Exploiting convolutional neural networks for phonotactic based dialect identification

Maryam Najafian[1], *Sameer Khurana*[1], *Suwon Shon*[1], *Ahmed Ali*[2], *James Glass*[1]

[1] MIT Computer Science and Artificial Intelligence Laboratory (CSAIL), Cambridge, MA, USA
[2]Qatar Computing Research Institute, HBKU, Doha, Qatar

[najafian,skhurana,swshon,glass]@csail.mit.edu, [amali]@qf.org.qa

Abstract

In this paper, we investigate different approaches for Dialect Identification (DID) in Arabic broadcast speech. Dialects differ in their inventory of phonological segments. This paper proposes a new phonotactic based feature representation approach which enables discrimination among different occurrences of the same phone n-grams with different phone duration and probability statistics. To achieve further gain in accuracy we used multi-lingual phone recognizers, trained separately on Arabic, English, Czech, Hungarian and Russian languages. We use Support Vector Machines (SVMs), and Convolutional Neural Networks (CNNs) as backend classifiers throughout the study. The final system fusion results in 24.7% and 19.0% relative error rate reduction compared to that of a conventional phonotactic DID, and i-vectors with bottleneck features.

Index Terms: Dialect identification, phonotactics, CNN

1. Introduction

The speech signal contains information beyond its linguistic content, including clues to the speaker's age, gender, dialect, social background, and level of education [1, 2, 3]. Dialects differ in their inventory of phonological segments and their distribution in the lexicon. Phonetic variation across different dialects can involve large spectro-temporal changes in realization of phonological units. Dialect variation is not just a shift in phonetic realization. The speech technology has yet to deal adequately with pronunciation variation across different groups of dialects. Each dialect has its own particular patterns of pronunciation, and there are often certain words or phrases that are specifically being spoken among speakers of a certain dialect. Dialect is a major source of variability for Automatic Speech Recognition (ASR) and leads to major drop in accuracy [4, 5]. In speech synthesis, synthetic voices are based on one accent due to the prevailing use of corpus-based synthesis methods operating from the speech of a single speaker. A good Dialect Identification (DID) system can facilitate the identification of dialectal segments from a transcribed speech dataset, and help with addressing the multi-conditional data problem caused by dialect variabilities [6]. Training an accurate DID will help with reducing the error rate for dialectal speech recognition by selecting a dialect-specific acoustic model, or a dialect-specific pronunciation dictionary, or by incorporating features which incorporate dialectal information features [4, 7, 5, 8].

This paper starts by reviewing related works for DID in Section 2 and describing our database in Section 3. Section 4 introduces two successful approaches to DID namely, phonotactics [9] and i-vectors [1]. Then, it introduces a new feature representation mechanism for the photostatic DID which incorporates the phone duration and probabiliy statistics. Finally, Section 5 compares the conventional and proposed systems and reports the system combination accuracy and the confusion matrix.

2. Related work

Low-level acoustic features help with distinguishing among different dialects on the basis of acoustic variabilities, while high-level phonetic and lexical features help with lexical variations across different Arabic dialects [1, 10, 11]. Proposed approaches for DID fall into three main categories, namely lexical, phonotactic and, acoustic. Character n-gram models, roots, morphology, words, and grammars have been studied as part of a lexical approach [12, 13], and effectiveness of logistic regression, recurrent neural networks, and SVM classifiers has been investigated [14, 15, 16]. Modeling phone n-gram sequences and subspaces were studied [9] as part of the phonotactic approach, and its fusion with acoustic approaches achieved high accuracy [17]. The use of acoustic features such as shifted delta Cepstral coefficients [3] and prosodic features [18], frame-by-frame phone posteriors [19], i-vectors [1, 20, 21] and classification using long short-term memory (LSTM)s [22], and non-negative factor analysis for GMM weight decomposition and adaptation [23] achieved major success.

