An Experimental Study on Structural-MAP Approaches to Implementing Very Large Vocabulary Speech Recognition Systems for Real-World Tasks

I-Fan Chen[1], Sabato Marco Siniscalchi[1,2], Seokyong Moon[3], Daejin Shin[3], Myong-Wan Koo[4], Minhwa Chung[5], and Chin-Hui Lee[1]

[1] School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA. USA
E-mail: ichen8@gatech.edu, chl@ece.gatech.edu

[2] Università degli Studi di Enna "Kore," Cittadella Universitaria, 94100 Enna, ITALY
E-mail: marco.siniscalchi@unikore.it

[3] Infinity Telecom Co., Ltd, Seoul, REPULIC OF KOREA
E-mail: symoon@infinity.co.kr, djshin@infinity.co.kr

[4] Department of Computer Science and Engineering, Sogang University, Seoul, REPULIC OF KOREA
E-mail: mwkoo9@gmail.com

[5] Department of Linguistics, Seoul National University, Seoul, REPULIC OF KOREA
E-mail: mchung@snu.ac.kr

***Abstract* — In this paper we present an experimental study exploiting structural Bayesian adaptation for handling potential mismatches between training and test conditions for real-world applications to be realized in our multilingual very large vocabulary speech recognition (VLVSR) system project sponsored by MOTIE (The Ministry of Trade, Industry and Energy), Republic of Korea. The goal of the project is to construct a national-wide VLVSR cloud service platform for mobile applications. Besides system architecture design issues, at such a large scale, performance robustness problems, caused by mismatches in speakers, tasks, environments, and domains, etc., need to be taken into account very carefully as well. We decide to adopt adaptation, especially the structural MAP, techniques to reduce system accuracy degradation caused by these mismatches. Being part of an ongoing project, we describe how structural MAP approaches can be used for adaptation of both acoustic and language models for our VLVSR systems, and provide convincing experimental results to demonstrate how adaptation can be utilized to bridge the performance gap between the current state-of-the-art and deployable VLVSR systems.**

I. INTRODUCTION

At the dawn of the 21st Century the automatic speech recognition (ASR) community is at a crossroad. On the one hand, we have learned a great deal about how to build practical speech recognition systems for almost any spoken language without the need of a detailed understanding of the language. Data-driven, machine learning techniques, such as the *hidden Markov model* (HMM) [1] and *artificial neural network* (ANN) [2-4], are becoming so prevalent that numerous software packages and development kits have been implemented and made available to the public (e.g., [5]) to develop their own applications with ease. With the vast collections of speech and language corpora sponsored by many business and government-funded projects in many countries, it is now quite straightforward to demonstrate *automatic speech recognition* (ASR) capabilities of new tasks for almost any spoken language. Advances in hardware, algorithms and data structures have also made implementation of *large vocabulary continuous speech recognition* (LVCSR) systems affordable. On the other hand, these existing systems are often overly restrictive, requiring that their users have to follow a very strict set of protocols to effectively utilize spoken language applications. The technology is somewhat fragile in that careful designs have to be rigorously practiced to overcome technology deficiencies. Furthermore, the ASR accuracy often declines drastically in adverse conditions to an extent that some ASR systems become unusable, even for cooperative users. When compared with *human speech recognition*, or HSR, the state-of-the-art ASR systems usually give much larger error rates even for rather simple tasks operating in clean environments [6]. In highly noisy conditions, such as those in moving vehicle, ASR often gives an error rate more than one to two orders of magnitude higher than HSR. Such a performance gap is unacceptable to users and makes the work of application designers difficult.

A. Data-Driven Pattern Matching Paradigm

Fig. 1 shows a framework of state-of-the-art HMM-based LVCSR systems. The main idea behind this framework is to treat speech as a stochastic pattern and adopt a statistical pattern matching paradigm. We assume a source-channel speech generation model [7] in which the message source produces a sequence of words, W. Because of uncertainty and inaccuracy in converting from W to an observed speech signal, S, we model the process as a noisy channel. ASR is then formulated as a *maximum a posteriori* (MAP) decoding problem, such that

$$\hat{W} = \arg\max_W P(W \mid X) = \arg\max_W P(X \mid W)P(W). \qquad (1)$$

The recognized sentence is obtained by searching the set of all permissible sequences of words. $P(X|W)$, often referred to as an *acoustic model* (AM), is the conditional probability of a speech feature representation vector, X, for a given W. A

comprehensive review of and a critical look at acoustic modeling and its interactions with the MAP decision rule in Eq. (1) can be found in [8]. *P(W)*, the *a priori* probability of generating the sentence *W*, is known as a *language model* (LM) [9-12]. A *pronunciation model* (PM) is used to model lexical variation in speech [13, 14].

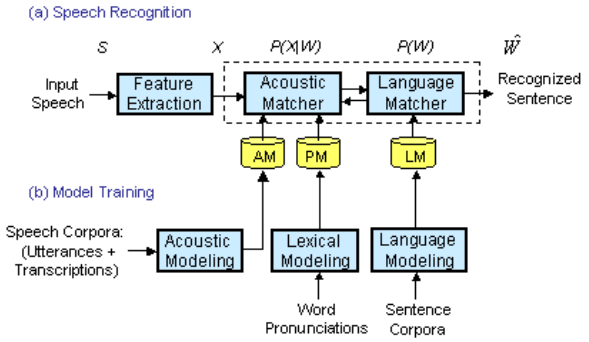

Fig. 1 State-of-the-art HMM-based LVCSR systems

Rather than establishing detailed models for the intermediate speech channels, including message-to-sentence generation, sentence-to-speech production, speaker variability, speaking environments, and transmission channels, the overall channel is collectively characterized in a *knowledge ignorance channel model*, *P(X|W)*. Since it is not feasible to have an exact and complete specification about such a noisy system, a statistical learning approach often assumes particular parametric forms for *P(X|W)* and *P(W)*. All model parameters needed to evaluate the acoustic and language probabilities are estimated from a collection of speech and text training corpora provided by a large population. The most widely used and successful modeling approach to ASR is the HMM [1]. Since an HMM is capable of jointly characterizing both the temporal and spectral character of the speech signal, many algorithms for model training have been proposed. These models mainly aim at two key objectives, namely: (1) model accuracy in estimating the feature distributions (e.g. [15-17]); and (2) model discrimination for minimizing the recognition error rate (e.g. [18, 19]).

B. The Robustness Problem

Based on our experience, speech is arguably one of the most difficult signals that we need to deal with. The inherent complexity of the speech production mechanism and the complicated interaction among speech, language and acoustics make it difficult to model the speech communication process exactly and analytically. Speech signals exhibit some measurable degree of intra- and inter-speaker variations. The transducer used to capture the signal, the channels in which the signal is transmitted, and the speaking environment. All of these factors can and do add distortion to the speech signal. As a result, ASR systems might work very well in one set of conditions, but then fail miserably in a slightly different situation. This so-called *robustness* problem, due to a possible mismatch between training and testing conditions, severely limits the widespread deployment of ASR applications and services. Compensation methods are often used to reduce these mismatches and consequently to improve ASR performance [20, 21]. These techniques include *feature compensation* [22], *parallel model combination* [23], *model adaptation* [8, 24, 25], *maximum likelihood linear regression* [26], *stochastic matching* [27, 28], *robust decision* [29], *uncertainty modeling* [30], etc. With so many possible combinations of speech variability factors, it is not possible to collect a large training set to cover every single condition and thereby train a set of focused models that works well in all cases. Therefore it seems clear that the robustness problem cannot be solved with only data-driven, top-down approaches. Some form of rapid adaptation of speech and language models to new conditions is definitely needed.

