Dialect/Accent Classification Using Unrestricted Audio

Rongqing Huang, *Student Member, IEEE*, John H. L. Hansen, *Fellow, IEEE*, and Pongtep Angkititrakul, *Member, IEEE*

***Abstract*—This study addresses novel advances in English dialect/accent classification. A word-based modeling technique is proposed that is shown to outperform a large vocabulary continuous speech recognition (LVCSR)-based system with significantly less computational costs. The new algorithm, which is named Word-based Dialect Classification (WDC), converts the text-independent decision problem into a text-dependent decision problem and produces multiple combination decisions at the word level rather than making a single decision at the utterance level. The basic WDC algorithm also provides options for further modeling and decision strategy improvement. Two sets of classifiers are employed for WDC: a word classifier $D_{W(k)}$ and an utterance classifier D_u. $D_{W(k)}$ is boosted via the AdaBoost algorithm directly in the probability space instead of the traditional feature space. D_u is boosted via the dialect dependency information of the words. For a small training corpus, it is difficult to obtain a robust statistical model for each word and each dialect. Therefore, a context adapted training (CAT) algorithm is formulated, which adapts the universal phoneme Gaussian mixture models (GMMs) to dialect-dependent word hidden Markov models (HMMs) via linear regression. Three separate dialect corpora are used in the evaluations that include the Wall Street Journal (American and British English), NATO N4 (British, Canadian, Dutch, and German accent English), and IViE (eight British dialects). Significant improvement in dialect classification is achieved for all corpora tested.**

***Index Terms*—Accent/dialect classification, AdaBoost algorithm, context adapted trianing, dialect dependency information, limited training data, robust acoustic modeling, word-based modeling.**

I. Introduction

DIALECT/ACCENT is a pattern of pronunciation and/or vocabulary of a language used by the community of *native/nonnative* speakers belonging to some geographical region. For example, American English and British English are two dialects of English; English spoken by native Chinese or German are two accents of English. Some researchers have a slightly different definition of dialect and accent, depending on whether they approach the problem from a linguistics or speech science/engineering perspective. In our study, we will use "dialect" and "accent" interchangeably. In this study, we wish to detect the dialect of an unrestricted (i.e., speaker-independent, transcript unknown) audio utterance from a predefined set of N dialect classes. Accent detection, or as it is sometimes referred to as accent classification [1], is an emerging topic of interest in the automatic speech recognition (ASR) community since accent is one of the most important factors next to gender that influence ASR performance [10], [12]. Accent knowledge could be used in various components of the ASR system such as pronunciation modeling [23], lexicon adaptation [36], and acoustic model training [14] and adaptation [3].

Dialect classification of a language is similar to language identification (LID). There are many previous studies on LID. The popular methods are based on phone recognition such as single language Phone recognition followed by language-dependent language modeling (PRLM), parallel PRLM, and language-dependent parallel phone recognition (PPR) [16], [40]. It is well known that low-level features such as Mel frequency cepstral coefficients (MFCCs) cannot provide sufficient discriminating information for LID. Jayram *et al.* [16] proposed a parallel subword recognizer for LID. Rouas *et al.* [32] evaluated the relevance of prosodic information such as rhythm and intonation for LID. Parandekar and Kirchhoff [28] applied an n-gram modeling of parallel streams of articulatory features which include manner of articulation, consonantal place of articulation, and even the phone sequence was treated as a feature stream. Gu and Shibata [9] proposed a predictive vector quantization (VQ) technique with several high-level features such as tone of voice, rhythm, style, and pace to identify languages. Most of the above techniques can be directly employed in dialect classification.

There are far fewer studies addressing dialect classification. Kumpf and King [19] applied linear discriminant analysis (LDA) classification on individual phonemes to analyze three accents in Australian English. Miller and Trischitta [27] selected phoneme sets including primary vowels for an analysis on the TIMIT American English dialects. Yan and Vaseghi [39] applied formant vectors instead of MFCC to train hidden Markov models (HMMs) and Gaussian mixture models (GMMs) for American, Australian, and British English accent analysis. Lincoln *et al.* [22] built phonotactic models with the CMU American English pronunciation dictionary and the BEEP British English pronunciation dictionary for American and British English accent classification. Angkititrakul and Hansen [1] proposed trajectory models to capture the phoneme temporal structure of Chinese, French, Thai, and Turkish accents in English.

Manuscript received August 22, 2005; revised March 11, 2006. This work was supported by the U.S. Air Force Research Laboratory, Rome, NY under Contract FA8750-04-1-0058. Any opinions, findings and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the U.S. Air Force. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mari Ostendorf.

R. Huang was with the Center for Robust Speech Systems (CRSS), Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX, 75083-0688 USA. He is now with Nuance Communications, Burlington, MA 01803 USA.

J. H. L. Hansen and P. Angkititrakul are with the Center for Robust Speech Systems (CRSS), Department of Electrical Engineering, University of Texas at Dallas, Richardson, TX, 75083-0688 USA (e-mail: John.Hansen@utdallas.edu; angkitit@utdallas.edu).

Digital Object Identifier 10.1109/TASL.2006.881695

In this paper, we focus our attention on English, and suggest that application to other languages is straightforward. In order to achieve reasonable identification accuracy in English dialect/accent classification, it is first necessary to understand how dialects differ. Fortunately, there are numerous studies that have considered English dialectology [30], [35], [37]. While there are many factors which can be considered in the analysis of dialect, English dialects generally differ in the following ways [37]:

1) phonetic realization of vowels and consonants;
2) phonotactic distribution (e.g., rhotic and nonrhotic in *farm*: /F AA R M/ versus /F AA M/);
3) phonemic system (the number or identity of phonemes used);
4) lexical distribution (word choice or word use frequency);
5) rhythmical characteristics:
   - syllable boundary (e.g., *self#ish* versus *sel#fish*);
   - pace (average number of syllables uttered per second);
   - lexical stress (across word or phrase);
   - intonation (sentence level, semantic focus);
   - voice quality (e.g., creaky voice versus breathy voice).

The first four areas above are visible at the word level. All the rhythmical characteristics except intonation can be, or at least partially, represented at the word level [37]. In [30], a single word "hello" was used to distinguish three dialects in American English. In our experiments, it is observed that human listeners can make comfortable decisions on English dialects based on isolated words. Individual words do encode high-level features such as formant and intonation structure to be useful for dialect classification. From a linguistic point of view, a word may be the best unit to classify dialects. However, for an automatic speech-based classification system, it is impossible to construct statistical models for all possible words from even a small subset of dialects. Fortunately, the words in a language are very unevenly distributed. The 100 most common words account for 40% of the occurrences in the Wall Street Journal (WSJ) corpus [24], which has 20K distinct words, and account for 66% in the SwitchBoard corpus [8], which has 26K distinct words. Therefore, only a small set of words is required for modeling. In [18], [24], and [31], word level information was embedded into phoneme models and improvement in language identification was achieved. In this study, a system based only on word models named Word-based Dialect Classification (WDC) is proposed and will be shown to outperform a large-vocabulary continuous speech recognition (LVCSR)-based system, which is claimed to be the best performing system in language identification [41].

The WDC turns a single text-independent decision problem into a multiple text-dependent decision problem. There are two sets of classifiers in the WDC system: a word classifier $D_{W(k)}$ and an utterance classifier D_u. WDC provides options for alternative decision and modeling technique improvement as well. The AdaBoost algorithm [6] is an ensemble learning algorithm. In [4], [5], and [26], different researchers applied the AdaBoost algorithm to GMM/HMM-based modeling and obtained small but consistent improvement with large computational costs. In this study, the AdaBoost algorithm is applied directly to our word classifier $D_{W(k)}$ in the probability space instead of the feature space, where the latter results in model training for each iteration. This method obtains significant improvement with small computational costs. The dialect dependency of words is also considered and embedded within the WDC framework through the utterance classifier D_u. For a small dialect corpus, the primary problem of formulating a word-based classification algorithm is that there is not sufficient training data to model each word for each dialect robustly. A context adaptive training (CAT) algorithm is formulated to address this problem. First, all dialect data is grouped together to train a set of universal phoneme GMMs; next, the word HMM is adapted from the phoneme GMMs with the limited dialect-specific word samples.

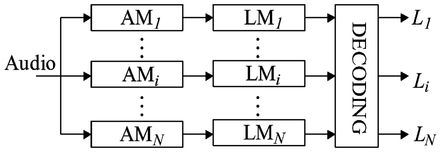


Fig. 1. LVCSR-based dialect classification system.

