Chapter 1

Robust Speech Recognition for Adverse Environments

Chung-Hsien Wu and Chao-Hong Liu

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/47843

1. Introduction

As the state-of-the-art speech recognizers can achieve a very high recognition rate for clean speech, the recognition performance generally degrades drastically under noisy environments. Noise-robust speech recognition has become an important task for speech recognition in adverse environments. Recent research on noise-robust speech recognition mostly focused on two directions: (1) removing the noise from the corrupted noisy signal in signal space or feature space - such as noise filtering: spectral subtraction (Boll 1979), Wiener filtering (Macho et al. 2002) and RASTA filtering (Hermansky et al. 1994), and speech or feature enhancement using model-based approach: SPLICE (Deng et al. 2003) and stochastic vector mapping (Wu et al. 2002); (2) compensating the noise effect into acoustic models in model space so that the training environment can match the test environment - such as PMC (Wu et al. 2004) or multi-condition/multi-style training (Deng et al. 2000). The noise filtering approaches require some assumption of prior information, such as the spectral characteristic of the noise. The performance will degrade when the noisy environment vary drastically or under unknown noise environment. Furthermore, (Deng et al. 2000; Deng et al. 2003) have shown that the use of denoising or preprocessing are superior to retraining the recognizers under the matched noise conditions with no preprocessing.

Stochastic vector mapping (SVM) (Deng et al. 2003; Wu et al. 2002) and sequential noise estimation (Benveniste et al. 1990; Deng et al. 2003; Gales et al. 1996) for noise normalization have been proposed and achieved significant improvement in noisy speech recognition. However, there still exist some drawbacks and limitations. First, the performance of sequential noise estimation will decrease when the noisy environment vary drastically. Second, the environment mismatch between training data and test data still exists and results in performance degradation. Third, the maximum-likelihood-based stochastic vector

mapping (SPLICE) requires annotation of environment type and stereo training data. Nevertheless, the stereo data are not available for most noisy environments.

In order to overcome the insufficiency of tracking ability in the sequential expectation-maximization (EM) algorithm, in this chapter, the prior models were introduced to provide more information in sequential noise estimation. Furthermore, an environment model adaptation is constructed to reduce the mismatch between the training data and the test data. Finally, minimum classification error (MCE)-based approach (Wu et al. 2002) was employed without the stereo training data and an unsupervised frame-based auto-clustering was adopted to automatically detect the environment type of the training data (Hsieh et al. 2008).

For recognition of disfluent speech, a number of cues can be observed when edit difluency occurs in the spontaneous speech. These cues can be detected from linguistic features, acoustic features (Shriberg et al. 2000) and integrated knowledge sources (Bear et al. 1992). (Shriberg et al. 2005) outlined phonetic consequences of disfluency to improve models for disfluency processing in speech applications. Four types of disfluency based on intonation, segment duration and pause duration were presented in (Savova et al. 2003). Soltau et al. used a discriminatively trained full covariance Gaussian system for rich transcription (Soltau et al. 2005). (Furui et al. 2005) presented the approaches to corpus collection, analysis and annotation for conversational speech processing.

(Charniak et al. 2001) proposed an architecture for parsing the transcribed speech using an edit word detector to remove edit words or fillers from the sentence string, and then a standard statistical parser was used to parse the remaining words. The statistical parser and the parameters estimated by boosting were employed to detect and correct the disfluency. (Heeman et al. 1999) presented a statistical language model that is able to identify POS tags, discourse markers, speech repairs and intonational phrases. A noisy channel model was used to model the disfluency in (Johnson et al. 2004). (Snover et al. 2004) combined the lexical information and rules generated from 33 rule templates for disfluency detection. (Hain et al. 2005) presented the techniques in front-end processing, acoustic modeling, language and pronunciation modeling for transcribing the conversational telephone speech automatically. (Liu et al. 2005) compared the HMM, maximum entropy, and conditional random fields for disfluency detection in detail.

In this chapter an approach to the detection and correction of the edit disfluency based on the word order information is presented (Yeh et al. 2006). The first process attempts to detect the interruption points (IPs) based on hypothesis testing. Acoustic features including duration, pitch and energy features were adopted in hypothesis testing. In order to circumvent the problems resulted from disfluency especially in edit disfluency, a reliable and robust language model for correcting speech recognition errors was employed. For handling language-related phenomena in edit disfluency, a cleanup language model characterizing the structure of the cleanup sentences and an alignment model for aligning words between deletable region and correction part are proposed for edit disfluency detection and correction.

Furthermore, multilinguality frequently occurs in speech content, and the ability to process speech in multiple languages by the speech recognition systems has become increasingly desirable due to the trend of globalization. In general, there are different approaches to achieving multilingual speech recognition. One approach employing external language identification (LID) systems (Wu et al. 2006) to firstly identify the language of the input utterance and the corresponding monolingual system is then selected to perform the speech recognition (Waibel et al. 2000). The accuracy of the external LID system is the main factor to the overall system performance.

Another approach to multilingual speech recognition is to run all the monolingual recognizers in parallel and select the output generated by the recognizer that obtains the maximum likelihood score. The performance of the multilingual speech recognition depends on the post-end selection of the maximum likelihood sequence. The popular approaches to multilingual speech recognition are the utilization of a multilingual phone set. The multilingual phones are usually created by merging the phones across the target languages that are acoustically similar in an attempt to obtain a minimal phone set that covers all the sounds existing in all the target languages (Kohler 2001).

In this chapter, an approach to phonetic unit generation for mixed-language or multilingual speech recognition is presented (Huang et al. 2007). The International Phonetic Alphabet (IPA) representation is employed for phonetic unit modeling. Context-dependent triphones for Mandarin and English speech are constructed based on the IPA representation. Acoustic and contextual analysis is investigated to characterize the properties among the multilingual context-dependent phonetic units. Acoustic likelihood is adopted for the pair-wise similarity estimation of the context-dependent phone models to construct a confusing matrix. The hyperspace analog to language (HAL) model is used for contextual modeling and then used for contextual similarity estimation between phone models.

The organization of this paper is as follows. Section 2 presents two approaches to cepstral feature enhancement for noisy speech recognition using noise-normalized stochastic vector mapping. Section 3 describes an approach to edit disfluency detection and correction for rich transcription. In Section 4, fusion of acoustic and contextual analysis is described to generate phonetic units for mixed-language or multilingual speech recognition. Finally the conclusions are provided in the last section.