C. Proposed Research: Bayesian Adaptation of Speaker, Environmental, Task and Language Models

In the following we address issues related to static and adaptive modeling of speech and linguistic units and proposed a mathematical framework for performing MAP-based adaptive learning of the parameters of stochastic models commonly used in automatic speech recognition (ASR) and natural language processing (NLP). Furthermore, due to the limitation of adaptation data collection, we need to take into account of structural dependencies in acoustic HMM states and n-gram word relationship. This raises the need of a *structural MAP* (SMAP) framework to perform adaptation of all acoustic and language model parameters even only a small amount of adaptation speech and text data is available. We will show that the SMAP formulation takes into account the correlation between the observation data and the model parameters so that adaptive learning of model parameters can be accomplished even with a small amount of training data. When compared to the conventional ML algorithms, the proposed MAP algorithms are both efficient and effective for speaker and task adaptation. Similar MAP-based adaptive learning applications can also be derived for other sparse training data scenarios. Many recent techniques can also be combined with Bayesian approaches to improve adaptation efficiency and robustness.

II. BAYESIAN STRUCTURAL SPEECH MODEL

Since it is impossible to collect a large amount of training data covering every single condition for model training, an alternative to using a large training set is to use some initial set of sub-word unit models and adapt the models over time (with new training material, possibly derived from actual test utterances) to the task, the language, the speaker and/or the environment. These adaptive training techniques can be employed for new speakers, tasks and environments. It will be shown that adaptive training is an effective way of creating a good set of problem-specific models (adaptive models) from a more general set of models (which are speaker, environment, task, and context independent).

In the following sections, *maximum a posteriori* (MAP) estimation (e.g., [24, 31]) is used to effectively and efficiently

carry out adaptive training. In particular, the Bayesian learning principle [24] is adopted to derive MAP estimates of the parameters of some useful speech and language models. The prior density needed in the MAP formulation is specified based on prior knowledge embedded in a large collection of data or in a set of speech and language models. The Bayesian learning framework offers a theoretically-justified way to incorporate the newly acquired application-specific data into existing models and to combine them in an optimal manner. It is therefore an efficient and effective technique for handling the sparse training data problem, which is typical in adaptive learning of model parameters in real-world applications.

A. MAP-Based Bayesian Adaptation

In MAP formulation three key practical issues arise: (i) the definition of prior densities for the model parameters of interest, (ii) the estimation of the prior density parameters, sometimes referred to as hyperparameters, and (iii) the solution to MAP estimation. The three issues are related, and a good definition of the prior densities is crucial in resolving these issues. For acoustic modeling of speech units, continuous-variable observations are often characterized by multivariate Gaussian densities and gamma densities, while discrete-variable observations are often modeled by multinomial distributions. For example in hidden Markov modeling, all the above three densities from the exponential family have been combined to characterize the initial probabilities, the transition probabilities, the mixture gains for continuous density HMMs with mixture Gaussian state densities, the duration probability, etc. In most cases, the use of the *conjugate prior* formulation, such as a *Dirichlet* density for the estimation of multinomial parameters and a *normal-Wishart* density for the estimation of Gaussian parameters, has been found effective [24, 32].

In the last few years, Bayesian learning has been developed as a mathematical framework for obtaining MAP estimates of HMM parameters. For a given set of training/adaptation data $\mathbf{X}$, the conventional ML estimation assumes that the HMM parameter $\mathbf{\Lambda}$ is fixed but unknown and solves the following equation: $\mathbf{\Lambda}_{ML} = \arg\max_{\Lambda} f(\mathbf{X}|\mathbf{\Lambda})$, where $f(\mathbf{X}|\mathbf{\Lambda})$ is the likelihood of observing $\mathbf{X}$. On the other hand the MAP formulation assumes the parameter $\mathbf{\Lambda}$ to be a *random* vector with a certain distribution. Furthermore, there is an assumed correlation between the observation vectors and the parameters so that a statistical inference $\mathbf{\Lambda}$ can be made using a small set of adaptation data $\mathbf{Y}$. Before making any new observations, the parameter vector is assumed to have a prior density $g(\mathbf{\Lambda})$; when new data $\mathbf{Y}$ are incorporated, the parameter vector is characterized by a posterior density $f(\mathbf{\Lambda}|\mathbf{Y})$. Now the MAP estimate maximizes the posterior density, i.e. $\mathbf{\Lambda}_{MAP} = \arg\max_{\Lambda} f(\mathbf{\Lambda}|\mathbf{Y}) = \arg\max_{\Lambda} f(\mathbf{Y}|\mathbf{\Lambda})g(\mathbf{\Lambda})$.

The prior distribution characterizes the statistics of the parameters of interest before any measurement was made. It can be used to impose constraints on the values of the parameters. If the parameter is fixed but unknown and is to be estimated from the data, then there is no preference to what the value of the parameter should be. In such a case, the prior distribution is often called a non-informative prior which is a constant for the entire parameter region of interest. When the prior of the HMM parameters is assumed to be the product of the conjugate priors for all HMM parameters, the MAP estimates can be solved with the expectation-maximization (EM) algorithm [24, 31, 33]. A theoretical framework of MAP estimation of HMM was first proposed for estimating the mean and the covariance matrix parameters of a continuous density HMM (CDHMM) with a multivariate Gaussian state observation density. It was then extended to handle all the HMM parameters, including the initial state probabilities, the transition probabilities, the duration density probabilities, the energy histogram probabilities, and the state observation probabilities, of a CDHMM with mixture Gaussian state density [24]. The same Bayesian formulation has also been applied to the estimation of the parameters of discrete HMMs and of tied-mixture (or semi-continuous) HMMs [32].

B. Structural Acoustic Model Adaptation – SMAP

When the amount of data is sufficiently large, MAP estimation yields recognition performance as good as that obtained using maximum-likelihood (ML) estimation. However, while the amount of adaptation data is small, a *structural maximum a posteriori* (SMAP) approach can be instead used to improve the MAP estimates obtained. In the SMAP approach, a hierarchical structure in the model parameter space is assumed, and the probability density functions for model parameters at one level are used as priors for those of the parameters at adjacent levels.

Consider for a set G of all Gaussian mixture components in an acoustic model, we have a tree structure as shown in Fig. 2 where K is the total number of layers or the depth of the tree. Each node in the K-th layer (leaf node) corresponds to one Gaussian mixture component in the set of CDHMMs. The root node (the first layer) corresponds to the whole set G of the mixture components. Each intermediate node corresponds to a subset of G, and each of its subordinate leaf nodes corresponds to an element of a subset. At each node in the tree, a normalized probability density function (pdf), which is shared among the mixture components in the corresponding subset of G, is assigned. For each node N_k, at the p-th cluster of the k-th layer, the ML estimates of the probability density function (pdf) parameters, $\tilde{\mathbf{v}}_k^{(p)}$ and $\tilde{\mathbf{\eta}}_k^{(p)}$, are first evaluated [34].