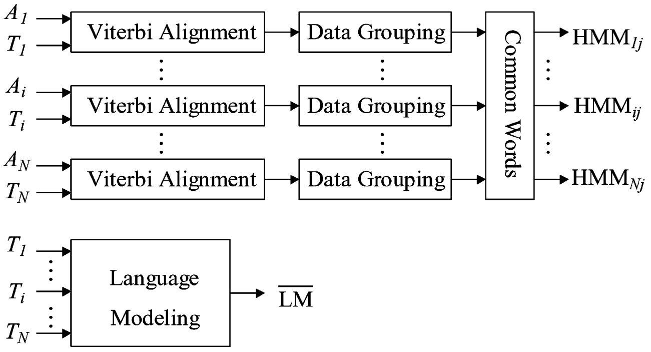


Fig. 2. Block diagram of WDC training framework.

The remainder of this paper is organized as follows: the LVCSR-based classification system is introduced in Section II as the baseline for our study. Section III is dedicated to the WDC algorithm and its extensions: Section III-A introduces the motivation of the basic WDC algorithm; Section III-B presents the method for boosting the word classifier $D_{W(k)}$; Section III-C introduces how to encode the dialect dependent information of words into the utterance classifier D_u; Section III-D proposes the CAT algorithm, which adapts the universal phoneme GMMs to the dialect-dependent word models. The CAT is specifically formulated for word modeling in a small audio corpus. Section IV presents system experiments using three corpora. Finally, conclusions are presented in Section V.

II. Baseline Classification System

It is known that LVCSR-based systems achieve high performance in language identification since they employ knowledge from individual phonemes, phoneme sequences within a word, and whole word sequences [41]. In several studies [11], [25], [34], LVCSR-based systems were shown to perform well for the task of language identification. A similar LVCSR-based system is employed as our dialect classification baseline system. Fig. 1 shows a block digram of the system, where N represents the

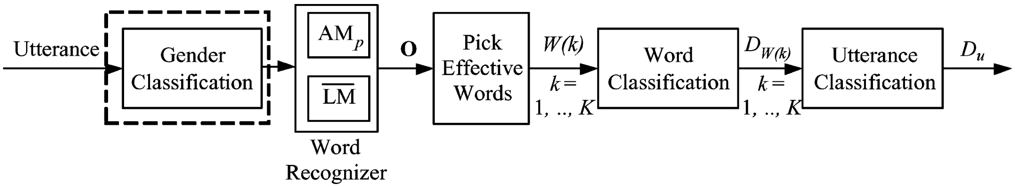


Fig. 3. Block diagram of WDC evaluation system.

number of dialects. In this figure, the blocks AM_i and LM_i represent the acoustic model (trained on triphones) and the language model (trained on word sequences) of dialect i respectively. AM_i and LM_i are trained with data from dialect i in the task. No additional data is added for model training. A common pronunciation dictionary is used for all $\mathrm{AM} + \mathrm{LM}$ pairs consisting of the publicly available CMU 125K American English dictionary [2]. Here, L_i represents the likelihood of dialect i. The final decision is obtained as follows:

$$D_L = \arg\max_i \; L_i, \quad i = 1, 2, \ldots, N. \tag{1}$$

The LVCSR-based system requires a significant amount of word level transcribed audio data to train the acoustic and language models for each dialect. Also, during the test phase, N recognizers must be employed in parallel. Because of this parallel structure, this computationally complex algorithm achieves very high dialect classification accuracy, and therefore represents a good baseline system for comparison.

III. WDC and Extensions

A. Basic WDC Algorithm

In this section, we formulate the basic word-based dialect classification algorithm. Fig. 2 shows the block diagram for training the WDC system. For dialect i, we require that audio data A_i and its corresponding word level transcript T_i are given. In this phase, Viterbi forced alignment is applied to obtain the word boundaries, and the data corresponding to the same word in that dialect is grouped together (i.e., "Data Grouping" block in Fig. 2). We determine the common words across all the dialects (i.e., "Common Words" block of Fig. 2) and maintain them as set $\mathcal{J}$. An HMM is trained for each word in set $\mathcal{J}$ and for each dialect. The number of states in the word HMM is set equal to the number of phonemes within the word. The number of Gaussian mixtures of the HMM is selected based on the size of the training data with a minimum of two. Therefore, the set of dialect-dependent word HMMs is summarized as

$$\Psi = \{\mathrm{HMM}_{ij}\}, \quad i = 1, 2, \ldots, N, \; j \in \mathcal{J}$$

where N is the number of dialects. Next, the transcript set $\mathcal{T} = \{T_1, \ldots, T_i, \ldots, T_N\}$ is used to train a language model $\overline{\mathrm{LM}}$ (see bottom of Fig. 2), which includes the common word set $\mathcal{J}$ and is used in the word recognizer (see Fig. 3) during the WDC evaluation.

Fig. 3 shows the block diagram of the WDC evaluation system. A gender classifier can be applied to the input utterance if gender-dependent dialect classification is needed. The common word recognizer is a dialect-independent recognizer and is applied to output word and boundary information of the incoming audio. The acoustic model in the word recognizer can be trained by grouping all dialect training data together. No additional data is necessary. A decision-tree triphone modeling technique is applied to train the dialect-independent acoustic model. However, we note that the accuracy of the acoustic model in the word recognizer has limited overall impact on dialect classification performance.[1] Therefore, it is not absolutely necessary to train an acoustic model for every new dialect classification task in a language. Since there are many existing well-trained triphone acoustic models in English available for speech recognition, a previously well-trained decision-tree triphone acoustic model AM_p can be used in our study as an alternative, which is independent of the dialect data and the task. The language model $\overline{\mathrm{LM}}$ in the common word recognizer is a task-dependent, dialect-independent model, which is trained with the transcripts of all the dialect data in the task as shown in Fig. 2. The task-dependent language model $\overline{\mathrm{LM}}$ is intentionally used for the word recognizer to output words which have previously trained word models. The common pronunciation dictionary is the publicly available CMU 125K American English dictionary [2]. The WDC system has small requirements on the word recognizer as shown in the experiments. Further discussion on the impact of acoustic and language models for the word recognizer will be presented in Section IV-A.

The word recognizer therefore outputs the word set $\mathbf{O}$ with boundary information. The effective word set $\mathbf{W}$ is represented as

$$W(k) \leftarrow O(l), \text{ if } O(l) \in \mathcal{J}, \; l = 1, 2, \ldots$$

where k is an index variable. After identifying and picking the words which have the pretrained dialect-dependent word HMMs, the words are scored and classified using these word HMMs. Word classification is based on a Bayesian classifier, where the decision $D_{W(k)}$ is

$$\begin{aligned} D_{W(k)} &= \arg\max_i Pr\left(W(k)|\mathrm{HMM}_{iW(k)}\right) \\ W(k) &\in \mathcal{J}, \; k = 1, 2, \ldots, K, \; i = 1, 2, \ldots, N \end{aligned} \tag{2}$$

where N is the number of dialects, $\mathcal{J}$ is the set of common words across the N dialects, $Pr(\cdot|\cdot)$ is the conditional probability, and $K = |\mathbf{W}|$ is the size of the effective word set $\mathbf{W}$. The final decision for the utterance classification is obtained by

[1] This observation will be shown in Section IV-A.

a majority vote of the word classifiers $D_{W(k)}, k = 1, 2, \ldots, K$, as

$$D_u = \arg\max_i \sum_{k=1}^{K} \mathcal{I}\left(D_{W(k)} = i\right), \quad i = 1, 2, \ldots, N. \tag{3}$$

Here, $\mathcal{I}(\nu)$ is the indicator function defined as

$$\mathcal{I}(\nu) = \begin{cases} 1, & \text{if } \nu \text{ is true} \\ 0, & \text{else.} \end{cases} \tag{4}$$

By comparing (1) with (2) and (3), we observe that the WDC system turns the single text-independent decision problem at the utterance level into a combination of text-dependent decision problems at the word level. The WDC framework also provides options for further modeling and decision space improvement which will be considered in the following sections.