3. Speech corpora

In this study our database comes from a multi-dialectal speech corpus comprising four Arabic dialects, namely Egyptian (EGY), Gulf (GLF), Levantine (LEV), and North African (NOR) as well as Modern Standard Arabic (MSA) from broadcast, debate and discussion programs from Al Jazeera, and as such contains a combination of spontaneous and scripted speech [24]. Egyptian is an urban dialect spoken in Cairo and Alexandria. Gulf is a dialect from the Arabic Gulf countries of Bahrain, Kuwait, Oman, Saudi Arabia, United Arab Emirates, and sometimes Iraq are often grouped together. The Levantine dialect includes dialects from Jordan, Palestine, and Syria. Modern Standard Arabic includes formal Arabic speech (news). The North African dialect includes Algeria, Libya, Morocco, and Tunisia [22]. The recordings are segmented in order to avoid speaker overlap, and any non-speech aspects, such as music and background noise, are removed; more detail about the training data can be found in [23]. As shown in Table 1, our database represents five Arabic dialects. In our experiments the hyper parameters are selected after 5 fold cross-validation on the training set and our final accuracy is reported on the unseen test set.

4. System description

Arabic dialects differ substantially in terms of phonology, morphology, lexical choice and syntax. Hence, Phonetically aware

	Training			Test		
	Sent.	Dur.	Words	Sent.	Dur.	Words
EGY	5k	23.4	87k	302	2	11.6k
GLF	4.7k	21.9	67.9k	250	2.1	12.3k
LAV	4.9k	20.6	63.3k	334	2	10.9k
MSA	4.2k	23.8	82.4k	262	1.9	13k
NOR	4.9k	20.4	47.1k	344	2.1	10.3k
Tot.	15524	110.1	347.5k	1492	10.1	58.1k

Table 1: *Data Corpus. # training & test sentences, # words, speech duration (Dur.) in hours*

models could be beneficial for dialect identification, since they provide a mechanism to focus attention on small phonetic differences between dialects with predominantly common phonetic inventories. In this section we describe both (1) conventional and (2) proposed phonotactic systems with CNN and SVM classifiers, and (3) i-vectors with bottleneck features.

4.1. Baseline: phonotactics system

Employing language-dependent parallel phone recognizers trained from labeled speech has proven to be a successful approach in a number of language and accent identification tasks [25, 26]. More recently, multi-lingual and multi-accent Parallel phonotactic systems have obtained a great success in recognizing 14 different British English regional accents [3, 17]. In our baseline system we use a conventional phonotactic system and investigate the classification accuracy using a SVM and a CNN classifier. The baseline phonotactic approach relies on the raw phone sequence distribution and phone n-gram frequency statistics to recognize the speaker's dialect.

4.2. Proposed: phonotactics system

Our proposed approach takes into account additional phone level statistics, such as phone duration and posterior probability. This proposed mechanism enables discrimination among different occurrences of the same phone sequences with different phone duration or probability statistics by adding a weight index to the corresponding phone representation in the sequence (relabeling stage). This relabeling stage draws classifier's attention not only to phone sequences and n-gram frequencies, but also to phone duration and phone probability statistics by using a new feature representation using a weight index for each phone. For each speaker, given the mean, M, and the standard deviation, S, of phone c's duration D, and phone c's probability, P, the relabeling mechanism is shown bellow. Each phone c is relabeled as cn with a weight index $n = 1, 2, 3, 4$.

Algorithm 1 Phone representation with phone duration index

for c in utterance's phone trasciption **do**
  if $D(c) < M - 0.5S$ **then**
    $c \leftarrow c1$
  else $\{M - 0.5S < D(c) < M\}$
    $c \leftarrow c2$
  else $\{M < D(c) < M + 0.5S\}$
    $c \leftarrow c3$
  else
    $c \leftarrow c4$
  end if
end for

Figure 1 shows a conventional and our proposed phonotactic system. After running our proposed DID phonotactics system, a round of score fusion using Linear Logistic Regression (LLR) [27] will be applied to the classification scores derived from a SVM or a CNN-based classifier. The LLR weights are selected after 5 fold cross-validation on the training set.

