Dialect Recognition Using a Phone-GMM-Supervector-Based SVM Kernel

Fadi Biadsy, Julia Hirschberg*, Michael Collins†*

*Department of Computer Science, Columbia University, New York, NY, USA

{fadi,julia}@cs.columbia.edu

† MIT Computer Science and Artificial Intelligence Laboratory, Cambridge, MA, USA

mcollins@csail.mit.edu

Abstract

In this paper, we introduce a new approach to dialect recognition which relies on the hypothesis that certain phones are realized differently across dialects. Given a speaker's utterance, we first obtain the most likely phone sequence using a phone recognizer. We then extract GMM Supervectors for each phone instance. Using these vectors, we design a kernel function that computes the similarities of phones between pairs of utterances. We employ this kernel to train SVM classifiers that estimate posterior probabilities, used during recognition. Testing our approach on four Arabic dialects from 30s cuts, we compare our performance to five approaches: PRLM; GMM-UBM; our own improved version of GMM-UBM which employs fMLLR adaptation; our recent discriminative phonotactic approach; and a state-of-the-art system: SDC-based GMM-UBM discriminatively trained. Our kernel-based technique outperforms all these previous approaches; the overall EER of our system is 4.9%.

1. Introduction

In recent years, there has been increasing interest in the speech science and technology communities in automatically identifying the regional dialect and accent of a speaker from a sample of his/her speech. The dialect recognition problem has been considered to be more difficult than language recognition since dialects of the same language are assumed to be somewhat similar. Although they may differ in morphology, lexicon, syntax, phonetics and phonology, these differences are likely to be more subtle across dialects than across languages.

There are many applications for dialect recognition. Arabic speakers with different dialects, for example, pronounce some words differently and consistently alter certain phones and morphemes. These differences negatively impact Arabic Automatic Speech Recognition (ASR) systems. Identifying the dialect prior to ASR will enable the system to adapt its pronunciation, acoustic, and language models appropriately. Dialect recognition is also useful for identifying a speaker's regional origin and ethnicity and helpful in forensic speaker profiling.

Phonotactic-based approaches, such as Phone Recognition followed by Language Modeling (PRLM), have been shown to be effective in identifying languages and dialects [1, 2]. Gaussian Mixture Models - Universal Background Model (GMM-UBM) with Shifted Delta Cepstral (SDC) has also achieved considerable success in speaker and language/dialect recognition ([3, 4]). Discriminative training has proven quite useful in recent language recognition systems (e.g., [5, 6]). Torres-Carrasquillo et al. [7], for example, showed that a GMM-UBM-based model discriminatively trained with SDC features with an eigen-channel compensation component and vocal-tract normalization (VTLN) produces good results for the recognition of American vs. Indian English, four Chinese dialects, and three Arabic dialects (Gulf, Iraqi, and Levantine). In addition to phonotactic and acoustic-based systems, prosodic features have also been found useful for dialect recognition (e.g., [8]).

In this paper, we test the hypothesis that certain phones are realized differently across dialects. We present results of a new approach to dialect recognition using SVM classifiers to distinguish between pairs of dialects using a kernel that computes phonetic similarity. We test our approach on four Arabic dialects and compare our results to multiple systems. We describe the corpora used in our experiments in Section 2. In Section 3, we describe the front-end and phone recognizer we have built for our approach. We describe our phone-GMM-Supervector-based SVM kernel approach to dialect recognition in Section 4 and discuss experimental results in Section 5. Finally, in Section 6, we conclude and describe our future work.

2. Corpora

We test our approach on spontaneous telephone conversations produced by native speakers of the following broad Arabic dialects and provided by the Linguistic Data Consortium: Iraqi, Gulf, Levantine, and Egyptian Arabic. We use Appen's corpora for the first three (478 Iraqi, 976 Gulf, and 985 Levantine Arabic speakers) [9], holding out 20% of speakers from each for testing. Each of the corpora contains male and female speakers speaking by landline or mobile phones. Since it is likely that the distribution of these categories may influence the trained models, we equalized the number of test speakers in each category. So, our test set for each dialect includes: 25% selected randomly from female speakers speaking on mobile phones; 25% from males speaking on mobile phones; 25% from females speaking on landline phones; and 25% from males speaking over landlines.