B. Boosting Word Classifier $D_{W(k)}$ in the Probability Space

Let us first consider word classifier $D_{W(k)}$ in (2). For simplicity, let us represent the word $W(k)$ and HMM as

$$m \leftarrow W(k), \quad \Theta \leftarrow \text{HMM}$$

and define a probability vector for word m as

$$\mathbf{p}^m = \log\left[Pr(m|\Theta_{1m})Pr(m|\Theta_{2m}) \ldots Pr(m|\Theta_{Nm})\right]$$

with a general hypothesis function such as

$$h(\mathbf{x}) = \arg\max_{1 \leq i \leq |\mathbf{x}|} x_i. \tag{5}$$

With this, we can represent (2) as

$$D_m = h(\mathbf{p}^m).$$

Without loss of generality, the word label term m is dropped, so as to obtain the following relation:

$$D = h(\mathbf{p}). \tag{6}$$

If there is sufficient training data and the model is an accurate representation of the training data, (6) is the best decision strategy. However, there are usually limitations on the size of the training data and the representation ability of the model. Therefore, it would be useful to explore the classification information of the training samples more closely and compensate for errors in the original decision strategy in (6). Given the training samples $(\mathbf{p}_j, y_j)$, where $y_j \in \{1, 2, \ldots, N\}$ and $j = 1, 2, \ldots, T$, where T is the total number of training samples of word m across the N dialects, the AdaBoost algorithm [6], [33] can be applied to learn a sequence of "base" hypotheses (where each hypothesis h_t has a corresponding "vote power" α_t) to construct a classifier which we expect to be better than the single-hypothesis classifier in (6). By adjusting the weights of the training samples, each hypothesis h_t focuses on the samples misclassified by the previous hypotheses (i.e., the misclassified samples have larger weights than other samples). The final classifier is an ensemble of base hypotheses and is shown to decrease the classification error exponentially fast as long as each hypothesis has a classification error smaller than 50% [6]. The idea for applying AdaBoost on word dialect classification is illustrated as follows:

Given the entire data set $\mathcal{D} = \{(\mathbf{p}_j, y_j) | j = 1, 2, \ldots, T\}$, for simplicity, we consider the two-class case (i.e., $y_j \in \{-1, +1\}$), and note that the multi-label classification can be fulfilled using a sequence of pair-wise decision modes instead.

1) Initialize weights $w_j = (1/T), j = 1, 2, \ldots, T$.

2) For t $=$ 1 : n:

   a) Build a weak learner (a tree stump is used in our study) $h_t(\mathbf{p}) \in \{-1, +1\}$ using the data $\mathcal{D}$ weighted according to $w_j, j = 1, \ldots, T$. The information gain is used to build the tree stump. In essence, choose attribute $A = p_i, i = 1, 2, \ldots, N$, and the splitting threshold c that maximizes

$$\text{Gain}(S, A) {=} \text{Entropy}(S) - \sum_{v \in \{A \leq c, A > c\}} \frac{[S_v]}{[S]} \text{Entropy}(S_v)$$

   where $S {=} \{w_1 y_1, w_2 y_2, \ldots, w_T y_T\}$, $[S] = \sum_{j=1}^{T} w_j$, and $[S_v] = \sum_{m=1}^{T_v} w_{k(m)}$. Here, T_v is the number of samples in the set v, and $k(m)$ is the vector of indices to these samples. The split is conducted on every dimension of the vector $\mathbf{p}$, and the split which maximizes Gain(S, A) is kept. For each dimension i, the split value c is obtained by searching in the range of the values of p_i in a certain step size. The entropy is defined as

$$\text{Entropy}(S) = -\rho_+ \log \rho_+ - \rho_- \log \rho_-$$

   where

$$\rho_+ = \frac{\sum_{m=1}^{T_+} w_{d(m)}}{\sum_{j=1}^{T} w_j} \quad \rho_- = \frac{\sum_{m=1}^{T_-} w_{q(m)}}{\sum_{j=1}^{T} w_j}.$$

   Here, T_+ is the number of samples in S with $y_j > 0$, and $d(m)$ is a vector of indices to these samples; T_- is the number of samples in S with $y_j < 0$, and $q(m)$ is a vector of indices to these samples. The Entropy(S_v) is similarly defined.

   b) Compute the error $\epsilon_t {=} (\sum_{j=1}^{T} w_j \mathcal{I}(y_j \neq h_t(\mathbf{p}_j))) / (\sum_{j=1}^{T} w_j)$, stop and set $n = t - 1$ if $\epsilon_t \geq 0.5$.

   c) Compute $\alpha_t = \log((1 - \epsilon_t)/\epsilon_t)$.

   d) Update the weights, $w_j \leftarrow w_j \exp[\alpha_t \mathcal{I}(y_j \neq h_t(\mathbf{p}_j))]$.

3) The final boosted classifier $D_{W(k)}$ is

$$D_{W(k)} = D = \begin{cases} +1, & \text{if } \sum_{t=1}^{n} \alpha_t h_t(\mathbf{p}) \geq 0 \\ -1, & \text{else.} \end{cases} \tag{7}$$

Here, n is the number of iterations and is usually on the order of several hundred for convergence. This reflects another motivation for us to boost the classifier in the probability space instead of the feature space, which has been previously considered in [4], [5], and [26] for general speech recognition. The feature space-based boosting results in HMM training for each iteration, and is therefore computationally expensive.

C. Boosting Utterance Classifier D_u via Dialect Dependency

Individual words typically encode a nonuniform level of dialect-dependent information. Essentially, there are a variable levels of "decision power" in (3) across the words $W(k)$, $k = 1, 2, \ldots, K$. A new boosted version of the utterance classifier D_u from (3) can be formed as follows:

$$D_u = \arg\max_i \sum_{k=1}^{K} \mathcal{I}\left(D_{W(k)} = i\right) \cdot l_{W(k)\cdot i}, \quad i = 1, 2, \ldots, N \tag{8}$$

where $l_{W(k)\cdot i}$ is the measure of dialect dependency for word $W(k)$ in dialect i, which is defined as

$$l_i = \frac{1}{N-1} \sum_{j=1, j\neq i}^{N} \left\{ \frac{1}{T_i} \sum_{t=1}^{T_i} \log \frac{Pr(X_{it}|\mathrm{HMM}_i)}{Pr(X_{it}|\mathrm{HMM}_j)} + \frac{1}{T_j} \sum_{t=1}^{T_j} \log \frac{Pr(X_{jt}|\mathrm{HMM}_j)}{Pr(X_{jt}|\mathrm{HMM}_i)} \right\}. \tag{9}$$

For simplicity, the word label term $W(k)$ is dropped here, where T_i is the number of training samples of word $W(k)$ in dialect i; X_{it} is the tth training sample in dialect i, $i = 1, 2, \ldots, N$, and N is the number of dialects. This formulation is motivated by a measure of the model distance as discussed in [17] for general speech recognition. For our formulation, the larger the model distance, the greater the dialect dependency (i.e., the higher the vote power is for word $W(k)$ in utterance classifier D_u). We note that $l_{W(k)\cdot i}$ can be computed during the training stage, so there is no additional computational cost during evaluation.

D. Context Adaptive Training (CAT) on Small Data Set

If the size of the training set is small or there are many dialects for a limited-size training set, it becomes a challenge to train a robust HMM for each word and each dialect, and therefore model adaptation techniques should be applied. From Section III-A, we set the number of states in the word HMM equal to the number of phonemes contained in the word. Therefore, the word HMMs can be adapted from the phoneme models, which can be trained using data from all the dialects, or data that is independent of the dialect data set. The proposed adaptation scheme is motivated by the well established maximum likelihood linear regression (MLLR) [21] method. To begin with, we define the following notation:

T Total number of frames for a word in a dialect.

T_w Total number of training samples for a word in a dialect.

S Number of states (or phonemes) for a word in a dialect.

M Number of Gaussian mixtures for each state in the HMM.

$N_{i,j}$ Number of frames for the jth training sample in state i.

$\mathbf{o}_t$ tth observation vector, where the dimension of the feature is n.

$\mu_{s,m}$, $\Sigma_{s,m}$ Mean vector and the diagonal covariance matrix of the mth Gaussian mixture in the sth state, where $diag(\Sigma_{s,m}) = [\sigma^2_{s,m,1}\ \sigma^2_{s,m,2}\ \cdots\ \sigma^2_{s,m,n}]'$.

ω_m Mixture weight of the mth Gaussian mixture (in state s).

$a_{i,j}$ Transition probability from state i to state j.

π_i Initial probability of being in state i.

λ Entire parameter set of the word HMM for a particular word in a dialect.

$\mathbf{W}$ $n \times (n+1)$ transformation matrix which must be estimated in the MLLR method.

v, $\hat{v}$ Transformed pair, where v is the original variable, and $\hat{v}$ is the updated/estimated variable of v.