Algorithm 2 Phone representation with phone probability index

for c in utterance's phone trasciption **do**
  if $P(c) < M - 0.5S$ **then**
    $c \leftarrow c1$
  else $\{M - 0.5S < P(c) < M\}$
    $c \leftarrow c2$
  else $\{M < P(c) < M + 0.5S\}$
    $c \leftarrow c3$
  else
    $c \leftarrow c4$
  end if
end for

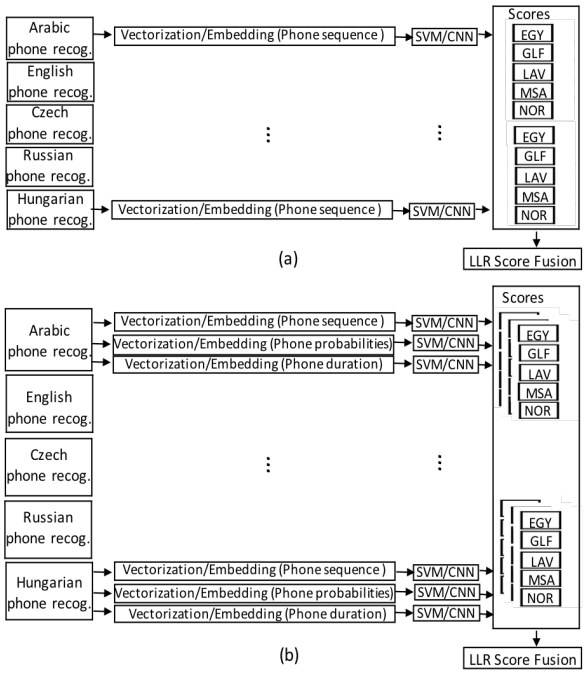

Figure 1: *Fusion of classifier's scores from parallel phonotactic DID systems (a) baseline, and (b) proposed systems*

4.2.1. Multi-lingual phone recognizers

In our phonotactic systems, we use multi-lingual phone recognizers. For the Arabic recognizer the LSTM-HMM based acoustic model is trained using 1400 hours of training data from the GALE Arabic and MGB3 Arabic datasets [4]. For more details see the system description in [4]. In addition to the Arabic recognizer we exploited four existing phone recognizers (English, Hungarian, Czech, and Russian) from a toolkit developed by Brno University of Technology [28]. The English system was trained on the TIMIT database, and the rest were trained on the SpeechDat-E databases using a hybrid approach based on Neural Networks and Viterbi decoding. In addition, since our recognition domain is for Arabic dialects, we built a GMM-HMM based recognizer.

4.2.2. SVM-based classifier

During a phonotactic DID process, each utterance is passed through a phone recognizer to generate phone-level transcriptions followed by a vectorization stage. During the vectorization, for each utterance the relative frequency of each pre-

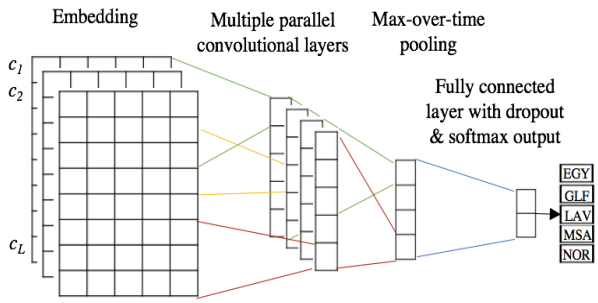

Figure 2: *Architecture of our CNN-based classifier*

defined set of phone n-grams is computed. Using the TFIDF weighting technique [29] those n-gram components which have a low occupancy in all accents are de-emphasized since they do not carry any useful information, while more emphasize is applied to more discriminative components that occur only for certain dialects. In this work a multi-class SVM [30, 31] classifier with a linear Kernel is trained to create a mapping between the phone n-gram sequences and different dialect labels. The phone level transcription from the test speaker's utterance is scored against each SVM (using a 'one against all' approach). Hyper-parameters for the SVM are distance from the hyperplane, set to 0.01, and l2 penalty. The dialect label with maximum score determines the test utterance's dialect.