2. Speech recognition in noisy environment

In this section, an approach to feature enhancement for noisy speech recognition is presented. Three prior models are introduced to characterize clean speech, noise and noisy speech, respectively. The framework of the system is shown in Figure 1. Sequential noise estimation is employed for prior model construction based on noise-normalized stochastic vector mapping (NN-SVM). Therefore, feature enhancement can work without stereo training data and manual tagging of background noise type based on auto-clustering on the estimated noise data. Environment model adaptation is also adopted to reduce the mismatch between the training data and the test data.

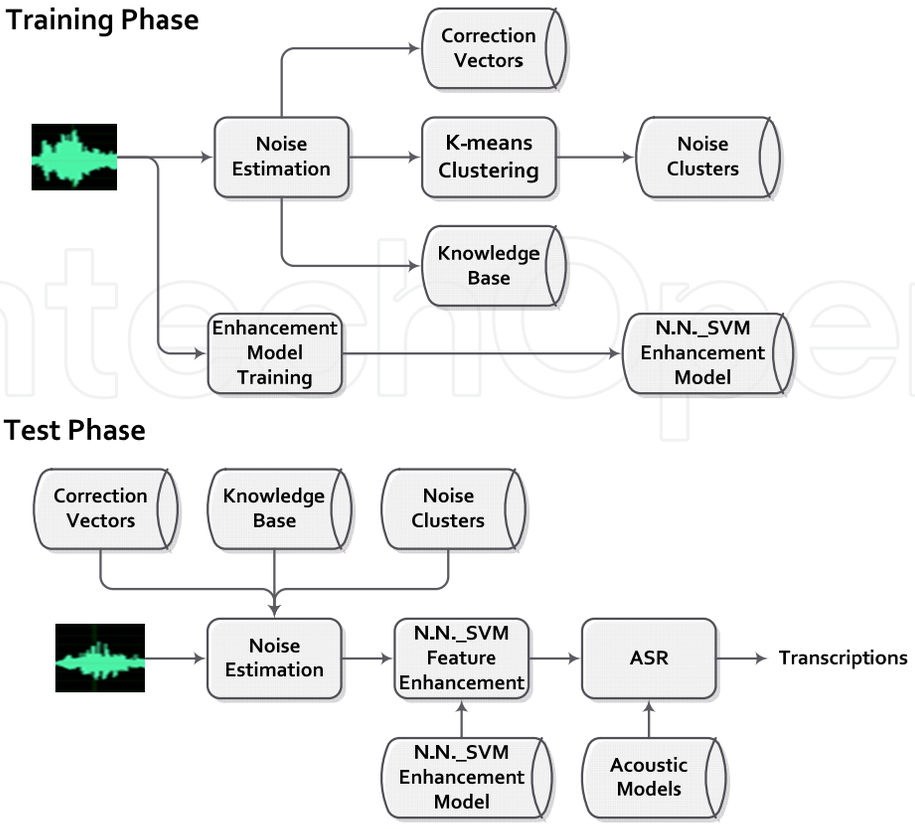

Figure 1. Diagram of training and test phases for noise-robust speech recognition

2.1. NN-SVM for cepstral feature enhancement

2.1.1. Stochastic Vector Mapping (SVM)

The SVM-based feature enhancement approach estimates the clean speech feature $\hat{x}$ from the noisy speech feature y through an environment-dependent mapping function $F\left(y;\Theta^{(e)}\right)$, where $\Theta^{(e)}$ denotes the mapping function parameters and e denotes the corresponding environment of noisy speech feature y.

Assuming that the training data of the noisy speech Y can be partitioned into N_e different noisy environments, the feature vectors of Y under an environment e can be modeled by a Gaussian mixture model (GMM) with N_k mixtures:

$$p\left(y \mid e;\Omega_{\mathrm{e}}\right)=\sum_{k=1}^{N_k} p\left(k \mid e\right)p\left(y \mid k,e\right)=\sum_{k=1}^{N_k}\omega_k^e \cdot N\left(y;\xi_k^e,R_k^e\right) \tag{1}$$

where Ω_{e} represents the environment model. The clean speech feature $\hat{x}$ can be estimated using a stochastic vector mapping function which is defined as follows:

$$\hat{x}\triangleq F\left(y;\Theta^{(e)}\right)=y+\sum_{e=1}^{N_E}\sum_{k=1}^{N_k} p\left(k \mid y,e\right)r_k^e \tag{2}$$

where the posterior probability $p(k \mid y,e)$ can be estimated using the Bayes theory based on the environment model Ω_e as follows:

$$p(k \mid y,e) = \frac{p(k \mid e)p(y \mid k,e)}{\sum_{j=1}^{N_k} p(j \mid e)p(y \mid j,e)} \tag{3}$$

and $\Theta^{(e)} = \left\{ r_k^{(e)} \right\}_{k=1}^{N_k}$ denotes the mapping function parameters. Generally, $\Theta^{(e)}$ are estimated from a set of training data using maximum likelihood criterion.

For the estimation of the mapping function parameter $\Theta^{(e)}$, if the stereo data, which contain a clean speech signal and the corrupted noisy speech signal with the identical clean speech signal, are available, the SPLICE-based approach can be directly adopted. However, the stereo data are not easily available in real-life applications. In this chapter an MCE-based approach is proposed to overcome the limitation. Furthermore, the environment type of the noisy speech data is needed for training the environment model $\Theta^{(e)}$. The noisy speech data are manually classified into N_E noisy environments types. This strategy assigns each noisy speech file to only one environment type and is very time consuming. Actually, each noisy speech file contains several segments with different types of noisy environment. Since the noisy speech annotation affects the purity of the training data for the environment model $\Theta^{(e)}$, this section introduces a frame-based unsupervised noise clustering approach to construct a more precise categorization of the noisy speech.

2.1.2. Noise-Normalized Stochastic Vector Mapping (NN-SVM)

In (Boll 1979), the concept of noise normalization is proposed to reduce the effect of background noise in noisy speech for feature enhancement. If the noise feature vector $\tilde{n}$ of each frame can be estimated first, the NN-SVM is conducted from Eq.**Error! Reference source not found.**(2) by replacing y and $\hat{x}$ with $y - \tilde{n}$ and $\hat{x} - \tilde{n}$ as

$$\hat{x} - \tilde{n} \triangleq F\left(y - \tilde{n}; \Theta^{(e)}\right) = y - \tilde{n} + \sum_{e=1}^{N_E} \sum_{k=1}^{N_k} p\left(k \mid y - \tilde{n}, e\right) r_k^e \tag{4}$$

The process for noise normalization makes the environment model Ω_e more noise-tolerable. Obviously, the estimation algorithm of noise feature vector $\tilde{n}$ plays an important role in noise-normalized stochastic vector mapping.