Using this notation, the mean vector in the HMM is updated through [21] as

$$\hat{\mu}_{s,m} = \mathbf{W}\xi_{s,m} \tag{10}$$

where $\xi_{s,m} = [1\ \mu'_{s,m}]'$, and

$$\mathbf{W}'_i = \left(\mathbf{G}^{(i)}\right)^{-1} \mathbf{Z}'_i \tag{11}$$

where $\mathbf{W}_i$ and $\mathbf{Z}_i$ are the ith row of $\mathbf{W}$ and $\mathbf{Z}$ ($\mathbf{Z}$ is an $n\times(n+1)$ matrix) respectively, and

$$\mathbf{Z} = \sum_{t=1}^{T}\sum_{s=1}^{S}\sum_{m=1}^{M} \gamma_{s,m}(\mathbf{o}_t)\Sigma^{-1}_{s,m}\mathbf{o}_t\xi'_{s,m} \tag{12}$$

$$\mathbf{G}^{(i)} = \left[g^{(i)}_{jq}\right]_{(n+1)\times(n+1)} \tag{13}$$

$$g^{(i)}_{jq} = \sum_{s=1}^{S}\sum_{m=1}^{M} v^{(s,m)}_{ii} d^{(s,m)}_{jq} \tag{14}$$

$$\mathbf{V}^{(s,m)} = \left[v^{(s,m)}_{ii}\right]_{n\times n} = \sum_{t=1}^{T} \gamma_{s,m}(\mathbf{o}_t)\Sigma^{-1}_{s,m} \tag{15}$$

$$\mathbf{D}^{(s,m)} = \left[d^{(s,m)}_{jq}\right]_{(n+1)\times(n+1)} = \xi_{s,m}\xi'_{s,m}. \tag{16}$$

Based on previous MLLR studies [7], we choose a diagonal covariance matrix for the update as follows:

$$\hat{\sigma}^2_{s,m,l} = R_l\sigma^2_{s,m,l}, \quad l = 1, 2, \ldots, n \tag{17}$$

where

$$R_l = \frac{\sum_{t=1}^{T}\sum_{s=1}^{S}\sum_{m=1}^{M}\gamma_{s,m}(\mathbf{o}_t)\left(\frac{(o_{t,l}-\hat{\mu}_{s,m,l})^2}{\sigma^2_{s,m,l}}\right)}{\sum_{t=1}^{T}\sum_{s=1}^{S}\sum_{m=1}^{M}\gamma_{s,m}(\mathbf{o}_t)}. \tag{18}$$

The term $\gamma_{s,m}(\mathbf{o}_t)$ denotes the probability of the tth frame being observed in the mth mixture of the sth state of the HMM. In the original formulation of MLLR, this term is computed through the forward–backward algorithm, whereas here the Viterbi algorithm is used. The term is defined as

$$\gamma_{s,m}(\mathbf{o}_t) = \begin{cases} P(m|\mathbf{o}_t, \lambda) = \frac{\omega_m b_m(\mathbf{o}_t)}{\sum_{k=1}^{M}\omega_k b_k(\mathbf{o}_t)}, & \text{if } S(\mathbf{o}_t) = s \\ 0, & \text{else} \end{cases} \tag{19}$$

where $S(\mathbf{o}_t)$ is the state which generates the frame $\mathbf{o}_t$, and $b_m(\mathbf{o}_t)$ is the probability of the tth observation vector being generated by the mth Gaussian mixture (in state s)

$$b_m(\mathbf{o}_t) = (2\pi)^{-n/2} \left(\Pi_{l=1}^{n} \sigma_{m,l}^2\right)^{-1/2} \times \exp\left[-\frac{1}{2}\sum_{l=1}^{n} \frac{(o_{t,l}-\mu_{m,l})^2}{\sigma_{m,l}^2}\right]. \quad (20)$$

Since the states of the word HMM are the phoneme sequence of the word obtained from a pronunciation dictionary (the CMU 125K American English dictionary [2] is used in our study), the HMM structure should be left-to-right. Also, using a one-state-skip structure will allow for single phoneme deletion (e.g., *farm* is pronounced as /F AA R M/ in the CMU dictionary; it is actually pronounced as /F AA M/ in British English). Since it is difficult to obtain a pronunciation dictionary which includes the pronunciation variations of all the dialects (further research could consider this), a phoneme recognizer may be applied to decode the phoneme sequence in order to capture the phoneme substitution, phoneme deletion and phoneme insertion characteristics. Therefore, we define three HMM structures in the CAT training. 1) CAT1-a: employs a no-skip left-to-right structure, where the phoneme sequence is obtained from the CMU pronunciation dictionary. 2) CAT1-b: employs a no-skip left-to-right structure, where the phoneme sequence is obtained from the phoneme recognizer. 3) CAT2: employs a one-state-skip left-to-right structure, where the phoneme sequence is obtained from the CMU pronunciation dictionary.

The steps employed for CAT training are summarized as follows.

1) Given the audio data and word-level transcripts, find the training samples for the words and phonemes using Viterbi forced alignment.
2) Train the universal (i.e., across all the dialects we work on) gender-dependent and/or gender-independent M mixture GMM for each phoneme using the entire corpus.
3) For each word and each dialect do the following:
   a) Initialize the word HMM. The corresponding S phoneme GMMs are concatenated to form an S state word HMM. The phoneme sequence of the word can be obtained by the pronunciation dictionary or by a phoneme recognizer. The initial state probabilities are set as

$$\pi_1 = 1,\ \pi_i = 0,\ i = 2, 3, \ldots, S. \quad (21)$$

If a no-skip left-to-right HMM structure (CAT1-a, CAT1-b) is used, the initial transition probabilities are set as follows,

$$\begin{cases} a_{i,i} = a_{i,i+1} = 1/2, & i = 1, 2, \ldots, S-1 \\ a_{i,i} = 1, & i = S \\ a_{i,j} = 0, & i = 1, 2, \ldots, S.j \notin \{i, i+1\}. \end{cases} \quad (22)$$

If a one-state-skip left-to-right HMM structure (CAT2) is used, the initial transition probabilities are set as follows:

$$\begin{cases} a_{i,i} = a_{i,i+1} = a_{i,i+2} = 1/3, & i = 1, 2, \ldots, S-2 \\ a_{i,i} = a_{i,i+1} = 1/2, & i = S-1 \\ a_{i,i} = 1, & i = S \\ a_{i,j} = 0, & i = 1, 2, \ldots, S. \\ & \quad j \notin \{i, i+1, i+2\}. \end{cases} \quad (23)$$

   b) Use Viterbi forced alignment to obtain the state and mixture sequences for each training sample.
   c) Update the HMM parameters as follows.
      i) Use (10) to update the Gaussian mixture mean vector $\mu_{s,m}$
      ii) Use (17) to update the Gaussian mixture diagonal covariance matrix $\Sigma_{s,m}$
      iii) The mixture weights are updated through

$$\hat{\omega}_{s,m} = \frac{1}{\sum_{j=1}^{T_w} N_{s,j}} \sum_{j=1}^{T_w} \sum_{k=1}^{N_{s,j}} P\left(m|\mathbf{o}_{j(k)}, \hat{\lambda}\right) \qquad s = 1, 2, \ldots, S,\ m = 1, 2, \ldots, M. \quad (24)$$

      iv) Here, three alternate methods are used for context adaptive training (CAT1-a, CAT1-b, and CAT2):

for CAT1-a and CAT1-b, the transition probabilities are updated through

$$\hat{a}_{i,i} = \frac{\sum_{j=1}^{T_w} N_{i,j}}{\sum_{j=1}^{T_w}(N_{i,j}+1)} \qquad \hat{a}_{i,i+1} = 1 - \hat{a}_{i,i}, \quad i = 1, 2, \ldots, S-1 \quad (25)$$

for CAT2, the transition probabilities are updated through

$$\begin{aligned} \hat{a}_{i,i} &= \frac{\sum_{j=1}^{T_w} N_{i,j}}{\sum_{j=1}^{T_w}(N_{i,j}+1)} \\ \hat{a}_{i,i+1} &= \frac{\sum_{j=1}^{T_w} \mathcal{I}(N_{i+1,j} \geq 1)}{\sum_{j=1}^{T_w}(N_{i,j}+1)} \\ \hat{a}_{i,i+2} &= 1 - \hat{a}_{i,i} - \hat{a}_{i,i+1}, \\ & \quad i = 1, 2, \ldots, S-2 \end{aligned}$$

and

$$\begin{aligned} \hat{a}_{S-1,S-1} &= \frac{\sum_{j=1}^{T_w} N_{S-1,j}}{\sum_{j=1}^{T_w}(N_{S-1,j}+1)} \\ \hat{a}_{S-1,S} &= 1 - \hat{a}_{S-1,S-1}. \end{aligned} \quad (26)$$

   d) Iterate between steps b) and c) until a preselected stopping iteration is reached or a model change threshold is achieved.