In the tree structure in Fig. 2, one node sequence from the root to a leaf corresponds to all the predecessor nodes that must be traversed to reach a particular mixture component. For a particular m-th mixture component in G (we therefore omit the suffix identifying the mixture component except when doing so causes confusion), here we show how its parameters are estimated. The procedure described below is general and can be used to estimate the parameter sets of all the other mixture components in CDHMMs. The normalized pdf parameters are used to estimate the corresponding HMM

parameters by the two transformations: $\boldsymbol{\mu}_m^{(p)} = \boldsymbol{\mu}_m^{(p)} + \boldsymbol{\Sigma}_m^{1/2}\mathbf{v}_m^{(p)}$ and $\boldsymbol{\Sigma}_m^{(p)} = [\boldsymbol{\Sigma}_m^{(p)}]^{1/2}\boldsymbol{\eta}_m^{(p)}[(\boldsymbol{\Sigma}_m^{(p)})^{1/2}]^T$, with $\boldsymbol{\mu}_m^{(p)}$ and $\boldsymbol{\Sigma}_m^{(p)}$ being the updated ML estimate of the mean and covariance of the m-th component.

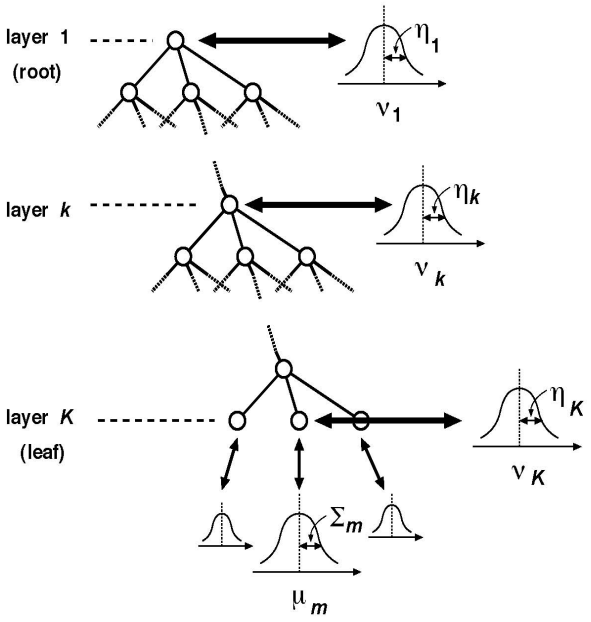

Fig. 2 Tree-based HMM state representations for SMAP

Let the node sequence from the root to the leaf corresponding to the m-th mixture component be $N_1, \ldots, N_K$ where N_1 is the root node and N_K is the leaf node directly attached to mixture component m. We denote $\boldsymbol{\lambda}_k = (\mathbf{v}_k, \boldsymbol{\eta}_k)$ as the Gaussian pdf parameters for node N_k. Now consider the problem of estimating, the parameter set $\boldsymbol{\lambda}_k$ that maximizes the posterior probabilities, after observing a sequence of feature vectors, $\mathbf{Y}$. It should be noted that once $\boldsymbol{\lambda}_K$ is obtained the parameter set can be obtained immediately by using the above transformations. In our approach, a set of priors $p(\boldsymbol{\lambda}_k \mid \hat{\boldsymbol{\lambda}}_{k-1})$ are used as *hierarchical priors* for estimating $\boldsymbol{\lambda}_K$, where $\boldsymbol{\lambda}_0$ is fixed to be $\hat{\boldsymbol{\lambda}}_0 = N(\mathbf{0}, \mathbf{I})$, with the pdf for node N_k with the parameter set $\boldsymbol{\lambda}_k$ has a hyperparameter $\hat{\boldsymbol{\lambda}}_{k-1}$ directly extended from its immediate parent node N_{k-1}. The posteriori density for $\boldsymbol{\lambda}_K$ can be approximated as follows [34]: $p(\boldsymbol{\lambda}_K) \approx \prod_{k=0}^{K-1} p(\boldsymbol{\lambda}_{k+1} \mid \hat{\boldsymbol{\lambda}}_k, \mathbf{Y})$.

The SMAP estimates for each node N_k can now be calculated as [34]:

$$\hat{\mathbf{v}}_K = \sum_{k=1}^{K} w_k \tilde{\mathbf{v}}_k = \sum_{k=1}^{K} \left[\frac{\Gamma_k}{\Gamma_k + \tau_k} \left(\prod_{i=k+1}^{K} \frac{\tau_i}{\Gamma_i + \tau_i} \right) \tilde{\mathbf{v}}_k \right],$$

where Γ_k is the data occupation of the node N_k and τ_k is a prior parameter for the node N_k needed to be specified.

The mean vector estimated using the SMAP method can be considered as a weighted sum of the ML estimates at the different layers of the tree. Two important characteristics of the weight, w_k, are highlighted: (1) the weight w_k at node N_k becomes larger as the amount of data at that node, Γ_k, becomes larger; and (2) the weight w_k at node N_k decreases as k becomes smaller.

These properties are desirable for adaptation. When the amount of data is small, the ML-estimated parameters in the upper layers are mainly responsible for the resulting pdf. And when the amount of data is large, the parameters in the lower layers predominate. On the other hand, the substantial number of the estimated parameters responsible for the resulting pdf increases as the amount of data increases. Because of the hierarchical structure in a tree, any amount of data can be used to adapt all the parameters of all Gaussian components in the leaf nodes of the tree.

III. Bayesian Structural Language Model

In addition to acoustic models, a language model to evaluate the sentence probability, *P(W)*, as shown in Fig. 1, is also a critical component in the state-of-the-art MAP decoding procedure for performing sentence-level matching. Therefore it is important to develop a rich and precise language model to achieve desirable speech recognition performance. Generally, the language model is represented by a statistical n-gram model, which calculates *P(W)* as

$$P(W) = P(w_1, \ldots, w_L) = \prod_{i=1}^{L} P(w_i \mid w_1, w_2, \ldots, w_{i-1}) \cong \prod_{i=1}^{L} P(w_i \mid w_{i-n+1}^{i-1}),$$

where L is the number of words in the sentence, W.

However, the n-gram model suffers from three kinds of insufficiencies: (i) amount of training data, (ii) domain knowledge, and (iii) long distance dependency. In literature, the issue of data sparseness was usually alleviated by applying parameter smoothing algorithms [35], e.g., Good-Turing smoothing, Witten-Bell smoothing, etc. For the issue of domain mismatch, it can be overcome to some extent by extracting updated domain knowledge from adaptation articles and merging the knowledge into the n-gram model. Language model adaptation algorithms were developed accordingly. Furthermore, in the issue of handling long distance dependency, we could apply latent semantic analysis and association pattern mining to find out relations between distant words for long distance language modeling. In the following section, we attempt to solve these three issues in a unified framework propose a hierarchical language model, i.e., *Bayesian structural language model* [36, 37].