2.2. Prior model for sequential noise estimation

This section employs a frame-based sequential noise estimation algorithm (Benveniste et al. 1990; Deng et al. 2003; Gales et al. 1996) by incorporating the prior models. In the procedure,

only noisy speech feature vector of the current frame is observed. Since the noise and clean speech feature vectors are missing simultaneously, the relation among clean speech, noise and noisy speech is required first. Then the sequential EM algorithm is introduced for online noise estimation based on the relation. In the meantime, the prior models are involved to provide more information for noise estimation.

2.2.1. The acoustic environment model

The nonlinear acoustic environment model is introduced first for noise estimation in (Deng et al. 2003). Given the cepstral features of a clean speech x, an additive noise n and a channel distortion h, the approximated nonlinear relation among x, n, h and the corrupted noisy speech y in cepstral domain is estimated as:

$$y \approx h + x + g\left(n - h - x\right),\ g(z) = \mathrm{C}\ln\left(I + \exp\left[\mathrm{C}^T\left(z\right)\right]\right) \tag{5}$$

where **C** denotes the discrete cosine transform matrix. In order to linearize the nonlinear model, the first order Taylor series expansion was used around two updated operating points n_0 and μ_0^x denoting the initial noise feature and the mean vector of the prior clean speech model, respectively. By ignoring the channel distortion effect, for which $h = 0$, Eq.**Error! Reference source not found.**(5) is then derived as:

$$y \approx \mu_0^x + g\left(n_0 - \mu_0^x\right) + G\left(n_0 - \mu_0^x\right)\left(x - \mu_0^x\right) + \left[I - G\left(n_0 - \mu_0^x\right)\right]\left(n - n_0\right) \tag{6}$$

where $G(z) = -C diag\left(\dfrac{I}{I + \exp\left[C^T z\right]}\right) C^T$.

2.2.2. The prior models

The three prior models Φ_n, Φ_x and Φ_y, which denotes noise, clean speech and noisy speech models respectively, can provide more information for sequential noise estimation. First, the noise and clean speech prior models are characterized by GMMs as:

$$p\left(n;\Phi_n\right) = \sum_{d=1}^{N_d} w_d^n \cdot N\left(n;\mu_d^n,\Sigma_d^n\right),\ p\left(x;\Phi_x\right) = \sum_{m=1}^{N_m} w_m^x \cdot N\left(x;\mu_m^x,\Sigma_m^x\right) \tag{7}$$

where the pre-training data for noisy and clean speech are required to train the model parameters of the two GMMs, Φ_n and Φ_x.

While the prior noisy speech model is needed in sequential noise estimation, the noisy speech model parameters are derived according to the prior clean speech and noise models using the approximated linear model around two operating points μ_0^n and μ_0^x as follows:

$$p\left(\mathrm{y};\Phi_y\right)=\sum_{m=1}^{N_m}\sum_{d=1}^{N_d}w_{m,d}^{y}\cdot N\left(\mathrm{y};\mu_{m,d}^{y},\Sigma_{m,d}^{y}\right) \tag{8}$$

$$\begin{aligned}\mu_{m,d}^{y}&=\mu_0^x+g\left(\mu_0^n-\mu_0^x\right)+G\left(\mu_0^n-\mu_0^x\right)\left(\mu_m^x-\mu_0^x\right)+\left[I-G\left(\mu_0^n-\mu_0^x\right)\right]\left(\mu_d^n-\mu_0^n\right)\\ \Sigma_{m,d}^{y}&=\left[I+G\left(\mu_0^n-\mu_0^x\right)\right]\Sigma_m^x\left[I+G^T\left(\mu_0^n-\mu_0^x\right)\right]^T\\ \mu_0^n&=E[\mu_d^n],\ \mu_0^x=E[\mu_m^x],\ w_{m,d}^{y}=w_m\bullet w_d\end{aligned} \tag{9}$$

The noisy speech prior model will be employed to search the most similar clean speech mixture component and noise mixture component in sequential noise estimation.

2.2.3. Sequential noise estimation

Sequential EM algorithm is employed for sequential noise estimation. In this section, the prior clean speech, noise and noisy speech model are considered to construct a robust noise estimation procedure. Based on the sequential EM algorithm, the estimated noise is obtained from $n_{t+1}=\arg\max_n Q_{t+1}(n)$. In the E-step of the sequential EM algorithm, an objective function is defined as:

$$Q_{t+1}(\mathrm{n})\triangleq E\left[\ln p(y_1^{t+1},M_1^{t+1},D_1^{t+1}\mid\mathrm{n})\mid y_1^{t+1},\mathrm{n}_1^t\right] \tag{10}$$

where M_1^{t+1} and D_1^{t+1} denote the mixture index sequence of the clean speech GMM and the noise GMM in which the noisy speech y occurs from frame 1 to frame t+1. The objective function is simplified for the M-step as:

$$\begin{aligned}Q_{t+1}(\mathrm{n})&\triangleq E\left[\ln p(y_1^{t+1},M_1^{t+1},D_1^{t+1}\mid\mathrm{n})\mid y_1^{t+1},\mathrm{n}_1^t\right]\\ &=E\left[\ln p(y_1^{t+1}\mid M_1^{t+1},D_1^{t+1},\mathrm{n})+\ln p(M_1^{t+1}\mid D_1^{t+1},\mathrm{n})+\ln p(D_1^{t+1}\mid\mathrm{n})\mid y_1^{t+1},\mathrm{n}_1^t\right]\\ &\approx E\left[\ln p(y_1^{t+1}\mid M_1^{t+1},D_1^{t+1},\mathrm{n})\mid y_1^{t+1},\mathrm{n}_1^t\right]+Const\\ &=\sum_{\tau=1}^{t+1}E\left[\left(\sum_{m=1}^{N_m}\sum_{d=1}^{N_d}\ln p(y_\tau\mid m,d,\mathrm{n})\cdot\delta_{m_\tau,m}\cdot\delta_{d_\tau,d}\right)\mid y_1^{t+1},\mathrm{n}_1^t\right]+Const\\ &=\sum_{\tau=1}^{t+1}\sum_{m=1}^{N_m}\sum_{d=1}^{N_d}E\left[\delta_{m_\tau,m}\delta_{d_\tau,d}\mid y_1^{t+1},\mathrm{n}_1^t\right]\ln p(y_\tau\mid m,d,\mathrm{n})+Const\\ &=\sum_{\tau=1}^{t+1}\sum_{m=1}^{N_m}\sum_{d=1}^{N_d}\gamma_\tau(m,d)\cdot\ln p(y_\tau\mid m,d,\mathrm{n})+Const\end{aligned} \tag{11}$$