4.2.3. CNN-based classifier

The convolutional neural network (CNN) model has been shown to be effective for many speech and language processing applications. CNN filters can learn specific patterns of sounds, phones, or characters from each input utterance that are relevant for a particular task. Our previous studies have shown encouraging results when using Deep Learning methods directly on raw acoustic features, different word and sub-word units for DID [10, 32]. Given a phone sequence and associated dialect labels, we need to find a neural network based predictor, embedding layer, to create a mapping between the phone sequence c to a vector sequence x. Here, a 5-fold cross-validation is performed on the training dataset to select the values for the hyper-parameters and the final evaluation is conducted on the test set. Our CNN architecture comprises of two convolution layers, the first layer of the CNN is followed by Max Pooling operation, and the second layer is followed by a global max pooling and a fully connected hidden layer with 0.2 dropout. During this stage vector sequence x is mapped to a single final hidden vector h representing the entire sequence. The final representation is fed to a softmax layer that maps h to a probability distribution over labels. During training, each sequence is fed into this network to create label predictions. As errors are back-propagated down the network, the weights at each layer are updated, including the embedding layer. During testing, the learned weights are used in a forward step to compute a prediction over the labels. We always take the best predicted label for evaluation.

Parameter set up: In this task we extract the corresponding phone sequence from multi-lingual recognizers and feed the sequence statistics to a multi-class SVM or a CNN-based classifier. The SVM classifier uses the phone sequence output from each phone recognizer and uses n-grams with $n = 1, 2, 3, 4, 5, 6$. After 5 fold cross-validation on the training dataset, the best accuracy on is obtained with $n = 5$. The final evaluation is conducted on the test set with $n = 5$.

The CNN classifier takes as input phone sequences generated by five different language recognizers, namely Arabic, English, Czech, Hungarian and Russian. Our predictor is a neural network over phone sequences generated by five different languages, namely Arabic, English, Czech, Hungarian and Russian. System hyperparameters, such as embedding layer dropout ρ_{emb}, fully-connected layer dropout ρ_{fc}, maximum text length L, phone embedding size d_{emb}, and fully-connected layer output size d_{fc}, were tuned on the development set for values of: $\rho_{emb} \in \{\underline{0.1}, 0.2, 0.5\}$, $\rho_{fc} \in \{0.1, \underline{0.2}, 0.5\}$, $L \in \{400, \underline{600}, 800\}$, $d_{emb} \in \{50, 150, \underline{200}, 300\}$, and $d_{fc} \in \{100, \underline{250}, 500\}$ (chosen parameters in underline). For the convolutional layers, we experimented with different combinations of filter widths and number of filters. We started with a single filter width and noticed that a width of 5-grams performs fairly well with enough filters (200). We then added multiple widths, and our best configuration on the development set was: $\{1*50, 2*50, 3*100, 4*100, 5*200, 6*200, 7*300, 8*300\}$, where $w * n$ indicates n filters of width w. We train the entire network jointly, including the embedding layer. We use the Adam optimizer [33] with the default original parameters to minimize the cross-entropy loss. Training is run with shuffled mini-batches of size 16 and stopped once the loss on the development set stops improving; we allow up to 20 epochs.

4.3. Baseline: I-vector with BNF features

I-vectors provide a low-dimensional representation of feature vectors that can be successfully used for classification and recognition tasks. The i-vector system is used as a comparison with the phonotactic DID systems. I-vectors are extracted using the standard pipeline [1]. Our i-vector system is based on two successive Deep Neural Network (DNN) ASR models, each with 5 hidden layers and 1 linear BN layer with tied-states as target outputs. The tied-state triphone labels are generated by a forced alignment from an HMM-GMM baseline trained on 1200 hours of MSA news recordings. The input to the first DNN consists of 23 critical-band energies that are obtained from Mel filter-bank. Pitch and voicing probability are then added. Eleven consecutive frames are then stacked together. The second DNN is used for correcting the posterior outputs of the first DNN. In this architecture, the input features of the second DNN are the outputs of the BN layer from the first DNN. Context expansion is achieved by concatenating frames with time offsets of -10, -5, 0, 5, and 10. Thus, the overall time context seen by the second DNN is 31 frames. After extracting the bottleneck features (BNFs) they are fed as an input to the i-vector system to train a Gaussian Mixture Model-Universal Background Model (GMM-UBM). The GMM-UBM's mean supervector extracted and adapted to each utterance. This update information is encoded in a low-dimensional latent vector known as an i-vector. In this work, the GMM-UBM model has 2048 Gaussian components, MFCC features are extracted using a 25 ms window and the i-vectors are 400 dimensional. We also perform Linear Discriminant Analysis (LDA) and Within-Class Co-variance Normalization (WCCN). The resulting i-vectors are input to an SVM classifier. The resulting i-vectors are input to an SVM classifier. Hyper-parameters for the SVM are distance from the hyperplane, set to 0.01, and l2 penalty.