For the Egyptian corpus, we use the 280 speakers in CallHome Egyptian and its supplement for training. To test our system under different acoustic conditions, we employ 120 speakers from CallFriend Egyptian for testing. The Egyptian data also includes male and female speakers, but it is not clear if the speakers used landlines, mobile phones, or both.

To identify speech regions in the audio files, we segment the files based on silence using Praat [10] using a silence threshold of -35db with a minimum silence interval of 0.5s and minimum sounding intervals of 0.5s. All segments are used in training. In this paper, we present results from testing our system on 30-second cuts. Each cut consists of consecutive speech segments totaling 30s in length.[1] Multiple cuts are extracted from each speaker. For Iraqi, we have 477 30s test cuts, and 801, 818, 1912 30s test cuts for Gulf, Levantine, and Egyptian, respectively.

[1] Sometimes we had to truncate speaker turns to achieve exactly 30s.

3. Context-Dependent Phone Recognizer

The dialect recognition approach we propose here makes use of phone hypotheses. We therefore build a continuous HMM-based triphone CD phone recognizer using IBM's Attila system [11]. The phone recognizer is trained on Modern Standard Arabic (MSA) using 50 hours of GALE speech data of broadcast news and conversations and consists of 230 CD acoustic models and a total of 20,000 Gaussians. We use one acoustic model for silence, one for non-vocal noise and another to model vocal noise. The front-end is a 13-dimensional Perceptual Linear Prediction (PLP) front-end with cepstral mean and variance normalization (CMVN). Each frame is spliced together with four preceding and four succeeding frames and then Linear Discriminant Analysis (LDA) is performed to yield 40D feature vectors. We use the LDA matrix derived for IBM's Attila Arabic ASR system here [11]. We utilize a unigram phone model trained on MSA to avoid bias for any particular dialect. The pronunciation dictionary used in this work is generated as in [12]. Our phone inventory includes 34 phones, 6 vowels and 28 consonants.

The phone-recognizer is a two-pass system. In the first pass, we obtain the most likely phone sequence hypothesis. The second pass uses this hypothesis to perform model adaptation, followed by decoding. We first apply feature space Maximum Likelihood Linear Regression (fMLLR) followed by MLLR adaptation, given the most likely phone sequence hypothesis.

4. Dialect Recognition Approach

The first stage in our dialect recognition process, after front-end pre-processing, is to use our phone recognizer to obtain the most likely phone sequence hypothesis for each utterance in the training corpora. We then extract the PLP feature vectors for each frame of each phone instance in the sequence. Note that these features are extracted after normalization (CMVN) and fMLLR transformation. We next train a GMM-UBM for each phone type using all frames of all instances of that phone type across all dialects. We denote this GMM-UBM as *phone GMM-UBM*. In this paper, all GMMs are ML (Maximum-Likelihood) trained, with 100 Gaussian components, using the EM algorithm.[2] To avoid a bias for any particular dialect in the GMM-UBM, we select an equal number of frames from each dialect for each phone GMM-UBM. From our 34 MSA phone inventory we thus generate 34 phone GMM-UBMs.

4.1. Creating Phone-GMM-Supervectors

To model acoustic-phonetic differences at the phone level, we need to extract a vector that captures these differences for each phone in the hypothesized phone sequence. To do this, we adopt the GMM-Supervector approach [13] — but at the level of phone instances, similar to our previous work [14]. We use the acoustic frames of each phone instance to MAP (Maximum A-Posteriori) adapt the corresponding phone GMM-UBM. We adapt only the means of the Gaussians using a relevance factor of $r = 0.1$. We denote the resulting GMM as the *adapted-phone GMM*. The intuition is that some of the means 'summarize' the spectral features of a phone instance. We represent each phone instance in a sequence by a *Supervector* which is the result of stacking all the mean vectors of the Gaussians of the adapted-phone GMM. We have previously observed that the duration of vowels and certain consonants significantly differ across Arabic dialects. So, we also include phone duration as an additional feature in the Supervectors [8]. The duration feature is computed as the log of the number of frames in the phone.