A. Structural Language Modeling – SMAP-LM

In the proposed framework, the n-grams are assembled in tree structures where the nodes in the first layer represent the unigrams, and those in the second layer represent the bigrams, and so on (as shown in Fig. 3). The goal in building hierarchical models is to adaptively estimate the posterior probabilities of n-grams in a *top-down* manner. To do that, a *prior/posterior evolution* mechanism should be developed to propagate statistics of n-gram from root to leaf nodes of the tree. As we know, *Dirichlet* density serves as *conjugate prior*

to represent statistical behavior of n-gram $P(w_i \mid w_{i-n+1}^{i-1})$ as *multinomial* density. By selecting a Dirichlet prior for each n-gram in a LM, we can control model complexity from coarse to fine models, and the sparseness problem is thus tackled. To make the terminology consistent for tree nodes in different layers, we generalize the notations for structural n-gram model in Fig. 3. An additional superscript is used to denote the layer index.

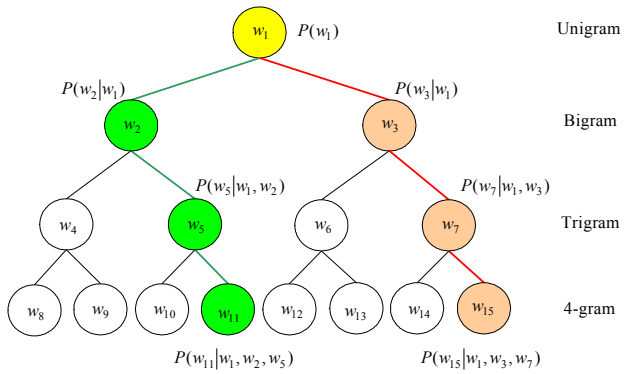

Fig. 3 Language model in different layers

Using the structural n-gram, we can specify the word probability for a tree node. Here, the *observation* probability of a word w_i in layer l is represented by

$$\theta_{hw_i}^{(l)} = P(w_i = w_i^{(l)} \mid h_i^{(l-1)} = \{w_i^{(1)}, w_i^{(2)}, \cdots, w_i^{(l-1)}\}) .$$

Histories, $h_i^{(l-1)}$, are located in the (l-1) layers of the parent nodes of the corresponding tree node and branch. We use *Dirichlet* density $g(P(w_i \mid h_i^{(l-1)})) = g(\theta_{hw_i}^{(l)} \mid \varphi_{hw_i}^{(l-1)}) \propto \theta_{hw_i}^{(l)\,\varphi_{hw_i}^{(l-1)}-1}$ to describe the prior density of *multinomial* observations. Given a training corpus $\mathbf{W}$ and an M-word dictionary, we are estimating *variable length* language model parameters for individual word given their historical words or a corresponding tree branch. Through *Bayesian learning*, language model parameters are estimated by maximizing the *posterior probability* accumulated for all tree nodes and branches. An exponential *forgetting factor* $0 \le \rho \le 1$ is incorporated to control the effect of the prior density in *MAP estimation* $\theta_{MAP} = \arg\max_{\theta} \log P(\mathbf{W} \mid \theta) + \rho \log g(\theta)$ at the individual tree node as [36]:

$$\theta_{hw_i,\mathrm{MAP}}^{(l)} = \frac{n(h_i^{(l-1)}, w_i) + \rho(\varphi_{hw_i}^{(l-1)} - 1)}{\sum_{w_j \in \Omega_w} [n(h_i^{(l-1)}, w_j) + \rho(\varphi_{hw_j}^{(l-1)} - 1)]} = \frac{n(h_i^{(l-1)}, w_i) + \rho(\varphi_{hw_i}^{(l-1)} - 1)}{n(h_i^{(l-1)}) + \sum_{w_j \in \Omega_w} \rho(\varphi_{hw_j}^{(l-1)} - 1)} .$$

In case of totally forgetting, ρ=0, it becomes an ML estimate

$$\theta_{hw_i,\mathrm{ML}}^{(l)} = \frac{n(h_i^{(l-1)}, w_i)}{n(h_i^{(l-1)})} .$$

We estimate the *hyperparameters* $\Phi = \{\varphi_{hw_i}^{(l)}\}$ of the tree nodes for Bayesian structural language modeling. Parallel to MAP estimation of LM parameters, we accumulate tree statistics from the root nodes to all leaf nodes. In a *top-down* manner, the hyperparameters $\varphi_{hw_i}^{(l)}$ of a tree node $w_i^{(l)}$ are inherited from those $\varphi_{hw_i}^{(l-1)}$ of its predecessor nodes along the tree branch $h_i^{(l-1)}$. However, it is crucial to develop a prior evolution mechanism of the posterior probability for finding structural parameters. *A posteriori probability* of an event $\{h_i^{(l-1)}, w_i\}$ ended in node $w_i^{(l)}$ is accumulated from the root node along the tree branch. Due to the property of *conjugate prior*, the posterior probability can be seen as a *prior probability* of the next layer nodes. The hyperparameter in layer l is propagated to estimation of the MAP estimate, $\theta_{hw_i,\mathrm{MAP}}^{(l)}$, in the successive layer in a *top-down* manner:

$$\theta_{hw_i,\mathrm{MAP}}^{(1)} \to \theta_{hw_i,\mathrm{MAP}}^{(2)} \to \cdots \to \theta_{hw_i,\mathrm{MAP}}^{(l)}, \text{ and}$$

$$\varphi_{hw_i}^{(1)} \to \varphi_{hw_i}^{(2)} \to \cdots \to \varphi_{hw_i}^{(l)},$$

where $\varphi_{hw_i}^{(l)} = n(h_i^{l-1}, w_i) + \gamma[\varphi_{hw_i}^{(l-1)} - 1] + 1$.

Hierarchical hyperparameters are established accordingly. In addition, we have to estimate the initial hyperparameters in the root nodes which represent the statistics of the individual words, $w_i^{(1)}$. We count the occurrences of the individual word, $w_i^{(1)}$, and adopt the initialization of the *Dirichlet* statistics as

$$\varphi_{hw_i}^{(1)} = 1 + \varepsilon \cdot n(w_i^{(1)}) .$$

Let us look at a physical meaning of the structural MAP language model, which is expressed in a *recursive* formula:

$$\theta_{hw_i,MAP}^{(l)} = \frac{n\left(h_i^{(l-1)}, w_i\right) + \rho\left(\varphi_{hw_i}^{(l-1)} - 1\right)}{n\left(h_i^{(l-1)}\right) + \sum_{w_j \in \Omega_w} \rho\left(\varphi_{hw_i}^{(l-1)} - 1\right)} .$$

Hyperparameters serve as *smoothing* factors coming from *(n-1)*-gram. Tracking of history words from the *low-order* to *high-order* n-grams is recorded in structure hyperparameters. The data sparseness problem is therefore relieved.

IV. Experimental Setup and Results

A. Experimental Setup

Experiments in this paper are conducted on two corpora collected by our MediaZen team: (i) English Twitter Speech corpus; and (ii) English Twitter Script corpus. The first English Twitter Speech corpus, which includes speech data of 20 speakers, was used for both acoustic and language model adaptation experiments. Each speaker in the corpus was asked to read a same set of 536 sentences, and the speeches were recorded under close talk clean condition at a sampling rate of 16KHz. The 536 short English sentences were collected from the English Twitter website. We selected 53 utterances out of the 536 sentences as test data for each speaker, while the rest 483 utterances were used for adaptation. The second English Twitter Script corpus is a pure text database, which includes 10,000 English sentences collected from the Twitters website. Among the total 10,000 sentences, 8,000 sentences were

selected as adaptation data for our language models, while the remaining 2000 sentences were used for test.