5. Results and discussions

In this section we compare the results of our proposed method with successful baseline acoustic and phonotactic DID ap-

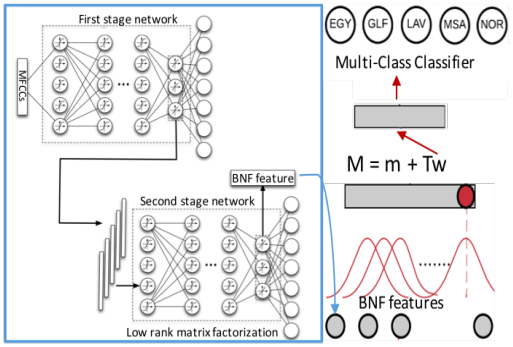

Figure 3: *I-vector based DID system*

proaches.

Baseline i-vector system: The i-vector based DID system achieves 67% accuracy using an SVM recognizer.

Phonotactic system: Table 2, shows the results for Arabic phonotactics DID employing five language-dependent parallel phone recognizers. The accuracy (Acc.) of a baseline phonotactic system using phone n-gram sequence (seq.) statistics with SVM and CNN classifiers is reported for each phone recognizer. Our experiment on the baseline phonotactic system shows that the CNN-based classifier outperforms the multi-class SVM classifier in all five setups. For instance, a baseline phonotactic DID trained on Hungarian achieves 57.85% accuracy which is surprisingly very close to that of a DID system trained on Arabic with 57.91% accuracy. The final system fusion across all 5 systems results in 64.50% and 62.12% DID accuracy using a CNN and a SVM classifier respectively.

Language	(%) Phone n-gram seq. Acc. SVM	(%) Phone n-gram seq. Acc. CNN
Arabic	56.82	57.91
English	56.03	56.88
Russian	56.25	57.12
Czech	56.64	57.62
Hungarian	56.71	57.85
Fusion	62.12	64.50

Table 2: *Employing language-dependent parallel PRLMS in a conventional versus an attention-based context for DID*

Proposed phonotactic system: As mentioned in Section 4.2, we use the output statistics of phone recognizers, (a) phone sequences, (b) phone duration, and (c) probability statistics to creat new phone represntations that capture not only the phone sequences but also their relative duration and probabilities with modified indexes. Other rows report the accuracy using the relabeled phone n-gram sequences for each language. For the proposed phonotactic DID task fusing all five parallel multi-lingual DIDs leads to 71.60 % accuracy while fusing the scores from the top three multi-lingual PRLMs, namely Arabic, Hungarian, and Czech achieves 73.27% accuracy.

Confusion matrix for the final system combination: The confusion matrix for our top best system combination is shown in Figures 4. In general these DID systems perform worst in case of GLF, NOR, and LAV dialects. Our proposed approach has led to 14.73%, 24.01%, 19.63%, 18.21%, and 23.97% relative DID error rate reduction for EGY, GLF, LAV, MSA, and NOR respectively. Interestingly, these results show that the relative error rate has reduced more dramatically for the difficult dialects (GLF, NOR, and LAV) while it had a lower impact on the less difficult dialects (EGY and MSA).

Language	System	(%) Acc.
Arabic	Phone n-gram sequence with CNN	57.91
	Phone n-gram (duration relabeled) with CNN	59.55
	Phone n-gram (probability relabeled) with CNN	59.72
	LLR fusion of 3 systems	68.95
English	Phone n-gram sequence with CNN	56.88
	Phone n-gram (duration relabeled) with CNN	56.30
	Phone n-gram (probability relabeled) with CNN	56.24
	LLR fusion of 3 systems	63.70
Russian	Phone n-gram sequence with CNN	57.12
	Phone n-gram (duration relabeled) with CNN	57.59
	Phone n-gram (probability relabeled) with CNN	57.29
	LLR fusion of 3 systems	65.10
Czech	Phone n-gram sequence with CNN	57.62
	Phone n-gram (duration relabeled) with CNN	57.71
	Phone n-gram (probability relabeled) with CNN	57.37
	LLR fusion of 3 systems	67.85
Hungarian	Phone n-gram sequence with CNN	57.85
	Phone n-gram (duration relabeled) with CNN	58.74
	Phone n-gram (probability relabeled) with CNN	58.90
	LLR fusion of 3 systems	68.31
Fusion	LLR fusion of all systems	71.60
	LLR fusion of Arabic, Hungarian, and Czech systems	73.27