IV. Experiments

The speech recognizer used in our studies is the Sonic system [29], which employs a decision-tree triphone acoustic model

TABLE I
THREE USED CORPORA

Data	**Total Training Set**				**Total Test Set**			**Dialects/ Accents**
	Vocab.	**Spkrs**	**Size**	**style**	**Spkrs**	**Size**	**Style**	
WSJ	20K	375	40 hours	read	22	1 hour	read	2(American,British)
N4	1159	211	22 hours	read/ Spontaneous	31	43 minutes	read/ Spontaneous	4(British,Canadian, Dutch,German)
IViE	320	64	5 hours	read	32	86 minutes	Spontaneous	8 British dialects (Belfast, Bradford,Cambridge, Cardiff, Leeds, Liverpool, London, Newcastle)

and back-off trigram language model. The acoustic and language models were trained using the WSJ American English data, which are represented as the AM$_p$ in Fig. 3, and referred to as the "WSJ AM" and "WSJ LM" in Table IV. The feature representation used in our study consists of a 39-dimensional MFCC vector (static, delta, and double delta).

Three corpora containing dialect sensitive material are used for evaluation, which include the WSJ American and British English corpora (WSJ0 and WSJCAM0 [38], [44]), the NATO N4 foreign accent and dialect of English corpus [20], and the IViE British dialect corpus [15]. Table I shows a summary of training and test sets used from the corpora. The length of each test utterance is 9 s in duration for all the corpora. The WSJ and N4 data sets represent large-size/vocabulary corpora, which are used to test the basic WDC system from Section III-A, AdaBoost processing from Section III-B, and the dialect dependency approach from Section III-C. The IViE is a small-size/ vocabulary corpus, which is used to test CAT methods from Section III-D. The CAT approaches are not necessary for much larger dialect training sets because sufficient training data for word modeling is available.

A. Basic WDC Algorithms

There are two major phases in the WDC system: word modeling and word recognition. Tables II and III show word modeling information from the training stage (using the training data set shown in Table I) and word usage in the final decision stage respectively (using the test data set shown in Table I). Here, we define vocabulary coverage C_v and occurrence coverage C_o as follows: $C_v =$ (number of modeled words/ total number of distinct words), $C_o =$ (number of occurrences of modeled words/ total occurrences of all words). Table II shows that a small set of words accounts for a large portion of word occurrences in the training data, and only this small set of words is required for modeling (i.e., between 8% and 10% of the unique words account for 64%–75% of the words occurring in the audio). Table III shows that this observation is also true in the test data. In Table III, the word usage is the ratio of used words to the total number of words in the utterance; K is the average number of words used in the utterance. Throughout our experiments, the minimum number of used words for dialect classification of a test utterance is greater than five; the maximum number of used words for dialect classification of a test utterance is less than 40. Furthermore, since the language model $\overline{\mathrm{LM}}$ in Fig. 3 is intentionally applied to encourage the word recognizer to output words which have previously trained models, the word usage is high in the test data (Table III), and even higher than C_o in the training data (Table II) for the N4 corpus.

TABLE II
WORD MODELING INFORMATION OF WDC TRAINING

Data	Vocab.	Models	C_v	C_o
WSJ	20K	1642	8%	75%
N4	1159	115	10%	64%

TABLE III
WORD USAGE OF THE RECOGNIZED UTTERANCE IN THE FINAL DECISION STAGE

Data	Word Usage	K (in Eq. 3) per Utterance
WSJ	69%	15
N4	71%	17

Table IV shows how the acoustic and language model settings impact the word error rate (WER) and the dialect classification error rate of the basic WDC algorithm using the N4 corpus. The "WSJ AM" and "WSJ LM" are pretrained acoustic and language models from the Sonic system and are trained with the WSJ American English data. The "N4 AM" and "N4 LM" are trained with N4 training data (see Table I). "N4-BE," "N4-CA," "N4-GE," and "N4-NL" are the two dialects (British, Canadian) and two accents (German, Netherlands-Dutch) of the NATO N4 English corpus. The word error rate and utterance dialect classification error is obtained using the test data from N4 (see Table I). From Table IV, we find that the language model is much more important than acoustic model for dialect classification, and the dialect classification error is much smaller than the word error rate. Therefore, we place more attention on language model training; whereas for the acoustic model, we use a previously well-trained model (i.e., AM$_p$ in Fig. 3, Section III-A) for all the experiments in order to save effort for retraining acoustic models in a language. The WDC system does not require an exact match of word outputs from the word recognizer. For example, if the words "works," "fridge," and "litter" are recognized as "work," "bridge," and "letter" incorrectly (i.e., it is also the idea that the word recognizer is encouraged to output the common words which have previously trained word HMMs), we expect it will not cause a problem for the word classifier $D_{W(k)}$ in the WDC system (i.e., partial dialect dependent words are generally sufficient since the word encodes abundant dialect dependent information). Furthermore, since the WDC system is based on a majority vote of the word classifiers, it has sufficient tolerance for errors due to the word recognizer. We feel this represents a key reason why

TABLE IV
WER(%) AND UTTERANCE DIALECT CLASSIFICATION ERROR (%) OF WDC UNDER DIFFERENT AM/LM SETTINGS IN THE NATO N4 CORPUS

AM and LM Settings	Word Error Rate				Average WER	**Dialect Classification Error Rate of WDC**
	N4-BE	N4-CA	N4-GE	N4-NL		
WSJ AM, N4 LM	42.1	24.1	21.3	16.1	25	3.4
WSJ AM, WSJ LM	101.1	74.2	88.7	77.8	85	17.4
N4 AM, WSJ LM	80.1	72.0	53.7	70.4	70	9.2
N4 AM, N4 LM	14.0	21.4	7.4	13.1	14	2.6

TABLE V
ADABOOST APPLIED ON THE CORPORA

Data	Word Models	Boosted Models	Model Coverage	Occurrence Coverage
WSJ	1642	51	3%	44%
N4	115	7	6%	27%

TABLE VI
CLASSIFICATION ERROR(%) OF ALGORITHMS

Data	**LVCSR**	**WDC**	**WDC+ DD**	**WDC+ AB**	**WDC+ AB+DD**
WSJ	5.9	3.1	2.7	2.1	1.9
N4	5.5	3.4	1.9	3.0	1.6

the WDC system has only small requirements on the word recognizer during the evaluation. The first setting in Table IV (i.e., uses a previously well-trained task-independent acoustic model ("WSJ AM") and train a language model using task-specific data; this setting is shown in Fig. 3, Section III-A) is used in all the following experiments. Although it is not the best configuration, this setting can achieve reasonable performance without retraining acoustic models for each task in the same language which is required in the fourth setting.

From Tables II–IV, it is observed that the basic WDC algorithm can achieve good dialect classification performance with small requirements on the word recognizer (i.e., it can use a dialect independent acoustic model, and the WER can be high while achieving good dialect classification performance). We also note that a small number of word models (compared to the vocabulary size of the corpora) are sufficient for utterance dialect classification.

B. Performance of Boosted Word Classifier $D_{W(k)}$

In order to determine the proper number of iterations for AdaBoost, 75% of the original training data is randomly selected for AdaBoost training, and the remainder of the training data is used for validation. In order to obtain robust classifiers, only the words which have sufficient training samples (say, 500 in our study) are boosted. Table V shows the information of boosted models. The model coverage is defined as the ratio of the number of AdaBoosted models to the total number of word models. The occurrence coverage is defined as the ratio of the number of occurrences of AdaBoosted models to the total number of occurrences of word models in the original training data set. From Table V, we observe that the AdaBoosted word models account for a large portion of word occurrences. Therefore, the boosted word models will improve the performance of the utterance classifier even when the number of boosted word models is small.

Fig. 4 shows the error rate of AdaBoosted word classifier $D_{W(k)}$ in the newly partitioned WSJ training and validation sets. From Fig. 4, we observe that setting the number of iterations to $n = 2^7 = 128$ will be appropriate for AdaBoost, and the absolute word classification error reduction of AdaBoosted word models to the baseline word models is about 8%.