The vocabulary size of our LVCSR systems is about 200k[1]. We constructed the vocabulary with this size by combining vocabularies of the English Gigaword corpus (5th edition) [38] and Rovereto Twitter *N*-Gram Corpus [39]. For lexicon, the CMU Pronunciation Dictionary [40], which contains about 134k words, was used; for words not contained in the CMU dictionary, their pronunciations were estimated by methods proposed in [41]. The phoneme set used in our systems is the 39-phoneme CMU set; acoustic features were 13 MFCC features with its first and second derivatives.

Both acoustic and language model baselines were trained by the SNU team. Speech corpora, including TIMIT, WSJ, and WSJ1, with about 150 hours of speech, were used for baseline intraword triphone model training. State mixture numbers of the whole acoustic model were set to be 16 except for the silence model, whose number of Gaussian mixture in each state was 32. The total state number of our baseline acoustic model is 10,231 and the overall Gaussian mixture number is 163,744. For the baseline language model, the Rovereto Twitter *N*-Gram Corpus [39] was used for training. The resulting baseline language model contained 190,591 unigrams, 3,155,905 bigrams, and 12,781,158 trigrams.

B. Acoustic Model Adaptation

Experiments for acoustic model adaptation were conducted on the English Twitter Speech corpus. Since the number of the parameters of the baseline AM is rather large, we are interested in how to effectively use the limited adaptation data to achieve good system performance. Here, we compared the traditional MAP adaptation approach and the structural-MAP approach, which allows Gaussians with same predecessors in a hierarchical tree structure to share adaptation information among them [34]. An 8-layer binary tree with the prior parameter τ set to be 1 for all tree nodes was used as the SMAP tree structure in our experiments. Besides algorithms, we are also interested in comparing different adaptation scenarios, namely ways of adjusting AMs to a specific user/environment for real-world applications to achieve the best system performance. For example, an LVCSR system may improve its performance by collecting speech data from a user every time the system is being used and adapt the original speaker-independent AM towards a user-dependent AM when the amount of data is enough for AM adaptation. In the meantime, if a system knows the working environment of a new user, it may use speech data of other users collected in the same background conditions to adapt an AM for the specific condition so that the new user experiences a better system performance when beginning to use such a system. In this study, three adaptation scenarios are tested:

1. **MS-Unseen** (Multiple Unseen Speaker adaptation scenario) – the AM is adapted with speech recorded from speakers other than a current system user for channel and environment adaptation. This scenario is suitable for adjusting the system's original AM to a channel/environment-matched AM before a new user starts using the system, where the system does not have speech data from the user yet. Since the adaptation data are not required to be speaker dependent, the amount of adaptation data in this scenario is much larger than the data amount in speaker-dependent adaptation. In this paper, for each test speaker, the scenario is realized by adapting the baseline AM with the rest 19 speakers in the English Twitter Speech corpus.
2. **SD** (Speaker Dependent adaptation scenario) – here the AMs are adapted by users' own speech. This approach is suitable for system users who have been using the system for a while (and enough user speech data are collected). Since the adaptation data are users' own speech, which contain both working environment and pronunciation information of the speaker, the adapted AMs match the test conditions better than **MS-Unseen** adapted AMs, which are adapted to environment only. However, since adaptation data are speaker-dependent now, the amount of data is expected to be less than the **MS-Unseen** scenario and might not be enough for the adaptation module to adjust to a good AM for usage.
3. **MS-Unseen+SD** scenario – this is a hybrid approach combining the first and second adaptation scenarios above. Instead of doing **SD** adaptation on top of the original AMs, **SD** adaptation is applied on the **MS-Unseen** adapted models. Here AMs fully utilize all available adaptation data for both environment and speaker adaptation, and therefore better performance is expected. However, a drawback is the computation cost since two AM adaptation steps are required.

Table I lists performances of average word error rate (WER) over the 20 speakers for systems using MAP and SMAP with the three adaptation scenarios. Due to acoustic mismatches, from channels and speakers, the WER of the baseline system is still very high, as to 46.95%, although the baseline AM was trained with a great amount of speech data. Fortunately, the high WER can be reduced by AM adaptation. From the second column in Table I we can find that for conventional MAP adaptation all the three adaptation scenarios improved the system performance. Interestingly, though the **SD** scenario provides the most matched adaptation data for the test speaker, the performance after MAP adaptation with the **SD** scenario is slightly worse than that with the **MS-Unseen** scenario since the adaptation data size in the **MS-Unseen** scenario is 19 times of the amount in the **SD** scenario. This result reveals a drawback of the conventional MAP adaptation approach, i.e., it needs a great amount of data for adaptation. However this difficulty can be alleviated by using the **MS-Unseen+SD** adaptation scenario. Namely an AM is first adapted with a great amount of speaker-independent data to adjust parameters in the original AM to better fit the new environment, and then a small amount of speaker-dependent data is used for a second adaptation stage to further fine tune

[1] This number is about 10 times larger than what most of the LVCSR systems use today.

to a speaker-dependent AM. After two MAP adaptation steps, the system WER can be reduced from 46.95% to 28.33%.

On the other hand, knowing the relations between each Gaussian in the original AM, SMAP is able to utilize the adaptation data more efficiently through data sharing to adjust all parameters in the original AM. Experimental results in the third column of Table I show that the SMAP approach is significantly better than the MAP approach. With the **MS-Unseen** scenario, WER of the SMAP adapted system achieves 27.68%, which is already better than the best result of the MAP-adapted systems. Under the **SD** scenario, the system WER can be further improved, from 46.95% to 16.49%. Interestingly, when the test data is in clean condition, which matches the training condition of the baseline AM, SMAP adapted system with the **MS-Unseen+SD** scenario does not have a better performance than the system with the **SD** scenario. In fact, the performance of the **MS-Unseen+SD** system is slightly worse than the **SD** system for the SMAP approach. This is different from what we observed in the MAP-adapted systems. It can be explained by the fact that since SMAP is able to use the speaker-dependent data to adjust all parameters in the AM, and the training and test environments are similar, speech data from other speakers provide noisy observations for speaker adaptation instead of helpful information for environment adaptation.

TABLE I. WORD ERROR RATE (WER IN %) OF THE BASELINE SYSTEM AND SYSTEMS WITH THREE ADAPTATION SCENARIOS USING MAP AND SMAP ADAPTATION METHODS UNDER CLEAN SPEECH ENVIRONMENT.

	MAP	SMAP
Baseline (Unadapted)	46.95	
MS-Unseen	34.29	27.68
SD	34.41	16.49
MS-Unseen + SD	28.33	17.27

Table I shows the system performance in the clean speech conditions. However, in real-world applications, speech is often captured in noisy environments for most scenarios. To investigate various adaptation approaches to noisy conditions, we built Noisy English Speech corpora with 8 noise types (shown in the first columns of Table II, Table III, and Table IV) provided in the Aurora 2 corpus. Noise was added to the clean English Twitter speech data using the FaNT tool [42], used in the Aurora 2 corpus, with signal-to-noise ratio (SNR) values set up to 20 dB. The system performances in these conditions using the **MS-Unseen**, **SD**, and **MS-Unseen+SD** adaptation scenarios are listed in Table II, Table III, and Table IV, respectively.