Table 3: *Employing language-dependent parallel PRLMS in a conventional versus an attention-based context for DID*

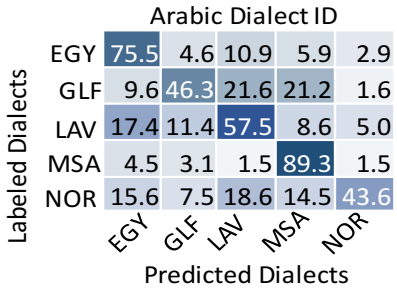

Figure 4: *DID Confusion Matrix for the final combined proposed phonotactic system*

6. Conclusions

In this paper, we present a comprehensive performance study of Spoken DID methods for the Arabic language. Along with investigating the traditional methods for DID such as i-vector and n-gram phonotactic features with an SVM classifier, we also investigate the advantages of using a CNNs for direct mapping of acoustic and phonotactic features to one of the five dialects. We have demonstrated a new approach for phonotactic dialect Identification with a noval feature representation methodology which captures phone duration, and probability statistics as well as phone sequences. This system achieves 73.27% accuracy using a system combination comprising multi-lingual phonotactic systems trained on Arabic, English, Russian, Czech, and Hungarian. We studied the dialect identification error patterns using a confusion matrix. The final system fusion for our proposed phonotactic system results in 24.7% and 19% relative error rate reduction compared to that of the fused baseline multi-lingual phonotactics and the ivector with BNF features. For our future work, we would continue our investigation into approaches that can directly map the raw acoustic waveform to the corresponding dialects. In particular, we would explore Long Short-Term Memory RNN to make dialect predictions per frame.

7. References

[1] A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. Harsha Yella, J. Glass, P. Bell, and S. Renals, "Automatic dialect detection in arabic broadcast speech," in *INTERSPEECH*, 2016, pp. 2934–2938.

[2] A. Hanani, M. Russell, and M. J. Carey, "Speech-based identification of social groups in a single accent of british english by humans and computers," in *ICASSP*, 2011, pp. 4876–4879.

[3] A. Hanani, M. J. Russell, and M. J. Carey, "Human and computer recognition of regional accents and ethnic groups from British English speech," *CSL*, pp. 59–74, 2013.

[4] M. Najafian, W.-N. Hsu, A. Ali, and J. Glass, "Automatic speech recognition of arabic multi-genre broadcast media," in *ASRU*, 2017.

[5] M. Najafian, A. DeMarco, S. J. Cox, and M. J. Russell, "Unsupervised model selection for recognition of regional accented speech," in *INTERSPEECH*, 2014, pp. 2967–2971.

[6] M. Najafian, S. Safavi, J. H. Hansen, and M. Russell, "Improving speech recognition using limited accent diverse british english training data with deep neural networks," in *MLSP*, 2016, pp. 1–6.

[7] M. Tjalve and M. Huckvale, "Pronunciation variation modelling using accent features," in *INTERSPEECH*, 2005, pp. 1341–1344.

[8] P. Motlícek, P. N. Garner, N. Kim, and J. Cho, "Accent adaptation using Subspace Gaussian Mixture Models," in *ICASSP*, 2013, pp. 7170–7174.

[9] M. A. Zissman, T. P. Gleason, D. Rekart, and B. L. Losiewicz, "Automatic dialect identification of extemporaneous conversational, Latin American Spanish speech," in *ICASSP*, vol. 2. IEEE, 1996, pp. 777–780.

[10] S. Khurana, M. Najafian, A. Ali, T. Al Hanai, Y. Belinkov, and J. Glass, "QMDIS: QCRI-MIT advanced dialect identification system," in *INTERSPEECH*, 2017.