where $\delta_{m_\tau,m}$ denotes the Kronecker delta function and $\gamma_\tau(m,d)$ denotes the posterior probability. $\gamma_\tau(m,d)$ can be estimated according to the Bayes rule as:

$$\begin{aligned}\gamma_\tau(m,d) &\equiv E\left[\delta_{m_\tau,m}\delta_{d_\tau,d} \mid y_1^{t+1}, \mathrm{n}_1^t\right] \\ &= \sum_{M_1^{t+1}}\sum_{D_1^{t+1}} p\left(M_1^{t+1}, D_1^{t+1} \mid y_1^{t+1}, \mathrm{n}_1^t\right)\delta_{m_\tau,m}\delta_{d_\tau,d} \\ &= p\left(m,d \mid y_\tau, \mathrm{n}_{\tau-1}\right) \\ &= \frac{p\left(y_\tau \mid m,d,\mathrm{n}_{\tau-1}\right)p\left(m,d \mid \mathrm{n}_{\tau-1}\right)}{\sum_{m=1}^{N_m}\sum_{d=1}^{N_d} p\left(y_\tau \mid m,d,\mathrm{n}_{\tau-1}\right)p\left(m,d \mid \mathrm{n}_{\tau-1}\right)} \\ &= \frac{p\left(y_\tau \mid m,d,\mathrm{n}_{\tau-1}\right)\cdot w_m \cdot w_d}{\sum_{m=1}^{N_m}\sum_{c=1}^{N_d} p\left(y_\tau \mid m,d,\mathrm{n}_{\tau-1}\right)\cdot w_m \cdot w_d}\end{aligned} \tag{12}$$

where the likelihood $p\left(y_\tau \mid m,d,\mathrm{n}_{\tau-1}\right)$ can be approximated using the approximated linear model as:

$$\begin{aligned}&p\left(y_\tau \mid m,d,\mathrm{n}_{\tau-1}\right) \sim N\left[y_\tau; \mu_{m,c}^y\left(\mathrm{n}_{\tau-1}\right), \Sigma_{m,d}^y\right] \\ &\mu_{m,d}^y\left(\mu_{\tau-1}^n\right) = \mu_0^x + g\left(\mathrm{n}_0 - \mu_0^x\right) + G\left(\mathrm{n}_0 - \mu_0^x\right)\left(\mu_m^x - \mu_0^x\right) + \left[I - G\left(\mathrm{n}_0 - \mu_0^x\right)\right]\left(\mathrm{n}_{\tau-1} - \mathrm{n}_0\right) \\ &\Sigma_{m,d}^y = \left[I + G\left(\mathrm{n}_0 - \mu_0^x\right)\right]\Sigma_m^x\left[I + G^T\left(\mathrm{n}_0 - \mu_0^x\right)\right]^T\end{aligned} \tag{13}$$

Also, a forgetting factor is employed to control the effect of the features of the preceding frames.

$$\begin{aligned}Q_{t+1}(\mathrm{n}) &= \sum_{\tau=1}^{t+1}\varepsilon^{t+1-\tau}\sum_{m=1}^{N_m}\sum_{d=1}^{N_d}\gamma_\tau(m,d)\cdot\ln p(y_\tau \mid m,d,\mathrm{n}) + Const \\ \widetilde{Q}_{t+1}(\mathrm{n}) &= -\sum_{\tau=1}^{t+1}\varepsilon^{t+1-\tau}\sum_{m=1}^{N_m}\sum_{d=1}^{N_d}\gamma_\tau(m,d)\cdot\left[y_\tau - \mu_{m,d}^y\left(\mathrm{n}_\tau\right)\right]^T\left(\Sigma_{m,d}^y\right)^{-1}\left[y_\tau - \mu_{m,d}^y\left(\mathrm{n}_\tau\right)\right] \\ &= \varepsilon\widetilde{Q}_t(\mathrm{n}) - R_{t+1}\left(\mathrm{n}\right) \\ R_{t+1}\left(\mathrm{n}\right) &= \sum_{m=1}^{N_m}\sum_{d=1}^{N_d}\gamma_{t+1}(m,d)\cdot\left[y_\tau - \mu_{m,d}^y\left(\mathrm{n}_\tau\right)\right]^T\left(\Sigma_{m,d}^y\right)^{-1}\left[y_\tau - \mu_{m,d}^y\left(\mathrm{n}_\tau\right)\right]\end{aligned} \tag{14}$$

In the M-step, the iterative stochastic approximation is introduced to derive the solution. Finally, sequential noise estimation is performed as follows:

$$
\begin{aligned}
&\mathrm{n}_{t+1} = \mathrm{n}_t + \left(K_{t+1}\right)^{-1} s_{t+1} \quad K_{t+1} = -\frac{\partial^2 Q_{t+1}}{\partial^2 \mathrm{n}}|_{\mathrm{n}=\mathrm{n}_t} \quad s_{t+1} = -\frac{\partial R_{t+1}}{\partial \mathrm{n}}|_{\mathrm{n}=\mathrm{n}_t} \\
&K_{t+1} = -\frac{\partial^2 Q_{t+1}}{\partial^2 \mathrm{n}}|_{\mathrm{n}=\mathrm{n}_t} \\
&\quad = \sum_{\tau=1}^{t+1} \varepsilon^{t+1-\tau} \sum_{\mathrm{m}=1}^{N_m} \sum_{d=1}^{N_d} \gamma_\tau(m,d) \left[I - G\left(\mathrm{n}_0 - \mu_0^x\right) \right]^T \left(\Sigma_{m,d}^y\right)^{-1} \left[I - G\left(\mathrm{n}_0 - \mu_0^x\right) \right] \\
&s_{t+1} = -\frac{\partial R_{t+1}}{\partial \mathrm{n}}|_{\mathrm{n}=\mathrm{n}_t} \\
&\quad = \sum_{\mathrm{m}=1}^{N_m} \sum_{d=1}^{N_d} \gamma_{t+1}(m,d) \left[I - G\left(\mathrm{n}_0 - \mu_0^x\right) \right]^T \left(\Sigma_{m,d}^y\right)^{-1} \left[y_{t+1} - \mu_{m,d}^y\left(\mathrm{n}_t\right) \right]
\end{aligned}
\tag{15}
$$

The prior models are used to search the most similar noise or clean speech mixture component. Given the two mixture components, the estimation of the posterior probability $\gamma_\tau(m,d)$ will be more accurate.