It is noted that performances of the baseline system in noisy conditions are significantly worse than those in the clean condition due to potential model mismatches. In the airport environment, the WER of the baseline system increases to 50.71% from the original 46.95%; while for the subway background noise, WER significantly rose to 77.89%. An average 63.23% WER over the 8 noisy conditions is observed. A system with such high WER is useless for most applications and thus adaptation is needed. Here, in addition to channels and speakers, environmental background noises are also adaptation targets for AMs. Therefore, for MAP-adapted systems, using the **MS-Unseen** scenario, which provides more environmental information for AM parameter tuning, is better than using the **SD** adaptation scenarios in most cases. The average WER for MAP-adapted systems with the **MS-Unseen** scenario is 38.51% and the number is 40.25% for systems with **SD** adaptation scenario.

TABLE II. WER (IN %) OF SYSTEMS ADAPTED WITH **MS-UNSEEN** ADAPTATION SCENARIO IN NOISY CONDITIONS

	Baseline (Unadapted)	MAP	SMAP
Airport	50.71	31.16	27.15
Babble	68.62	38.51	32.44
Car	54.36	37.49	30.32
Exhibition	57.80	36.73	29.72
Restaurant	67.73	34.63	29.80
Street	66.79	43.78	36.58
Subway	77.89	46.28	36.75
Train	61.97	39.52	32.79
Average	63.23	38.51	31.95

On the other hand, comparing the third and fourth columns in Table II~IV, we can find that SMAP-adapted systems, regardless adaptation scenarios, outperform the MAP adapted systems. This shows that SMAP works not only for speaker adaptation but also for environment adaptation. When only information about noise is available, namely in the **MS-Unseen** scenario, the SMAP-adapted systems achieved a WER of 31.95% in average, while using data containing both environment and speaker information, as in the **SD** scenario, an average WER of the SMAP-adapted systems can further be reduced to 20.19% (shown in the last column of Table III).

TABLE III. WER (IN %) WITH **SD** ADAPTATION IN NOISY CONDITIONS

	Baseline (Unadapted)	MAP	SMAP
Airport	50.71	29.60	16.14
Babble	68.62	40.96	19.53
Car	54.36	40.05	19.14
Exhibition	57.80	37.15	19.46
Restaurant	67.73	34.10	18.32
Street	66.79	46.68	23.49
Subway	77.89	51.88	24.98
Train	61.97	41.62	20.46
Average	63.23	40.25	20.19

The system performance can be further improved by using the **MS-Unseen+SD** adaptation scenario. This is shown in Table IV for the 8 noisy conditions. Similar to those observed in the clean speech experiments, MAP-adapted systems are benefited from using this scenario. The average WER of the MAP-adapted systems is 30.11%. For SMAP in noisy conditions, unlike what we observed in the clean speech experiments, systems adapted with the **MS-Unseen+SD** scenario are better than systems with the **SD** scenario. The average WER of the SMAP-adapted systems with the **MS-Unseen+SD** scenario is 18.47% comparing to 20.19% with the **SD** scenario. This trend is consistent among all types of noises. The trend difference between clean and noisy speech conditions for SMAP can be explained by the scale of environment mismatch for the baseline AM in these two test conditions. Since environment mismatches contribute a larger portion of the acoustic disparities between the original clean-

trained AM and the test speech than speaker mismatches, combining the **SD** scenario with the **MS-Unseen** scenario, which provides more information of the noise distributions, helps the baseline system to alleviate the mismatch problems more than those with the **SD** scenario alone.

TABLE IV. WER (IN %) OF SYSTEMS ADAPTED WITH **MS-UNSEEN+SD** ADAPTATION SCENARIO IN NOISY CONDITIONS

	Baseline (Unadapted)	MAP	SMAP
Airport	50.71	23.80	15.23
Babble	68.62	30.10	18.92
Car	54.36	30.20	18.16
Exhibition	57.80	28.13	17.48
Restaurant	67.73	26.34	16.38
Street	66.79	35.43	21.39
Subway	77.89	35.98	21.77
Train	61.97	30.91	18.40
Average	63.23	30.11	18.47

It is clear that the **MS-Unseen+SD** adaptation scenario significantly helps MAP-adapted systems in both the clean and noisy conditions. However, the performance differences between SMAP-adapted systems with the **SD** and **MS-Unseen+SD** scenarios are relatively small. To verify if the differences are signification, we ran a set of significance tests for these two scenarios. Table V shows the p-values [43] of the significance tests for systems using these two scenarios in the 9 background noise conditions. A lower p-value indicates a more significant difference. In Table V, the p-values in bold font indicate more significant differences between these two scenarios at the 0.01 significance level. It is clear that for the conventional MAP-adapted **MS-Unseen+SD** scenario is significantly better than the **SD** scenario in all conditions.

On the other hand, though the p-values for the SMAP tests are not as low as the MAP-adapted systems, the differences between these two scenarios are significant for most of the conditions at the 0.05 significance level (except the Babble condition). Especially for noises like exhibition, street, subway, and train, advantages of using the **MS-Unseen+SD** scenario is more significant than using the **SD** scenario alone. Notice that these four noises are less stationary than the rest of the noises presented here, thus SMAP needs more data for parameter adjustment. For clean condition, we find that the difference between the **MS-Unseen+SD** and **SD** scenarios is also significant at the 0.05 significance level. It means the SMAP-adapted system with the **MS-Unseen+SD** scenario (WER 17.27%) is significantly worse than using the **SD** scenario (WER 16.49%) alone.

TABLE V. SIGNIFICANCE TEST OF THE **MS-UNSEEN+SD** VS. **SD** SCENARIO.

	MAP MS+SD vs. SD p-value	SMAP MS+SD vs. SD p-value
Clean	**2.62738E-05**	0.019
Airport	**1.68909E-06**	0.047
Babble	**1.95703E-07**	0.075
Car	**7.84328E-07**	0.035
Exhibition	**5.06452E-07**	**0.002**
Restaurant	**1.11135E-05**	0.015
Street	**3.20601E-08**	**0.0004**
Subway	**1.05398E-08**	**0.002**
Train	**6.23774E-08**	**0.002**

In summary, SMAP adaptation significantly outperforms conventional MAP adaptation in all test environments. Thus SMAP is a better choice than MAP for real-world systems. In addition, to achieve a best system performance using SMAP, a system designer should be aware to building an SMAP-adapted system with the **SD** scenario in clean conditions. While in noisy test conditions, the SMAP systems should be adapted with the **MS-Unseen+SD** scenario in order to achieve a better system performance.

C. Language Model Adaptation

In the above AM adaptation experiments, LMs used in the adapted systems are the baseline LM. However, in real-world applications, topics and contents of users' speech may change from time to time. If a system is able to realize the topic a system user is currently on and adjust its LM towards the topic, it may achieve a further performance improvement after AM adaptation. In this section, we test different LM adaptation methods, including using an adaptation-data-trained LM directly, interpolating probabilities of the baseline LM with an adaptation-data-trained LM, and the proposed SMAP-LM adaptation. To measure the similarity between probability distributions of terms estimated by an LM and the distributions of real test data, we adopt perplexity (PPL) for the performance evaluation. The lower the PPL implies the closer the two distributions, i.e., the adapted-LM has a higher chance to correctly *guess* the next word in an unfinished sentence given words seen so far. Therefore, we can expect an LVCSR system using this LM for processing utterances on this topic to have a better performance. Empirically, an LM which supports good recognition performance for an LVCSR system has a PPL value below 100.