C. Evaluation on the WDC and Extensions

Section IV-B showed that the AdaBoost algorithm can boost the word classifier $D_{W(k)}$ significantly. Now, the boosted classifiers are applied to the basic WDC ($\mathrm{WDC} + \mathrm{AB}$, Section III-B). The dialect dependency can also "boost" the utterance classifier D_u ($\mathrm{WDC} + \mathrm{DD}$, Section III-C). The dialect dependency term l_i in (9) for all the word models are computed in the training stage, and used as a "vote power" term in the decision stage as shown in (8). $D_{W(k)}$ and D_u can be boosted simultaneously, and this configuration is referred to as $\mathrm{WDC} + \mathrm{AB} + \mathrm{DD}$. We note that there are no specific tuning parameters required in the WDC algorithm and its extensions. Finally, we employ the LVCSR-based approach as the baseline system for dialect classification. Table VI shows the utterance dialect classification error of the above algorithms using the test data summarized in Table I.

From Table VI, the basic WDC significantly outperforms the LVCSR-based system, which has been claimed to be the best performing system for language identification [41]. The WDC requires much less computation, especially in the evaluation stage since only one recognizer is used instead of N parallel recognizers. Next, the word classifier $D_{W(k)}$ is directly boosted by the AdaBoost algorithm in the probability space ($\mathrm{WDC} + \mathrm{AB}$), and the utterance classifier D_u is boosted by dialect dependency ($\mathrm{WDC} + \mathrm{DD}$), and these extensions to the basic WDC also show great performance improvement (see Table VI). Finally, combining both extensions results in a relative error rate reduction from the baseline LVCSR system to the "$\mathrm{WDC} + \mathrm{AB} + \mathrm{DD}$" for dialect classification of 67.8% for the WSJ corpus and 70.9% for the NATO N4 corpus. Since only a few word models in the N4 corpus are AdaBoosted (see Table V), the "$\mathrm{WDC} + \mathrm{AB}$" configuration for the NATO N4 corpus does not show the same level of improvement as experienced in the WSJ corpus.

In our study, the utterance boundaries are normally available. However, it would be interesting to explore performance for an on-the-fly condition (i.e., the utterance boundaries are unknown). Here, a previously formulated $\mathrm{T}^2 - \mathrm{BIC}$ [13], [42], [43] segmentation scheme is used to detect the boundaries. Table VII shows the classification errors for an on-the-fly condition using the same test data as in Table VI. The λ parameter used in the BIC is set to 1 in order to detect as many potential acoustic break points as possible (i.e., false alarm break points are therefore

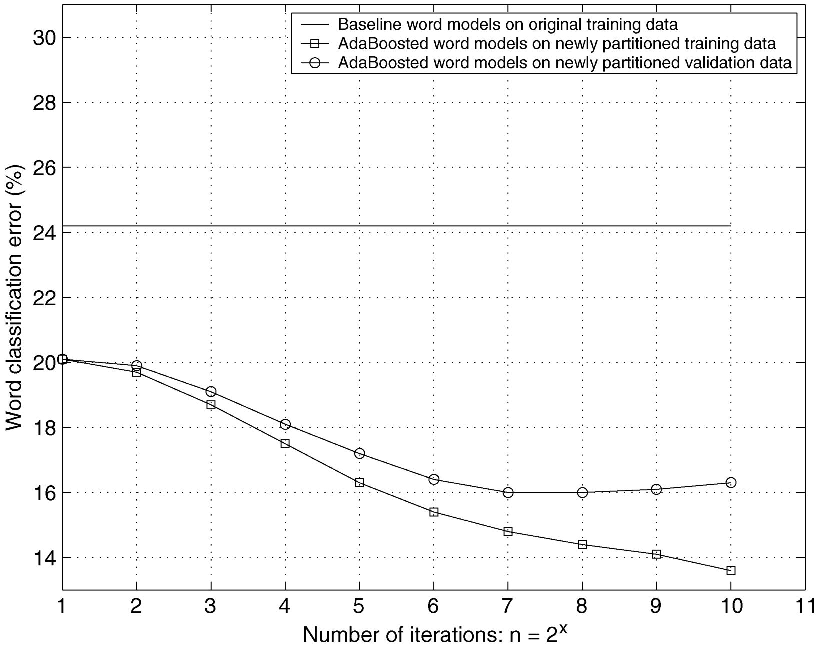


Fig. 4. Word dialect classification error versus number of $AdaBoost\ iterations(n = 2^x)$ in the newly partitioned WSJ training and validation sets.

TABLE VII
DIALECT CLASSIFICATION ERROR OF ALGORITHMS FOR AN ON-THE-FLY CONDITION

Data	**Segmentation**		**Dialect Classification Error Rate (%)**				
	False Alarm Rate	**Miss Rate**	**LVCSR**	**WDC**	**WDC+ DD**	**WDC+ AB**	**WDC+ AB+DD**
WSJ	9.1%	3.2%	10.8	7.8	6.8	6.2	5.3
N4	17.8%	5.6%	8.8	7.0	5.1	6.0	4.3

higher). The threshold for a correct break point is 2 s in duration. The average length of an utterance after $\mathrm{T}^2-\mathrm{BIC}$ segmentation is about 8 s in duration. From Table VII, it is observed that the $\mathrm{T}^2-\mathrm{BIC}$ algorithm can be applied in the on-the-fly condition, since the miss rate is quite low, even with an acceptable high false alarm rate. Compared with the LVCSR-based dialect classification approach, the "WDC + AB + DD" method achieves a relative dialect classification error rate reduction of 50.9% and 51.1% for WSJ corpus and NATO N4 corpus, respectively.

D. CAT on Small Size Data Set

From Table I, it is observed that there is on the average less than 40 min of training data for each dialect in the IViE corpus. Therefore, it is hard to train a robust HMM for each word of each dialect. The CAT algorithm is applied for this limited size corpus. For the baseline system, we originally implemented a similar PRLM system as in [40] since the LVCSR baseline system would not achieve very good performance due to the limited training data in the IViE corpus. However, the PRLM system could not distinguish three of the eight dialects at all, and the overall classification error was larger than 50%. Since the IViE training data is read speech, and the speakers in the eight dialects of the training data read basically similar documents, there is little dialect difference among the phoneme sequence. We believe this is probably why the PRLM system does not work well here. Therefore, we still apply the LVCSR based system as our baseline system.

As shown in Table I, there are 96 IViE speakers in total, where each speaker has produced both read and spontaneous speech. We use the read speech of 64 speakers as the training data. The read speech of the remaining 32 speakers is used to search for the best HMM topology for the word models, with the result shown in Fig. 5. The spontaneous speech of the remaining 32 speakers is used in the utterance dialect classification evaluation, with the result shown in Table VIII.

Fig. 5 shows the word dialect classification error of the three CAT structures versus the baseline WDC training algorithm for the eight-dialect IViE corpus. From Fig. 5, we see that all three CAT-based methods outperform the baseline WDC training algorithm significantly on the words with three-or-more phonemes, and the three CAT structures achieve almost the same performance for word classification. Since all eight dialects are from the U.K., there are not many differences across each dialect for phoneme deletion and phoneme insertion. If the dialects were from the U.K. and the U.S., we would expect more differences. This may be the reason why CAT1-b and CAT2 do not outperform the CAT1-a structure. Therefore, the CAT1-a is used in the following experiment.

Table VIII shows the utterance classification errors of different algorithms. "WDC + CAT" means that the word models are trained by the CAT algorithm (CAT1-a is used). "WDC" means the basic WDC algorithm as in Section III-A. The relative error reduction from the baseline LVCSR system to the "WDC + CAT" system is 35.5%. The AdaBoost algorithm in

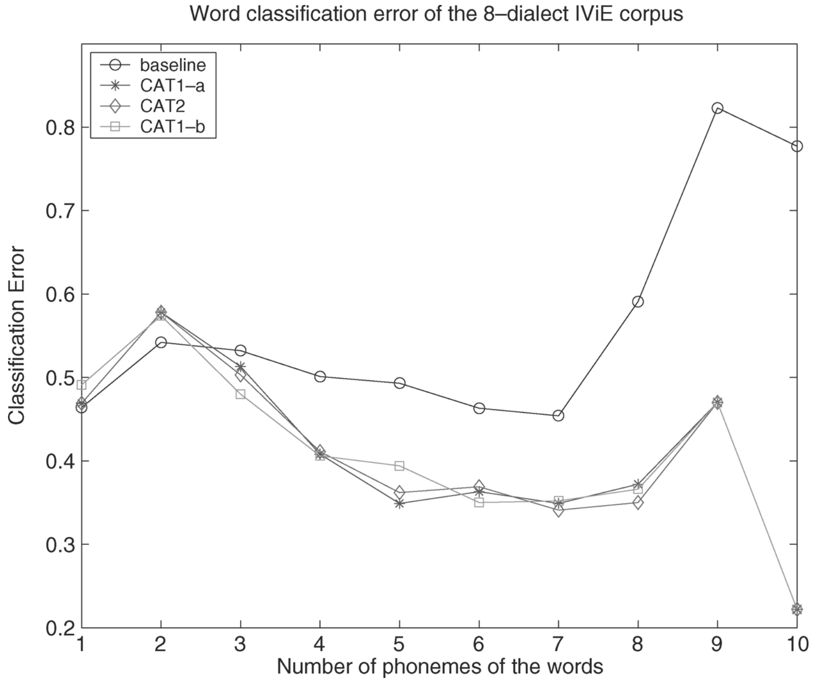


Fig. 5. The three CAT and baseline WDC word classification errors versus the number of phonemes in the word.