2.3. Environment model adaptation

Because the prior models are usually not complete enough to represent the universal data, the environment mismatch between the training data and the test data will result in the degradation on feature enhancement performance. In this section, an environment model adaptation strategy is proposed before the test phase to deal with the problem. The environment model adaptation procedure contains two parts: The first one is model parameter adaptation on noise prior model Φ_n and noisy speech prior model Φ_y in the training phase and adaptation phase. The second is on noise-normalized SVM function $\Theta^{(e)}$ and environment model Ω_e in the adaptation phase.

2.3.1. Model adaptation on noise and noisy speech prior models

For noise and noisy speech prior model adaptation, MAP adaptation is applied to the noise prior model Φ_n first. The adaptation equations for the noise prior model parameters given T frames of the adaptation noise data z, which is estimated using the un-adapted prior models, are defined as:

$$
\begin{aligned}
&\widetilde{w}_d = \left(\nu_d - 1\right) + \sum_{t=1}^{T} s_{d,t} \Big/ \sum_{d=1}^{N_d} \left(\nu_d - 1\right) + \sum_{d=1}^{N_d} \sum_{t=1}^{T} s_{d,t} \\
&\widetilde{\mu}_d^n = \tau_d \rho_d + \sum_{t=1}^{T} s_{d,t} \cdot z_t \Big/ \tau_d + \sum_{t=1}^{T} s_{d,t} \\
&\widetilde{\Sigma}_d^{n\,-1} = \upsilon_d + \sum_{t=1}^{T} s_{d,t} \left(z_t - \widetilde{\mu}_d^n \right) \left(z_t - \widetilde{\mu}_d^n \right)^T + \tau_d \left(\rho_d - \widetilde{\mu}_d^n \right) \left(\rho_d - \widetilde{\mu}_d^n \right)^T \Big/ \left(\alpha_d - p\right) + \sum_{t=1}^{T} s_{d,t}
\end{aligned}
\tag{16}
$$

where the conjugate prior density of the mixture weight is the Dirichlet distribution with hyper-parameter ν_d and the joint conjugate prior density of mean and variance parameters

is the Normal-Wishart distribution with hyper-parameters τ_d, ρ_d, α_d, and υ_d. The two distributions are defined as follows:

$$g\left(w_1,...,w_{N_d} \mid \nu_1,...,\nu_{N_d}\right) \propto \prod_{d=1}^{N_d} w_d^{\nu_d - 1}$$
$$g\left(\mu_d^n, \Sigma_d^n \mid \tau_d, \rho_d, \alpha_d, \upsilon_d\right) \propto \left|\Sigma_d^n\right|^{(\alpha_d - p)/2} \exp\left[-\frac{\tau_d}{2}\left(\mu_d^n - \rho_d\right)^T \tau_d \left(\mu_d^n - \rho_d\right)\right] \exp\left[-\frac{1}{2}\mathrm{tr}\left(\upsilon_d \Sigma_d^n\right)\right] \quad (17)$$

where $\nu_d > 0$, $\alpha_k > p-1$ and $\tau_k > 0$. After adaptation of noise prior model, the noisy speech prior model Φ_y is then adapted using the clean speech prior model Φ_x and the newly adapted noise prior model Φ_n based on Eq.**Error! Reference source not found.**(8).

2.3.2. Model adaptation of noise-normalized SVM (NN-SVM)

For NN-SVM adaptation, model parameters Ω_e and mapping function parameters in $F\left(y;\Theta^{(e)}\right)$ need to be adapted in the adaptation phase. First, adaptation of model parameter Ω_e is similar to that of noise prior model. Second, the adaptation of $\Theta^{(e)} = \left\{r_k^{(e)}\right\}_{k=1}^{N_k}$ is an iterative procedure. While $\Theta^{(e)} = \left\{r_k^{(e)}\right\}_{k=1}^{N_k}$ is not a random variable and does not follow any conjugate prior density, a maximum likelihood (ML)-based adaptation which is similar to the correction vector estimation of SPLICE is employed as:

$$\widetilde{r_k^{(e)}} = \sum_t p\left(k \mid y_t - \tilde{n}, e\right)\left(\widetilde{x_t} - y_t\right) \Big/ \sum_t p\left(k \mid y_t - \tilde{n}, e\right) \quad (18)$$

where the temporal estimated clean speech $\widetilde{x_t}$ are estimated using the un-adapted noise normalized stochastic mapping function in Eq.(4).

2.4. Experimental results

Table 1 shows the experimental results of the proposed approach on AURORA2 database. The AURORA2 database contains both clean and noisy utterances of the TIDIGITS corpus and is available from ELDA (Evaluations and Language resources Distribution Agency). Two results of previous research were illustrated for comparison and three experiments were conducted for different experimental conditions: no denoising, SPLICE with recursive EM using stereo data (Deng et al. 2003), the proposed approach using manual annotation without adaptation, and the proposed approach using auto-clustered training data without and with adaptation. The overall results show that the proposed approach slightly outperformed the SPLICE-based approach with recursive EM algorithm under the lack of stereo training data and manual annotation. Furthermore, based on the results in Set B with 0.11% improvement (different background noise types to the training data) and Set C with

0.04% improvement (different background noise types and channel characteristic to the training data), the environment model adaptation can slightly reduce the mismatch between the training data and test data.