We represent each utterance U as a sequence S_U of tuples $(\vec{v}_i, \phi_i)$, such that $\vec{v}_i$ is the Supervector of the i^{th} phone in the sequence and ϕ_i is the identity of that phone. Thus, an utterance U is represented as a sequence of tuples $S_U = \{(\vec{v}_1, \phi_1), (\vec{v}_2, \phi_2), ..., (\vec{v}_n, \phi_n)\}$, where n is the number of phones in U. It should be noted that our representation retains the dependency between the phone identity and the Supervector which 'summarizes' the spectral characteristics of the phone. More importantly, the Supervector representation retains *some* of the dependency across the spectral features obtained from the entire phone segment (albeit without frame order).

4.2. Designing a Phone-Based SVM Kernel

From the sequences of tuples S_U produced for the utterances U of the training corpora, we next train an SVM classifier for each pair of dialects to distinguish one dialect from another. We design a kernel function to compute the similarity between pairs of utterances U_a and U_b. Let $S_{U_a} = \{(\vec{v}_i, \phi_i)\}_{i=1}^{n}$ and $S_{U_b} = \{(\vec{u}_j, \psi_j)\}_{j=1}^{m}$ be the tuple sequences of U_a and U_b, respectively. Our kernel function is defined in (1).

$$K(S_{U_a}, S_{U_b}) = \sum_{i,j:\phi_i=\psi_j} e^{-\|\vec{v}_i - \vec{u}_j\|^2/2\sigma^2} \quad (1)$$

This function computes the sum of RBF kernels between every pair of Supervectors of phone instances with the same type (i.e. the same MSA phone) across the two utterances. It is straightforward to show that this kernel is positive definite, satisfying the Mercer condition. Note that this kernel ignores the order of Supervectors in the sequence. As a result, phonotactic features, for example, are not captured. Further research will be required to incorporate the sequential aspect in the kernel.

4.3. SVM Classification

Recall that our goal is to test our system on 30s cuts; however, our training files are substantially longer. We therefore divide all training files into segments of approximately 30s each (after removing silence). Employing the kernel function above, we first compute a kernel matrix for each pair of dialects using the tuple sequences extracted for each of these 30s segments. Next we train a standard binary SVM classifier for each pair of dialects using the pair's kernel matrix.[3] The regularization parameter C and σ (in the kernel function 1) are selected by 10-fold cross-validation on the training data. Since our goal in this paper is to recognize four Arabic dialects, we train a total of six binary classifiers.

$$f(S_U) = \sum_{i=1}^{N} \alpha_i y_i K(S_U, x_i) + b \quad (2)$$

During testing, we first run the phone recognizer to obtain the most likely phone sequence hypothesis for U along with the frame alignment for each phone instance. We next extract the Supervector for each phone instance in the sequence, as described above, to obtain S_U. Using our kernel function, we then compute the kernel values $K(S_U, x_i)$, for all N support vectors x_i ($1 \leq i \leq N$). The final class prediction is then the sign of $f(S_U)$ in expression (2), where α_i and b are the estimated

[2] In future, we plan to test the sensitivity of our approach to the choice of number of Gaussians and to experiment with data-driven methods to obtain the number of Gaussians for each phone type.

[3] In our implementation, we use LibSVM toolkit [15] to train our SVM models.

parameters of the dialect-pair SVM model (after training) and $y_i \in \{-1, 1\}$, the class label of support vector i.

5. Dialect Recognition Experiments

We evaluate our kernel approach on the task of Arabic dialect recognition. We compare it to five approaches: a standard PRLM; a standard GMM-UBM; our own improved version of GMM-UBM which employs fMLLR adaptation [14]; our recent discriminative phonotactic approach [14]; and a SDC-based GMM-UBM discriminatively trained [7]. We adopt the NIST language/dialect and speaker recognition evaluation framework to report detection results instead of identification. In the detection task, we are given a hypothesis and a set of test trials. We are asked to give a decision for each test trial to accept or reject the hypothesis, along with a confidence score. Using these scores, we report our results using Detection Error Tradeoff (DET) figures, which plot false alarms versus miss probabilities, and Equal Error Rate (EER), the error rate when both false alarm and miss probabilities are equal. To plot an overall DET, our results are pooled across each pair of dialects with dialect priors equalized to discount the impact of different number of test trials in each dialect.[4]