TABLE VIII
DIELECT CLASSIFICATION ERROR(%) OF ALGORITHMS ON EIGHT-DIALECT IViE CORPUS

LVCSR	WDC	WDC+ DD	WDC+ CAT	WDC+ CAT+DD
32.4	26.2	23.2	20.9	19.9

Section III-B requires a large amount of training samples, so it is not applicable here. However, the dialect dependency (DD) in Section III-C can still be applied in the limited size corpus. The "WDC + DD" reduces the absolute error rate by 3% from the "WDC" system. Furthermore, The dialect dependency information is calculated after the CAT word model training, so the "$\mathrm{WDC} + \mathrm{CAT} + \mathrm{DD}$" achieves further error rate reduction from the "WDC + CAT" system. The relative error reduction from the baseline LVCSR system to the "$\mathrm{WDC} + \mathrm{CAT} + \mathrm{DD}$" system is 38.6%.

Therefore, when only a small training corpus is available, the "$\mathrm{WDC}+\mathrm{CAT}+\mathrm{DD}$" system is able to provide effective dialect classification performance.

V. CONCLUSION

In this study, we have investigated a number of approaches for dialect classification. All the dialects considered showed great differences existing at the word level. An effective word-based dialect classification technique called WDC was proposed. A direct comparison between a LVCSR-based dialect classifier versus WDC shows that WDC achieves better performance with less computational requirements. The basic WDC algorithm also offers a number of areas for improvement that included modeling techniques and decision space extensions. The AdaBoost algorithm and dialect dependency are embedded into the word classifier $D_{W(k)}$ and utterance classifier D_u, respectively. Further dialect classification improvement is achieved with these extensions. The relative utterance dialect classification error reduction from the baseline LVCSR system to the "WDC + AB + DD" is 69.3% on the average. A CAT algorithm is formulated and shows promising performance when only a small size dialect corpus is available. The relative utterance dialect classification error reduction from the baseline LVCSR system to the "WDC + CAT + DD" is 38.6%. The "$\mathrm{WDC} + \mathrm{AB} + \mathrm{DD}$" system is therefore an effective approach for dialect classification when sufficient training data is available, and "$\mathrm{WDC} + \mathrm{CAT} + \mathrm{DD}$" is the preferred method when limited training data is available.

REFERENCES

[1] P. Angkititrakul and J. H. L. Hansen, "Use of trajectory model for automatic accent classification," in *Proc. EuroSpeech*, Geneva, Switzerland, Sep. 2003, pp. 1353–1356.

[2] *The CMU Pronunciation Dictionary*. Pittsburgh, PA: Carnegie Mellon Univ. [Online]. Available: http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[3] V. Diakoloukas, V. Digalakis, L. Neumeyer, and J. Kaja, "Development of dialect-specific speech recognizers using adaptation methods," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Munich, Germany, Apr. 1997, vol. 2, pp. 1455–1458.

[4] C. Dimitrakakis and S. Bengio, "Boosting HMMs with an application to speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Montreal, QC, Canada, May 2004, vol. 5, pp. 621–624.

[5] S. W. Foo and L. Dong, "A boosted multi-HMM classifier for recognition of visual speech elements," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, China, Apr. 2003, vol. 2, pp. 285–288.

[6] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997.

[7] M. J. F. Gales and P. C. Woodland, “Mean and variance adaptation within the MLLR framework,” in *Comput. Speech Lang.*, 1996, vol. 10, pp. 249–264.

[8] S. Greenberg, “On the origins of speech intelligibility in the real world,” in *Proc. ESCA Workshop on Robust Speech Recognition for Unknown Communication Channels*, Pont-a-Mousson, France, 1997, vol. 1, pp. 23–32.

[9] Q. Gu and T. Shibata, “Speaker and text independent language identification using predictive error histogram vectors,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, China, Apr. 2003, vol. 1, pp. 36–39.

[10] V. Gupta and P. Mermelstein, “Effect of speaker accent on the performance of a speaker-independent, isolated word recognizer,” *J. Acoust. Soc. Amer.*, vol. 71, pp. 1581–1587, 1982.

[11] J. L. Hieronymus and S. Kadambe, “Robust spoken language identification using large vocabulary speech recognition,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Munich, Germany, Apr. 1997, vol. 2, pp. 1111–1114.

[12] C. Huang, T. Chen, S. Li, E. Chang, and J. L. Zhou, “Analysis of speaker variability,” in *Proc. EuroSpeech*, Aalborg, Denmark, Sep. 2001, vol. 2, pp. 1377–1380.

[13] R. Huang and J. H. L. Hansen, “Advances in unsupervised audio segmentation for the broadcast news and NGSW corpora,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Montreal, QC, Canada, May 2004, vol. 1, pp. 741–744.

[14] J. J. Humphries and P. C. Woodland, “The use of accent-specific pronunciation dictionaries in acoustic model training,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Seattle, WA, May 1998, vol. 1, pp. 317–320.

[15] “IViE, British dialect corpus,” [Online]. Available: http://www.phon.ox.ac.uk/~esther/ivyweb/

[16] A. Sai Jayram, V. Ramasubramanian, and T. Sreenivas, “Language identification using parallel sub-word recognition,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, China, Apr. 2003, vol. 1, pp. 32–35.

[17] B.-H. Juang and L. R. Rabiner, “A probabilistic distance measure for hidden Markov models,” *AT&T Tech. J.*, vol. 64, no. 2, pp. 391–408, 1985.

[18] S. Kadambe and J. L. Hieronymus, “Language identification with phonological and lexical models,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Detroit, MI, May 1995, vol. 5, pp. 3507–3510.

[19] K. Kumpf and R. W. King, “Foreign speaker accent classification using phoneme-dependent accent discrimination models and comparisons with human perception benchmarks,” in *Proc. EuroSpeech*, Rhodos, Greece, Sep. 1997, vol. 4, pp. 2323–2326.

[20] A. Lawson, D. Harris, and J. Grieco, “Effect of foreign accent on speech recognition in the NATO N-4 corpus,” in *Proc. EuroSpeech*, Geneva, Switzerland, Sep. 2003, vol. 3, pp. 1505–1508.

[21] C. J. Leggetter and P. C. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models,” in *Comput. Speech Lang.*, 1995, vol. 9, pp. 171–185.

[22] M. Lincoln, S. Cox, and S. Ringland, “A comparison of two unsupervised approaches to accent identification,” in *Proc. Int. Conf. Spoken Language Processing*, Sydney, Australia, Nov. 1998, vol. 1, pp. 109–112.

[23] M. K. Liu, B. Xu, T. Y. Huang, Y. G. Deng, and C. R. Li, “Mandarin accent adaptation based on context-independent/context-dependent pronunciation modeling,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Istanbul, Turkey, Jun. 2000, vol. 2, pp. 1025–1028.

[24] D. Matrouf, M. Adda-Decker, L. F. Lamel, and J. L. Gauvain, “Language identification incorporating lexical information,” in *Proc. Int. Conf. Spoken Lang. Process.*, Sydney, Australia, Dec. 1998, vol. 1, pp. 181–185.

[25] S. Mendoma, L. Gillick, Y. Ito, S. Lowe, and M. Newman, “Automatic language identification using large vocabulary continuous speech recognition,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Atlanta, GA, May 1996, vol. 2, pp. 785–788.

[26] C. Meyer, “Utterance-level boosting of HMM speech recognizers,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Orlando, FL, May 2002, vol. 1, pp. 109–112.