Methods	Training-Mode	Set A	Set B	Set C	Overall
No Denoising	Multi-condition	87.82	86.27	83.78	86.39
	Clean only	61.34	55.75	66.14	60.06
MCE	Multi-condition	92.92	89.15	90.09	90.85
	Clean only	87.82	85.34	83.77	86.02
SPLICE with Recursive-EM	Multi-condition	91.49	89.16	89.62	90.18
	Clean only	87.82	87.09	85.08	86.98
Proposed approach (manual tag, no adaptation)	Multi-condition	91.42	89.18	89.85	90.21
	Clean only	87.84	86.77	85.23	86.89
Proposed approach (auto-clustering, no adaptation)	Multi-condition	91.06	90.79	90.77	90.89
	Clean only	87.56	87.33	86.32	87.22
Proposed approach (auto-clustering, with adaptation)	Multi-condition	91.07	90.90	90.81	90.95
	Clean only	87.55	87.44	86.38	87.27

Table 1. Experimental results (%) on AURORA2

2.5. Conclusions

In this section two approaches to cepstral feature enhancement for noisy speech recognition using noise-normalized stochastic vector mapping are presented. The prior model was introduced for precise noise estimation. Then the environment model adaptation is constructed to reduce the environment mismatch between the training data and the test data. Experimental results demonstrate that the proposed approach can slightly outperform the SPLICE-based approach without stereo data on AURORA2 database.

3. Speech recognition in disfluent environment

In this section, a novel approach to detecting and correcting the edit disfluency in spontaneous speech is presented. Hypothesis testing using acoustic features is fist adopted to detect potential interruption points (IPs) in the input speech. The word order of the utterance is then cleaned up based on the potential IPs using a class-based cleanup language model. The deletable region and the correction are aligned using an alignment model. Finally, a log linear weighting mechanism is applied to optimize the performance.

3.1. Edit disfluency ANalsis

In conversational utterances, several problems such as interruption, correction, filled pause, and ungrammatical sentence are detrimental for speech recognition. The definitions of disfluencies have been discussed in SimpleMDE. Edit disfluencies are portions of speech in which a speaker's utterance is not complete and fluent; instead the speaker corrects or alters the utterance, or abandons it entirely and starts over. In general, edit disfluencies can be divided into four categories: repetitions, revisions, restarts and complex disfluencies. Since complex disfluencies consist of multiple or nested edits, it seems reasonable to consider the complex disfluencies as a combination of the other simple disfluencies: repetitions, revisions, and restarts. Edit disfluencies have a complex internal structure, consisting of the deletable region (delreg), interruption point (IP) and correction. Editing terms such as fillers, particles and markers are optional and follow the IP in edit disfluency.

In spontaneous speech, acoustic features such as short pause (silence and filler), energy and pitch reset generally appear along with the occurrence of edit dislfuency. Based on these features, we can detect the possible IPs. Furthermore, since IPs generally appear at the boundary of two successive words, we can exclude the unlikely IPs whose positions are within a word. Besides, since the structural patterns between the deletable word sequence and correction word sequence are very similar, the deletable word sequence in edit disfluency is replaceable by the correction word sequence.

3.2. Framework of edit disfluency transcription system

The overall transcription task for conversational speech with edit disfluency in the proposed method is composed of two main mechanisms; IP detection module and edit disfluency correction module. The framework is shown in Figure 2. IP detection module predicts the potential IPs first. Edit disfluency correction module generates the rich transcription that contains information of interruption, text transcription from the speaker’s utterances and the cleaned-up text transcription without disfluencies. Figure 3 shows the correction process for edit disfluency.

The speech signal is fed to both acoustic feature extraction module and speech recognition engine in IP detection module. Information about durations of syllables and silence from speech recognition is provided for acoustic feature extraction. Combined with side information from speech recognition, duration-, pitch-, and energy-related features are extracted and used to model the IPs using a Gaussian mixture model (GMM). Besides, in order to perform hypothesis testing on IP detection, an anti-IP GMM is also constructed based on the extracted features from the non-IP regions. The hypothesis testing verifies if the posterior probability of the acoustic features of a syllable boundary is above a threshold and therefore determines if the syllable boundary is an IP. Since IP is an event that happens in interword location, we can remove the detected IPs that do not appear in the word boundary.

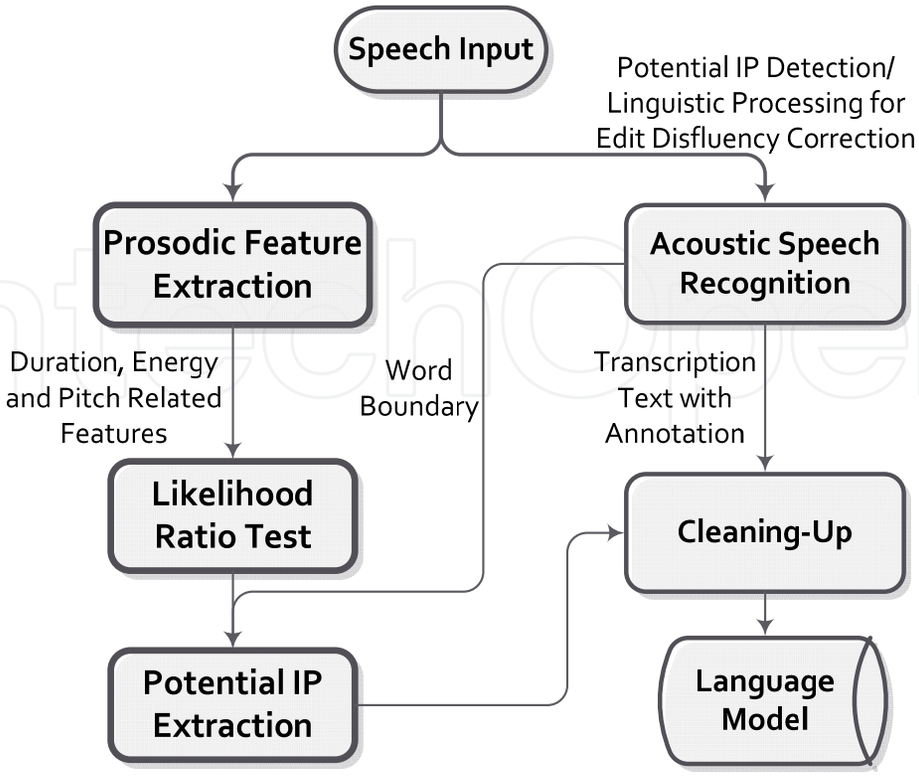

Figure 2. The framework of transcription system for spontaneous speech with edit disfluencies