We ran adaptation experiments on both the English Twitter Speech and English Twitter Script corpora. For the English Twitter speech corpus, we used the same 483 sentences used in AM adaptation experiments for LM adaptation while the rest 53 sentences were used as test data. There were no out-of-vocabulary (OOV) words in the English Twitter Speech corpus for our baseline system. In the adaptation-data-trained LM, Good-Turing smoothing was used. All parameters, including weights for the baseline LM in the LM interpolation method and hyperparameters used in SMAP-LM, were estimated by cross-validation on the adaptation data. The PPL results on the English Twitter Speech corpus are shown in Table VI.

TABLE VI. PERPLEXITIES OF DIFFERENT LANGUAGE MODELS ON ENGLISH TWITTER SPEECH CORPUS TEST TRANSCRIPTIONS.

	Perplexity
Baseline LM	204.54
Adaptation Text trained LM	1020.23
Interpolated -LM w=0.5	99.23
SMAP-LM ($\varepsilon = 0.1,\ \gamma = 0.0001,\ \rho = 0$)	56.95

In Table VI we can find the PPL of the baseline LM is 204.54. Since the amount of adaptation data in the English Twitter Speech corpus for LM is quite small (only 483 sentences), many terms in the test data are unseen to the

adaptation-data-trained LM. Therefore, the PPL of this LM at 1020.23 was really high. However, if we interpolate this LM with the baseline LM, the PPL of the resulting LM is 99.23, which is reasonably good for an LVCSR system. Finally, if we applied SMAP-LM adaptation to the baseline LM, the adapted-LM achieved a PPL of 56.95. Note that the parameter ρ used in SMAP-LM being equal to zero means if a term can be found in the adaptation data, then the adapted-LM used probabilities estimated from the adaptation data for that term. While if a term could not be found in the adaptation data, the smoothed baseline LM probability for the term was used.

A similar trend can be found on the English Twitter Script corpus, which contains much more sentences. In this corpus, there are 117 OOV words for the baseline LM. Table VII shows the PPL of LMs adapted with different approaches. The PPL of the baseline LM on the test data was 250. With more adaptation data, the adaptation-data-trained LM has a much lower PPL when compared to the case in the English Twitter Speech corpus. However, it is still higher than the baseline LM's PPL, at 418. If these two LMs are interpolated, the resulting LM has a PPL of 125, which is smaller than the PPL of the baseline LM. The best LM is still the SMAP-LM-adapted LM, which achieves 58.27 in PPL.

TABLE VII. PPLS OF LMS ON THE ENGLISH TWITTER SCRIPT TEST SENTENCES.

	Perplexity
Baseline LM	250
Adaptation Text trained LM	418
Interpolated LM w=0.5	125
SMAP-LM ($\varepsilon = 0.1,\ \gamma = 0.0001,\ \rho = 0$)	58.27

V. CONCLUSIONS

In this paper, we present a set of approaches to exploit structural-MAP techniques for AM and LM adaptation to alleviate some real-world robustness problems in VLVSR systems caused by potential mismatches in speakers, speaking environments and application domains. Experimental results show that Bayesian adaptation outperforms systems without adaptation. Moreover, it was also found that structural-MAP approaches significantly outperform the conventional MAP-based methods.

For AM adaptation, three adaptation scenarios, namely **MS-Unseen**, **SD**, and **MS-Unseen+SD** adaptation each corresponds to a standard operational procedure for a VLVSR system, are also considered. Results shows that in clean test conditions, where the training and test environments are mostly matched, using SMAP with the **SD** scenario leads a system toward the best performance, and the WER 46.95% of the baseline system was reduced to 16.49%; while in noisy conditions, using SMAP with the **MS-Unseen+SD** scenario improves system performance the most, from a WER of 63.23% for the baseline system to 18.47%.

For LM adaptation, experiments on the two tasks show the SMAP-LM approach also outperforms other commonly used adaptation methods, such as LM interpolation. Moreover while the baseline LM has a PPL higher than 200, the SMAP-adapted LMs achieve a PPL at about 50 for both experimental tasks. These results show that the SMAP family of techniques offers convincing algorithms for tackling the robustness problems in real-world tasks.

For future personalization needs in ASR, online adaptation algorithms are readily available (e.g., [25, 44]). Stochastic matching (e.g., [27, 28]) and unsupervised SMAP [45] also provide an online mechanism for enhancing robustness. Utterance verification (e.g., [46-49]) is also a critical area of technology to provide confidence measures to accept speech recognition hypotheses when deploying field applications.

ACKNOWLEDGEMENT

This research was supported by MOTIE (The Ministry of Trade, Industry and Energy), Republic of Korea, under the Global collaborative R&D Program (2010-TD-300802-003) supervised by KIAT (Korea Institute for Advancement of Technology). The authors also thank their colleagues at Korea Telecom for sharing their decoder which was used to carry out most of the VLVSR experiments. We also appreciate the effort of MediaZen to provide all the speech and language corpora used in this study.

REFERENCE

[1] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. IEEE,* vol. 77, pp. 257-286, Feb. 1989.

[2] S. Haykin, *Neural Networks: A Comprehensive Foundation*: McMillan, 1994.

[3] H. Bourlard and N. Morgan, *Connectionist Speech Recognition – A Hybrid Approach*: Kluwer Academic Press, 1994.

[4] S. Katagiri, Ed., *Handbook of Neural Network for Speech Processing*. Artech House Publisher, 2000.

[5] S. J. Young, G. Evermann, M. J. F. Gales, D. Kershaw, G. Moore, J. J. Odell*, et al.* (2006). *The HTK book version 3.4*.

[6] R. Lippmann, "Speech Recognition by Human and Machines," *Speech Communication,* vol. 22, pp. 1-14, 1997.

[7] L. R. Bahl, F. Jelinek, and R. L. Mercer, "A Maximum Likelihood Approach to Continuous Speech Recognition," *IEEE Trans. Pattern Analysis, Machine Intelligence,* vol. 5, pp. 179-190, 1983.

[8] C.-H. Lee and Q. Huo, "On Adaptive Decision Rules and Decision Parameter Adaptation for Automatic Speech Recognition," *Proceedings of the IEEE,* vol. 88, pp. 1241-1269, 2000.

[9] S. M. Katz, "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer," *IEEE Trans. Acoustic, Speech and Signal Proc.,* vol. 35, pp. 400-401, 1987.

[10] L. R. Bahl, P. F. Brown, P. V. D. Souza, and R. L. Mercer, "Tree-Based Language Model for Natural Language Speech Recognition," *IEEE Trans. Acoustic, Speech and Signal Proc.,* vol. 37, pp. 1001-1008, 1989.

[11] R. Rosenfeld, "Two Decades of Statistical Language Modeling: Where Do We Go from Here?," *Proc. IEEE,* vol. 88, pp. 1270-1278, 2000.

[12] J. R. Bellegarda, "Exploiting Latent Semantic Information for Statistical Language Modeling," *Proc. IEEE,* vol. 88, pp. 1279-1296, 2000.

[13] M. D. Riley, "A Statistical Model for Generating Pronunciation Networks," in *Proc. ICASSP*, 1991, pp. 737-740.

