kohler, R.J. Greene, D.A Reynolds, and J.R. Deller,, *"Approaches to language identification using Gaussian mixture models and shifted delta cepstral features".* Proc. ICSLP 02, 2002: p. 89-92.

[i] *http://niko.brummer.googlepages.com/focalmulticlass*
[ii] *http://asi.insa-rouen.fr/enseignants/~arakotom/toolbox*