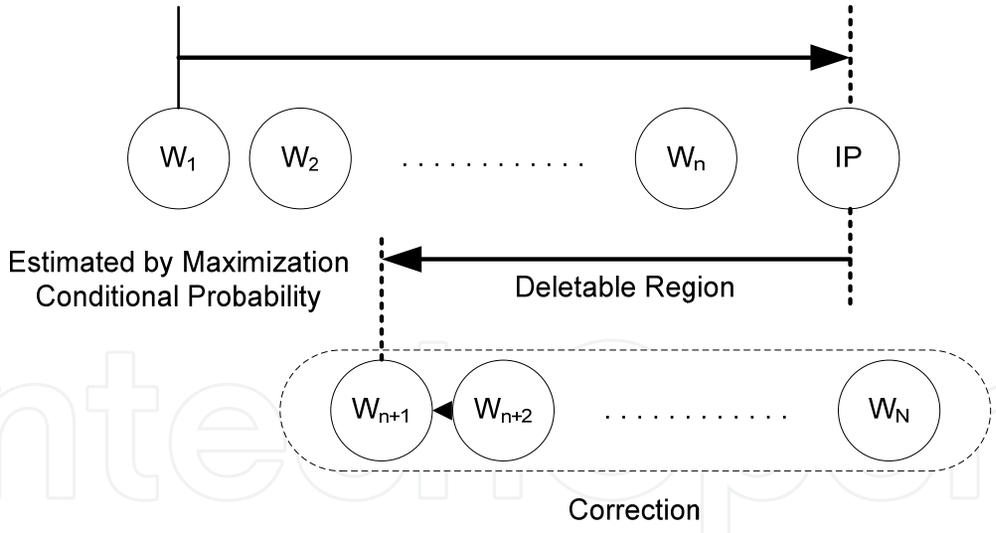

Figure 3. The correction process for the edit disfluency

There are two processing stages in the edit disfluency correction module: cleanup and alignment. As shown in Figure 4, cleanup process divides the word string into three parts: deletable region (delreg), editing term, and correction according to the locations of potential IPs detected by the IP detection module. Cleanup process is performed by shifting the correction part and replaces the deletable region to form a new cleanup transcription. The edit disfluency correction module is composed of an *n*-gram language model and the alignment model. The *n*-gram model regards the cleanup transcriptions as fluent utterances

and models their word order information. The alignment model finds the optimal correspondence between deletable region and correction in edit disfluency.

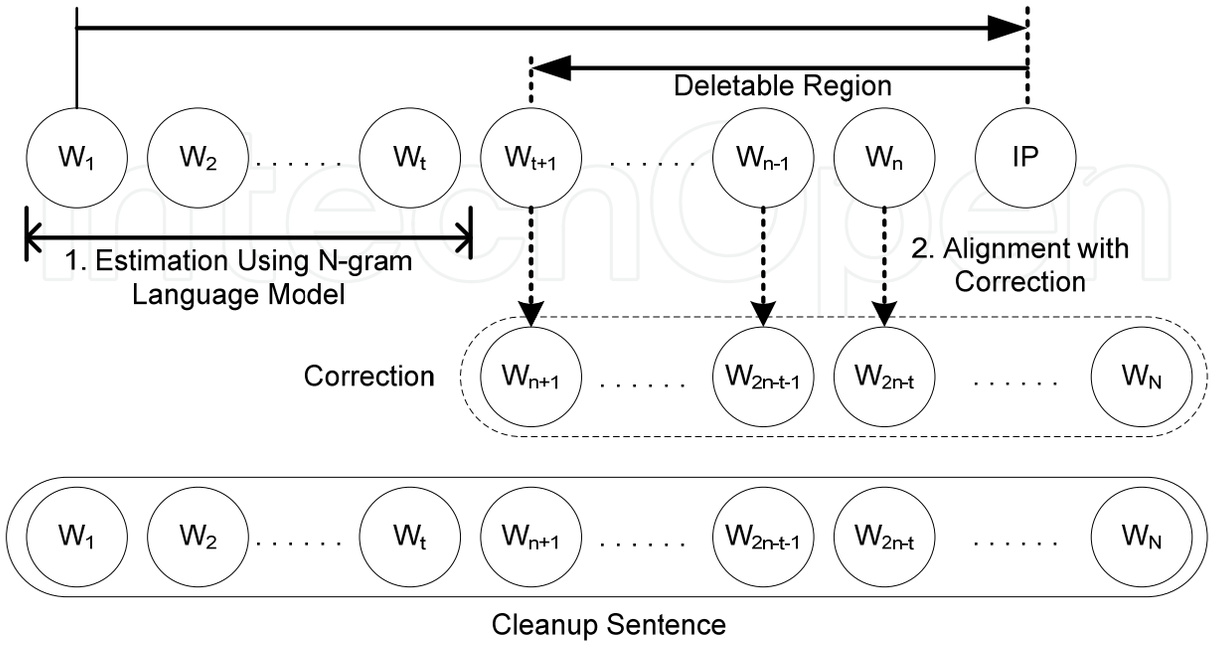

Figure 4. The cleanup language model for the edit disfluency

3.3. Potential interruption point detection

For IP detection, instead of detecting exact IP, potential IPs are selected for further processing. Since the IP is the point at which the speaker breaks off the deletable region, some acoustic events will go along with it. For syllabic languages like Chinese, every character is pronounced as a monosyllable, while a word is composed of one to several syllables. The speech input of the syllabic languages with n syllables can be described as a sequence,

$$Seq_{syllable\_silence} \equiv syllable_1,\ silence_1,\ syllable_2,\ silence_2,...,silence_{n-1},\ syllable_n,$$

and then this sequence can be separated into a syllable sequence

$$Seq_{syllable} \equiv syllable_1,\ syllable_2,\ ...,\ syllable_n,$$

and a silence sequence

$$Seq_{silence} \equiv silence_1,\ silence_2,...,silence_{n-1}.$$

We model the interruption detection problem as choosing between H_0, which is termed the IP not embedded in the silence hypothesis, and H_1 which is the IP embedded in the silence hypothesis. The likelihood ratio test is employed to detect the potential IPs. The function $L\left(Seq_{syllable\_silence}\right)$ is termed the likelihood ratio since it indicates for each value of $Sequence_{syllable\_silence}$ the likelihood of H_1 versus the likelihood of H_0.

$$L\left(Seq_{syllable\_silence}\right)=\frac{P\left(Seq_{syllable\_silence};H_1\right)}{P\left(Seq_{syllable\_silence};H_0\right)} \tag{19}$$