[27] D. Miller and J. Trischitta, “Statistical dialect classification based on mean phonetic features,” in *Proc. Int. Conf. Spoken Lang. Process.*, Philadelphia, PA, Oct. 1996, vol. 4, pp. 2025–2027.

[28] S. Parandekar and K. Kirchhoff, “Multi-stream language identification using data-driven dependency selection,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, China, Apr. 2003, vol. 1, pp. 28–31.

[29] B. Pellom, “Sonic: The University of Colorado Continuous Speech Recognizer,” Univ. Colorado, Boulder, Tech. Rep. TR-CSLR-2001-01, Mar. 2001.

[30] T. Purnell, W. Idsardi, and J. Baugh, “Perceptual and phonetic experiments on American English dialect identification,” *J. Lang. Soc. Psychol.*, vol. 18, no. 1, pp. 10–30, Mar. 1999.

[31] P. Ramesh and E. Roe, “Language identification with embedded word models,” in *Proc. Int. Conf. Spoken Lang. Process.*, Yokohama, Japan, Sep. 1994, vol. 4, pp. 1887–1890.

[32] J.-L. Rouas, J. Farinas, F. Pellegrino, and R. Andre-Obrecht, “Modeling prosody for language identification on read and spontaneous speech,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, China, Apr. 2003, vol. 1, pp. 40–43.

[33] R. E. Schapire and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” *Mach. Learn.*, vol. 37, no. 3, pp. 297–336, 1999.

[34] T. Schultz, I. Rogina, and A. Waibel, “LVCSR-based language identification,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Atlanta, GA, May 1996, vol. 2, pp. 781–784.

[35] P. Trudgill, *The Dialects of England*, 2nd ed. Oxford, U.K.: Blackwell, 1999.

[36] W. Ward, H. Krech, X. Yu, K. Herold, G. Figgs, A. Ikeno, D. Jurafsky, and W. Byrne, “Lexicon adaptation for LVCSR: speaker idiosyncracies, non-native speakers, and pronunciation choice,” presented at the ISCA Workshop Pronunciation Modeling and Lexicon Adaptation, Estes Park, CO, Sep. 2002.

[37] J. C. Wells, *Accents of English*. Cambridge, U.K.: Cambridge University Press, 1982, vol. I, II, III.

[38] “WSJ0 corpus,” [Online]. Available: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC93S6A

[39] Q. Yan and S. Vaseghi, “Analysis, modeling and synthesis of formants of British, American and Australian accents,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, China, Apr. 2003, vol. 1, pp. 712–715.

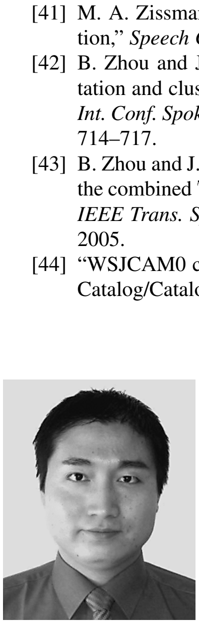
[40] M. A. Zissman, “Comparison of four approaches to automatic language identification of telephone speech,” *IEEE Trans. Speech Audio Process.*, vol. 4, no. 1, pp. 31–44, Jan. 1996.

[41] M. A. Zissman and K. M. Berkling, “Automatic language identification,” *Speech Commun.*, vol. 35, pp. 115–124, 2001.

[42] B. Zhou and J. H. L. Hansen, “Unsupervised audio stream segmentation and clustering via the Bayesian information criterion,” in *Proc. Int. Conf. Spoken Lang. Process.*, Beijing, China, Oct. 2000, vol. 1, pp. 714–717.

[43] B. Zhou and J. H. L. Hansen, “Efficient audio stream segmentation via the combined $\mathrm{T}^2-\mathrm{BIC}$ statistic and Bayesian information criterion,” *IEEE Trans. Speech Audio Process.*, vol. 13, no. 4, pp. 467–474, Jul. 2005.

[44] “WSJCAM0 corpus,” [Online]. Available: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC95S24



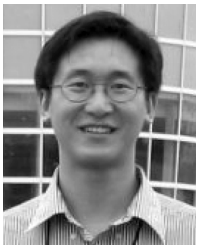
Rongqing Huang (S’01) was born in China in 1979. He received the B.S. degree from University of Science and Technology of China (USTC), Hefei, in 2002 and the M.S. and Ph.D. degrees from the University of Colorado, Boulder, in 2004 and 2006, respectively, all in electrical engineering.

From 2000 to 2002, he was with the USTC iFlyTek Speech Laboratory. From 2002 to 2005, he was with the Robust Speech Processing Group, Center for Spoken Language Research, University of Colorado. He was a Ph. D. Research Assistant in the Department of Electrical and Computer Engineering. In 2005, he was a summer intern with Motorola Labs, Schaumburg, IL. He was a Research Intern with the Center for Robust Speech Systems, University of Texas at Dallas, Richardson, in 2005 and 2006. In 2006, he joined Nuance Communications, Burlington, MA. His research interests include the general areas of spoken language processing, machine learning and data mining, signal processing, and multimedia information retrieval.

John H. L. Hansen (S'81–M'82–SM'93–F'06) received the B.S. degree in electrical engineering degree from Rutgers University, New Brunswick, NJ, in 1982 and the M.S. and Ph.D. degrees in electrical engineering from the Georgia Institute of Technology, Atlanta, in 1983 and 1988, respectively.

He joined the Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas (UTD), Richardson, in the fall of 2005, where he is Professor and Department Chairman of Electrical Engineering, and holds a Distinguished Chair in Telecommunications Engineering. He also holds a joint appointment in the School of Brain and Behavioral Sciences (Speech and Hearing). At UTD, he established the Center for Robust Speech Systems (CRSS) which is part of the Human Language Technology Research Institute. Previously, he served as Department Chairman and Professor in the Department of Speech, Language, and Hearing Sciences (SLHS), and Professor in the Department of Electrical and Computer Engineering, at the University of Colorado, Boulder (1998–2005), where he cofounded the Center for Spoken Language Research. In 1988, he established the Robust Speech Processing Laboratory (RSPL) and continues to direct research activities in CRSS at UTD. His research interests span the areas of digital speech processing, analysis and modeling of speech and speaker traits, speech enhancement, feature estimation in noise, robust speech recognition with emphasis on spoken document retrieval, and in-vehicle interactive systems for hands-free human-computer interaction. He has supervised 36 (18 Ph.D., 18 M.S.) thesis candidates. He is a coauthor of the textbook *Discrete-Time Processing of Speech Signals* (IEEE Press, 2000), coeditor of *DSP for In-Vehicle and Mobile Systems* (Springer, 2004), and lead author of the report "The Impact of Speech Under 'Stress' on Military Speech Technology," NATO RTO-TR-10, 2000.

Dr. Hansen is serving as IEEE Signal Processing Society Distinguished Lecturer for 2005/2006, is a Member of the IEEE Signal Processing Technical Committee, and has served as Technical Advisor to U.S. Delegate for NATO (IST/TG-01), Associate Editor for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING (1992–1999), Associate Editor for the IEEE SIGNAL PROCESSING LETTERS (1998–2000), and Editorial Board Member for the IEEE *Signal Processing Magazine* (2001–2003). He has also served as Guest Editor of the October 1994 special issue on Robust Speech Recognition for the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING. He has served on the Speech Communications Technical Committee for the Acoustical Society of America (2000–2003), and is serving as a member of the ISCA (International Speech Communications Association) Advisory Council (2004–2007). He was recipient of the 2005 University of Colorado Teacher Recognition Award as voted by the student body and author/coauthor of 222 journal and conference papers in the field of speech processing and communications. He also organized and served as General Chair for ICSLP-2002: International Conference on Spoken Language Processing, September 16–20, 2002, and will serve as Technical Program Chair for the IEEE ICASSP-2010.



Pongtep Angkititrakul (S'04–M'05) was born in Khonkaen, Thailand. He received the B.S. degree in electrical engineering from Chulalongkorn University, Bangkok, Thailand, in 1996 and the M.S. and Ph.D. degrees in electrical engineering from University of Colorado, Boulder, in 1999 and 2004, respectively.

From 2000 to 2004, he was a Research Assistant in the Robust Speech Processing Group, Center for Spoken Language Research (CSLR), University of Colorado. In February 2006, he joined the Center for Robust Speech Systems (CRSS), University of Texas at Dallas, Richardson, as a Research Associate. His research interests are in the general areas of robust speech/speaker recognition, pattern recognition, data mining, human–machine interaction, and speech processing.