[11] S. Shon, A. Ali, and J. Glass, "MIT-QCRI Arabic Dialect Identification System for the 2017 Multi-Genre Broadcast Challenge," *ArXiv e-prints arXiv:1709.00387*, aug 2017. [Online]. Available: http://arxiv.org/abs/1709.00387

[12] D. A. Reynolds, W. M. Campbell, W. Shen, and E. Singer, "Automatic language recognition via spectral and token based approaches," in *Springer Handbook of Speech Processing*, 2008, pp. 811–824.

[13] A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. H. Yella, J. Glass, P. Bell, and S. Renals, "Automatic dialect detection in arabic broadcast speech," *SLT*, 2015.

[14] S. Khurana and A. Ali, "QCRI advanced transcription system (QATS) for the arabic multi-dialect broadcast media recognition: MGB-2 challenge," in *SLT*, 2016, pp. 292–298.

[15] A. Zirikly, B. Desmet, and M. Diab, "The gw/lt3 vardial 2016 shared task system for dialects and similar languages detection," *VarDial 3*, p. 33, 2016.

[16] S. Malmasi, M. Zampieri, N. Ljubešic, P. Nakov, A. Ali, and J. Tiedemann, "Discriminating between similar languages and arabic dialect identification: A report on the third dsl shared task," *VarDial 3*, p. 1, 2016.

[17] M. Najafian, S. Safavi, P. Weber, and M. J. Russell, "Identification of British English regional accents using fusion of i-vector and multi-accent phonotactic systems," in *ODYSSEY*, 2016, pp. 1–6.

[18] D. Martínez, L. Burget, L. Ferrer, and N. Scheffer, "ivector-based prosodic system for language identification," in *ICASSP*. IEEE, 2012, pp. 4861–4864.

[19] O. Plchot, M. Diez, M. Soufifar, and L. Burget, "Pllr features in language recognition system for rats." in *INTERSPEECH*, 2014, pp. 3047–3051.

[20] M. H. Bahari, R. Saeidi, D. Van Leeuwen *et al.*, "Accent recognition using i-vector, gaussian mean supervector and gaussian posterior probability supervector for spontaneous telephone speech," in *ICASSP*, 2013, pp. 7344–7348.

[21] A. DeMarco and S. J. Cox, "Native accent classification via i-vectors and speaker compensation fusion," in *INTERSPEECH*, 2013, pp. 1472–1476.

[22] A. Hanani, A. Qaroush, and S. Taylor, "Classifying ASR transcriptions according to Arabic dialect," *VarDial 3*, p. 126, 2016.

[23] M. H. Bahari, N. Dehak, L. Burget, A. M. Ali, J. Glass *et al.*, "Non-negative factor analysis of gaussian mixture model weight adaptation for language and dialect recognition," *ACM transactions on ASLP*, vol. 22, no. 7, pp. 1117–1129, 2014.

[24] S. Wray and A. Ali, "Crowdsource a little to label a lot: Labeling a speech corpus of dialectal Arabic," *INTERSPEECH*, pp. 2824–2828, 2015.

[25] M. Zissman, E. Singer *et al.*, "Automatic language identification of telephone speech messages using phoneme recognition and n-gram modeling," in *ICASSP-94*, vol. 1, 1994, pp. I–305.

[26] F. S. Richardson, W. M. Campbell, and P. A. Torres-Carrasquillo, "Discriminative n-gram selection for dialect recognition." in *INTERSPEECH*, 2009, pp. 192–195.

[27] L.-F. Zhai, M.-H. Siu, X. Yang, and H. Gish, "Discriminatively trained language models using Support Vector Machines for language identification," in *ODYSSEY*, 2006, pp. 1–6.

[28] P. Matejka, P. Schwarz, J. Cernockỳ, and P. Chytil, "Phonotactic language identification using high quality phoneme recognition." in *INTERSPEECH*, 2005, pp. 2237–2240.

[29] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information processing & management*, vol. 24, no. 5, pp. 513–523, 1988.

[30] L.-F. Zhai, M.-H. Siu, X. Yang, and H. Gish, "Discriminatively trained language models using Support Vector Machines for language identification," in *ODYSSEY*, 2006, pp. 1–6.

[31] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[32] Y. Belinkov and J. Glass, "A Character-level Convolutional Neural Network for Distinguishing Similar Languages and Dialects," in *VarDial*, 2016.

[33] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CVRL*, 2014.