[14] J. E. Fosler-Lussier, "Dynamic Pronunciation Models for Automatic Speech Recognition," Ph.D Dessertation, University of California, Berkeley, 1999.

[15] B.-H. Juang, S. E. Levinson, and M. M. Sondhi, "Maximum Likelihood Estimation for Multivariate Mixture Observations of Markov Chains," *IEEE Trans. Information Theory,* vol. IT-32, pp. 307-309, 1986.

[16] M. Ostendorf and S. Roukos, "A Stochastic Segment Model for Phoneme-Based Continuous Speech Recognition," *IEEE Trans. Acoustic, Speech and Signal Proc.,* vol. 37, p. 1857, 1989.

[17] C.-H. Lee, L. R. Rabiner, R. Pieraccini, and J. G. Wilpon, "Acoustic Modeling for Large Vocabulary Speech Recognition," *Computer, Speech and Langauge,* vol. 4, pp. 127-165, April 1990.

[18] B.-H. Juang, W. Chou, and C.-H. Lee, "Discriminative Methods for Speech Recognition," *IEEE Trans. Speech and Audio Proc.,* vol. 5, pp. 257-265, May 1997.

[19] S. Katagiri, B.-H. Juang, and C.-H. Lee, "Pattern Recognition Using A Generalized Probabilistic Descent Method," *Proc. IEEE,* vol. 86, pp. 2345-2373, 1998.

[20] C.-H. Lee, "On stochastic feature and model compensation approaches to robust speech recognition," *Speech Communication,* vol. 25, pp. 29-47, August 1998.

[21] J.-C. Junqua and J.-P. Haton, *Robustness in Automatic Speech Recognition: Fundamentals and Applications*. Boston: Kluwer Academic, 1996.

[22] A. Acero, *Acoustical and Environmental Robustness in Automatic Speech Recognition*: Kluwer Academic, 1992.

[23] M. J. F. Gales and S. J. Young, "Parallel Model Combination for Speech Recognition in Noise," CUED/ TR135, 1993.

[24] J.-L. Gauvain and C.-H. Lee, "Maximum A Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chains," *IEEE Trans. Speech and Audio Proc.,* vol. 2, pp. 291-298, April 1994.

[25] Q. Huo and C.-H. Lee, "On-line Adaptive Learning of the Continuous Density Hidden Markov Model Based on Approximate Recursive Bayes Estimate," *IEEE Trans. Speech and Audio Proc.,* vol. 5, pp. 161-172, March 1997.

[26] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language,* vol. 9, pp. 171-185, 1995.

[27] A. Sankar and C.-H. Lee, "A Maximum-Likelihood Approach to Stochastic Matching for Robust Speech Recognition," *IEEE Trans. Speech and Audio Proc.,* vol. 4, pp. 190-202, May 1996.

[28] A. C. Surendran, C.-H. Lee, and M. Rahim, "Non-Linear Compensation for Stochastic Matching," *IEEE Trans. Speech and Audio Proc.,* vol. 7, pp. 643-655, 1999.

[29] Q. Huo and C.-H. Lee, "Robust Speech Recognition Based on Adaptive Classification and Decision Strategies," *Speech Communication,* vol. 34, pp. 175-194, 2001.

[30] J. A. Arrowood, "Using Observation Uncertainty in HMM Decoding," in *Proc. ICSLP-2002*, Boulder CO, 2002.

[31] C.-H. Lee, C.-H. Lin, and B.-H. Juang, "A Study on Speaker Adaptation of the Parameters of Continuous Density Hidden Markov Models," *IEEE Trans. Acoustic, Speech and Signal Proc.,* vol. ASSP-39, pp. 806-814, April 1991.

[32] Q. Huo, C. Chan, and C.-H. Lee, "Bayesian Adaptive Learning of the Parameters of Hidden Markov Model for Speech Recognition," *IEEE Trans. on Speech and Audio Proc.,* vol. 3, pp. 334-345, Sept. 1995.

[33] J.-L. Gauvain and C.-H. Lee, "Bayesian Learning for Hidden Markov Model with Gaussian Mixture State Observation Densities," *Speech Communication,* vol. 11, pp. 205-213, June 1992.

[34] K. Shinoda and C.-H. Lee, "A Structural Bayes Approach to Speaker Adaptation," *IEEE Trans. Speech and Audio Proc.,* vol. 9, pp. 276-287, March 2001.

[35] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proc. of the 34th annual meeting on Association for Computational Linguistics*, 1996, pp. 310-318.

[36] S. Yaman, J.-T. Chien, and C.-H. Lee, "Structural Bayesian Language Modeling and Adaptation," in *Proc. Interspeech*, Antwerp, Belgium, 2007.

[37] S. Yaman, S. M. Siniscalchi, and C.-H. Lee, "A Multi-Objective Programming Based Approach to Language Model Adaptation," in *Proc. Workshop on Deep Learning for Speech Recognition and Related Applications*, Whistler, BC, Canada, 2009.

[38] R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda, "English Gigaword Fifth Edition," *Linguistic Data Consortium,* 2011.

[39] A. Herdağdelen. Rovereto Twitter N-Gram Corpus: An n-gram dataset of Twitter messages with gender labels and time of posting [Online]. Available: http://clic.cimec.unitn.it/amac/twitter_ngram/

[40] *The CMU Pronouncing Dictionary*. Available: http://www.speech.cs.cmu.edu/cgi-bin/cmudict

[41] M. Bisani and H. Ney, "Joint-Sequence Models for Grapheme-to-Phoneme Conversion," *Speech Communication,* 2008.

[42] H.-G. Hirsch. (2005). *FaNT - Filtering and Noise Adding Tool*. Available: http://dnt.kr.hs-niederrhein.de/download.html

[43] R. A. Fisher, *Statistical Methods for Research Workers*, 1st ed. Edingburgh: Oliver & Boyd, 1925.

[44] Q. Huo and C.-H. Lee, "On-line Adaptive Learning of the Correlated Continuous Density Hidden Markov Model for Speech Recognition," *IEEE Trans. Acoustic, Speech and Signal Proc.,* vol. 6, pp. 386-397, July 1998.

[45] K. Shinoda and C.-H. Lee, "Unsupervised Adaptation Using Structural Bayes Approach," in *ICASSP*, Seattle, WA, 1998, pp. 793-796.

[46] R. A. Sukkar and C.-H. Lee, "Vocabulary Independent Discriminative Utterance Verification for Non-Keyword Rejection in Subword Based Speech Recognition," *IEEE Trans. on Speech and Audio Proc.,* vol. 4, pp. 420-429, Nov. 1996.

[47] M. Rahim, C.-H. Lee, and B.-H. Juang, "Discriminative Utterance Verification for Connected Digit Recognition," *IEEE Trans. on Speech and Audio Proc.,* vol. 5, pp. 266-277, May 1997.

[48] T. Kawahara, C.-H. Lee, and B.-H. Juang, "Key-Phrase Detection and Verification for Flexible Speech Understanding," *IEEE Trans. on Speech and Audio Proc.,* vol. 6, pp. 558-568, Nov. 1998.

[49] M.-W. Koo, C.-H. Lee, and B.-H. Juang, "Speech Recognition and Utterance Verification Based on a Generalized Confidence Score," *IEEE Trans. on Speech and Audio Proc.,* vol. 9, pp. 821-832, Nov. 2001.