5.1. Scoring for PRLM and GMM-UBM

To score the test trials from the PRLM and both our GMM-UBM (with and without fMLLR adaptation) approaches, we employ the following scoring procedure, similar to [6, 14]. We denote the feature vector extracted for a given test trial r, as $\mathcal{O}_r$. Every test trial is given a confidence score of belonging to target dialect D_t. Since we do pairwise detection, for score computation we can make use of the knowledge that an utterance belongs to either the target or non-target dialect (D_{nt}). Assuming that the dialect priors are equal, the posterior probability of $\mathcal{O}_r$ can be reduced to the expression in (3). We use these posterior probabilities to represent our scores for these approaches. $p(\mathcal{O}_r|\lambda_{D_x})$ represents the likelihood of $\mathcal{O}_r$ given the model λ_{D_x} of dialect D_x, and τ_r normalizes duration differences across trials.

$$P(D_t|\mathcal{O}_r) = \frac{p(\mathcal{O}_r|\lambda_{D_t})^{\tau_r}}{p(\mathcal{O}_r|\lambda_{D_t})^{\tau_r} + p(\mathcal{O}_r|\lambda_{D_{nt}})^{\tau_r}} \qquad (3)$$

5.2. PRLM and GMM-UBM Approaches

We have previously shown that the standard PRLM approach is effective in identifying Arabic dialects [2]. Training a phonotactic trigram model for each dialect, the overall EER obtained by pooling the six pairs of dialects is 17.3%, as shown in Figure 1 (see [14] for the details of this approach).

Since our kernel approach relies upon acoustic features, we believe that it is also essential to compare the performance of our approach to an approach that utilizes similar features. For GMM-UBM, we use the same front-end described in Section 3 to extract the 40D PLP features, followed by CMVN. We use an equal number of training frames from three dialects (Iraqi, Gulf, and Levantine) to ML train the UBM with 2048 Gaussian components. Although it has been shown that broader temporal (e.g., SDC) features typically outperform the standard cepstral features [16], we use the same front-end used in our kernel approach to allow for a simple comparison. Moreover, our features are extracted from a relatively wide context; recall that our 40D PLP features span 9 frames, with dimensionality reduction performed using LDA.

[4]We use the NIST scoring software developed for LRE07: www.itl.nist.gov/iad/mig/tests/lre/2007

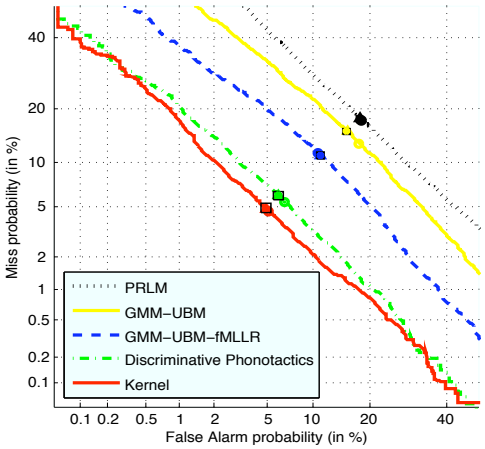

Figure 1: Overall DET curves for all approaches

A GMM (λ_{D_x}) is created for each dialect (D_x) by MAP-adapting only the means of the UBM using the training corpus for that dialect. We run the MAP adaptation in 5 iterations with a relevance factor of $r = 16$, similar to [7]. We do not employ fast scoring. During testing, we calculate the scores as in (3), where $\mathcal{O}_r$ represents the sequence of 40D PLP features of trial r, and $p(\mathcal{O}_r|\lambda_{D_x})$ represents the likelihood of $\mathcal{O}_r$ given GMM λ_{D_x} of dialect D_x, and τ_r is the inverse of the number of frames in the sequence $\mathcal{O}_r$. Similar to the PRLM approach, we use the test data of the four dialects (in Section 2) to test the performance of the GMM-UBM approach. The GMM-UBM approach achieves an EER of 15.3% and significantly outperforms the PRLM approach, as shown in Figure 1.

5.3. GMM-UBM with fMLLR Adaptation

It has been shown that the GMM-UBM approach can be improved by applying some normalization/transformation techniques for the acoustic signal. For example, VTLN to remove speaker-dependent features improves language and dialect recognition results [7, 17]. In addition, channel compensation techniques to retain only language-dependent information have been shown to significantly improve performance (e.g., [7]). In this paper, as in our recent work [14], we apply the fMLLR adaptation technique to transform the feature space given the phone hypotheses. Specifically, we first run the *context-dependent* (CD) phone recognizer to obtain the most likely CD-phone sequences.[5] Next, we use the *CD*-phone sequences to transform the acoustic data. Finally, we use the transformed frames as new features in our GMM-UBM approach. Applying the same settings of the GMM-UBM experiment above, but with fMLLR adaptation, we interestingly obtain a significant improvement: EER of 11.0% (see Figure 1).

5.4. Kernel-based Approach

We use the SVM classifier described in Section 4.3 for each pair of dialects to identify each 30s utterance (U) to one of the dialects. To be able to plot a DET curve, we need confidence scores. We employ Wu et al. [18]'s technique, implemented in LibSVM, which allows us to train SVM models that estimate

[5]Recall that our phone recognizer employs fMLLR.

posterior probabilities. Again on the hypothesis that each trial is either a target dialect, D_t or non-target D_{nt}, we use the posterior probability provided by the corresponding SVM model ($\Theta_{D_t D_{nt}}$) to represent our trial score: $p(D_t|S_U; \Theta_{D_t D_{nt}})$. Using the same training/testing cuts as our previously described approaches above, the overall EER obtained by pooling the six pairs of dialects is 4.9%, as shown in Figure 1.

We also compare our system to our recent *discriminative phonotactics* approach which relies upon CD-phonetic and phonotactic differences across dialects [14]. Briefly, in this approach, we classify CD-phones across dialects using SVM classifiers. We use the output of these classifiers to augment phonotactic features, which are then given to a logistic classifier to obtain detection scores. Like our kernel-based approach, we rely on the hypothesis that dialects differ in their phonetic realizations. The advantage of this approach is its ability to automatically identify important linguistic knowledge – the subtle differences that distinguish between dialects. Using the same training/testing splits as the current work, the discriminative phonotactic approach achieves an EER of 6.0%, better than every approach *except* our kernel method, which is significantly better (Figure 1). Note also that the kernel-based approach is simpler to implement and faster to train and test. It has the advantage that we need not train a classifier for every CD-phone. Instead, we combine the phonetic differences using a single kernel function, giving us one classifier for each pair of dialects.

5.5. Discriminatively-Trained GMM-based Approach

In a state-of-the-art system, Torres-Carrasquillo et al. [7] showed that a GMM-UBM-based model discriminatively trained with SDC features with an eigen-channel compensation component and VTLN and with a back-end classifier achieves an EER of 7.0% on three Arabic dialects (Gulf, Iraqi, and Levantine) using the same Appen corpora employed here. To compare our performance to this work, we conducted experiments with our kernel-based method, using both the training *and* the development data used by [7] to train our SVM models; we tested on the test cuts used in [7].[6] Using this data segmentation, our approach achieves a slightly better result than [7]: an EER of 6.4%. Note that we cannot be sure whether this represents a significant improvement over [7], since we lack sufficient information about their performance for each dialect. Nonetheless, our results suggest that the kernel-based approach has considerable potential, particularly when VTLN and channel compensation components are added. Note that we achieve higher EER on these three dialects than our overall EER for the four dialects due to the fact that Egyptian Arabic is the most distinguishable dialect of the four (see [8, 14]).

6. Conclusions and Future Work

In this work, we introduce a novel approach for dialect recognition, based on the notion that some phones are realized quite differently across dialects. Given an input utterance, we employ a phone recognizer to obtain the most likely phone sequence. We extract GMM Supervectors for each phone instance in the sequence. Using these Supervectors together with phone identity, we employ a novel kernel function that computes similarities between like phones across pairs of utterances. With this kernel we train an SVM classifier for each pair of dialects. We perform dialect recognition by classifying test utterances using these binary classifiers. We have conducted a series of experiments to test our approach on four Arabic dialects of spontaneous telephone conversations and to compare our results to previous approaches. On 30s utterances, we significantly outperform the following previous approaches: PRLM, GMM-UBM, GMM-UBM-fMLLR, and our own recent approach, discriminative phonotactics. The overall EER of our system is 4.9% on four Arabic dialects. Our kernel-based approach also performs slightly better than a state-of-the-art approach in dialect recognition (SDC-based GMM-UBM discriminatively trained) with VTLN and channel compensation components.

In future, we will compare the performance of our system on 3s and 10s utterances to previous results. As mentioned above, VTLN and channel compensation techniques have been shown to improve language and dialect recognition systems; we will test the impact of such techniques on our approach. Finally, we will test our approach on other dialects and accented languages as well as on Arabic sub-dialects.

[6] We thank P. Torres-Carrasquillo and N. Chen for providing us with the segmentations.

7. References

[1] M.A. Zissman, T. Gleason, D. Rekart, and B. Losiewicz, "Automatic Dialect Identification of Extemporaneous Conversational, Latin American Spanish Speech," in *Proceedings of the ICASP*, USA, 1996.

[2] F. Biadsy, J. Hirschberg, and N. Habash, "Spoken Arabic Dialect Identification Using Phonotactic Modeling," in *Proceedings of EACL 2009 Workshop on Computational Approaches to Semitic Languages*, Athens, Greece, 2009.

[3] D.A. Reynolds, T.F. Quatieri, and R.B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1-3, pp. 19 – 41, 2000.

[4] P.A. Torres-Carrasquillo, T. P. Gleason, and D. A. Reynolds, "Dialect identification using Gaussian Mixture Models," in *Proceedings of the Speaker and Language Recognition Workshop, Spain*, 2004.

[5] B. Burget, P. Matejka, and J. Cernock, "Discriminative training techniques for acoustic language identification," in *Proceedings of ICASSP*, 2006.

[6] P. Matejka, L. Burget, P. Schwarz, and J. Cernocky, "Brno university of technology system for nist 2005 language recognition evaluation," in *Proceedings of Odyssey*, 2006.

[7] P.A. Torres-Carrasquillo, D. Sturim, D. Reynolds, and A. McCree, "Eigen-channel Compensation and Discriminatively Trained Gaussian Mixture Models for Dialect and Accent Recognition," in *INTERSPEECH*, Brisbane, Australia, 2008.

[8] F. Biadsy and J. Hirschberg, "Using Prosody and Phonotactics in Arabic Dialect Identification," in *Proceedings of INTERSPEECH*, UK, 2009.

[9] Appen Pty Ltd, "Gulf/Iraqi/Levantine Arabic Conversational Telephone Speech – Linguistic Data Consortium, Philadelphia," Sydney, Australia, 2006, 2007.

[10] P. Boersma and D. Weenink, "Praat, a system for doing phonetics by computer," 2001, Software available at www.praat.org.

[11] H. Soltau, G. Saon, B. Kingsbury, H.K. Kuo, D. Povey, and A. Emami, "Advances in arabic speech transcription at IBM under DARPA GALE program," *EEE Transactions on Audio, Speech and Language Processing*, vol. 17, no. 5, pp. 884–895, 2009.

[12] F. Biadsy, N. Habash, and J. Hirschberg, "Improving the Arabic Pronunciation Dictionary for Phone and Word Recognition with Linguistically-Based Pronunciation Rules," in *Proceedings of NAACL/HLT 2009, Colorado, USA*, 2009.

[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, no. 5, pp. 308–311, May 2006.

[14] F. Biadsy, H. Soltau, L. Mangu, J. Navratil, and J. Hirschberg, "Discriminative phonotactics for dialect recognition using context-dependent phone classifiers – to be published," in *Odyssey*, 2010.

[15] C.C. Chang and C.J. Lin, "LIBSVM: a library for support vector machines," 2001, Software available at www.csie.ntu.edu.tw/ cjlin/libsvm.

[16] P.A. Torres-Carrasquillo, E. Singer, M.A. Kohler, R.J. Greene, D.A. Reynolds, and J.R. Deller Jr., "Approaches to language identification using gaussian mixture models and shifted delta cepstral features," in *Proceedings of ICSLP*, 2002.

[17] E. Wong and S. Sridharan, "Methods to improve gaussian mixture model based language identification system," in *ICSLP*, 2002.

[18] T.F. Wu, C.J. Lin, and R.C. Weng, "Probability estimates for multi-class classification by pairwise coupling," in *Journal of Machine Learning Research 5*, 2004.