By introducing the threshold γ to adjust the precision and recall rates, $H_1 : L\left(\text{Seq}_{\text{syllable\_silence}}\right) \geq \gamma$ means the IP is embedded in $silence_k$. Conceptually, $silence_k$ is a potential IP. Under the assumption of independence, the probability of IP appearing in $silence_k$ can be regarded as the product of probabilities obtained from $silence_k$ and the syllables around it. The probability density functions (PDFs) under each hypothesis are denoted and estimated as

$$\begin{aligned} P\left(Seq_{syllable\_silence};H_1\right) &= P\left(Seq_{syllable\_silence} \mid E_{ip}\right) \\ &= P\left(Seq_{silence} \mid E_{ip}\right)\times P\left(Seq_{syllable} \mid E_{ip}\right) \end{aligned} \tag{20}$$

and

$$\begin{aligned} P\left(Seq_{syllable\_silence};H_0\right) &= P\left(Seq_{syllable\_silence} \mid \neg E_{ip}\right) \\ &= P\left(Seq_{silence} \mid \neg E_{ip}\right)\times P\left(Seq_{syllable} \mid \neg E_{ip}\right) \end{aligned} \tag{21}$$

Where E_{ip} denotes that IP is embedded in $silence_k$ and $\neg E_{ip}$ means that IP does not appear in $silence_k$, that is,

E_{ip} : Interuption point $\in silence_k$

$\neg E_{ip}$: Interuption point $\notin silence_k$

3.3.1. IP detection using posterior probability of silence duration

Since IPs always appear at the inter-syllable position, the n-1 silence positions between n syllables will be considered as the IP candidates. By this, we can take the IP detection as the problem to verify whether each of the n-1 silence positions is an IP or not. In conversation, speakers may hesitate to find the correct words when disfluency appears. Hesitation is usually realized as a pause. Since the length of silence is very sensitive to disfluency, we use normal distributions to model the posterior probabilities of that IP appears and does not appear in $silence_k$, respectively.

$$P\left(Seq_{silence} \mid E_{ip}\right)=\frac{2}{\sqrt{2\pi}\sigma_{ip}}\exp\left(-\frac{\left(Seq_{silence}-\mu_{ip}\right)^2}{2{\sigma_{ip}}^2}\right) \tag{22}$$

$$P\left(Seq_{silence} \mid \neg E_{ip}\right)=\frac{2}{\sqrt{2\pi}\sigma_{nip}}\exp\left(-\frac{\left(Seq_{silence}-\mu_{nip}\right)^2}{2{\sigma_{nip}}^2}\right) \tag{23}$$

Where μ_{ip}, μ_{nip}, ${\sigma_{nip}}^2$ and ${\sigma_{ip}}^2$ denote the means and variances of the silence duration containing and not containing the IP, respectively.

3.3.2. Syllable-based acoustic features extraction

Acoustic features including duration, pitch, and energy for each syllable (Soltau et al. 2005) are adopted for IP detection. A feature vector of the syllables within an observation window around the silence is formed as the input of the GMM. That is, we are interested in the syllables around the silence that may appear as an IP. A window of $2w$ syllables with w syllables after and before $silence_k$ is used. First, the subscript will be translated according to the position of silence as $Syl_{n-k} \leftarrow Syl_n$. And we then extract the features of syllables within the observation windows.

Since the durations of syllables are not the same even for the same syllable, the duration ratio is defined as the average duration of the syllable normalized by the average duration over all syllables.

$$nf_{duration_i} \equiv \frac{\sum_{j=1}^{n_i} duration\left(syllable_{i.j}\right)}{\frac{|syllable|}{\sum_{i=1}} \sum_{j=1}^{n_i} duration\left(syllable_{i.j}\right)} \tag{24}$$

Where $syllable_{i,j}$ means the j-th samples of syllable i in the corpus. |syllable| means the number of the syllable. n_i is the number of syllable i in the corpus. Similarly, for energy and pitch, frame-based statistics are used to calculate the normalized features for each syllable.

Considering the result of speech recognition, the features are normalized to be the first order features. For modeling the speaking rate and variation in the energy and pitch during the utterance, the 2nd order feature called delta-duration, delta-energy and delta-pitch are obtained from the forward difference of the 1st order features. The following equation shows the estimation for delta-duration, which can also be applied for the estimation of delta-energy and delta-pitch.

$$\Delta nf_{duration_i} = \begin{cases} nf_{duration_{i+1}} - nf_{duration_i} & if \ -w < i < w \\ 0 & others \end{cases} \tag{25}$$

Where w is half of the observation window size. Totally, there are three kinds of two orders features after feature extraction. We combine these features to form a vector with $24w$-6 features to be the observation vector of the GMM. The acoustic features are denoted as the syllable-based observation sequence that corresponds to the potential IP, $silence_k$, by

$$\left\{O = \left[O_D, O_P, O_E\right] \in R^{\dim}\right\} \tag{26}$$

Where $O_s \in R^{\dim_s}$, $S \in \{D, P, E\}$ represents the single kind feature vectors and *dim* means the dimensions of the feature vector consisting of duration-related, pitch-related and energy-related features. The following equation shows the estimation for duration-related features.

$$O_D \equiv \begin{bmatrix} nf_{duration_{-w+1}}, ..., nf_{duration_{-1}}, nf_{duration_0}, nf_{duration_{+1}}, nf_{duration_{+2}}, ..., nf_{duration_{+w}} \\ \Delta nf_{duration_{-w+1}}, ..., \Delta nf_{duration_{-1}}, \Delta nf_{duration_0}, \Delta nf_{duration_{+1}}, \Delta nf_{duration_{+2}}, ..., \Delta nf_{duration_{+w-1}} \end{bmatrix}^T \quad (27)$$

3.3.3. Gaussian mixture model for interruption point detection

The GMM is adopted for IP detection using the acoustic features.

$$P\left(Seq_{syllable} \mid C_j\right) \equiv P\left(O_t \mid \lambda_j\right) = \sum_{i=1}^{W} \omega_i N\left(O_t; \mu_i, \Sigma_i\right) \quad (28)$$

Where $C_j = \{E_{ip}, \neg E_{ip}\}$ means the hypothesis set for $silence_k$ containing and not containing the IP. λ_j is the GMM for class C_j and ω_i is a mixture weight which must satisfy the constraint $\sum_{i=1}^{W} \omega_i = 1$, where W is the number of mixture components, and $N(\cdot)$ is the Gaussian density function:

$$N\left(O_t; \mu_i, \Sigma_i\right) = \frac{1}{(2\pi)^{\dim/2} \left|\Sigma_i\right|^{1/2}} \exp\left(-\frac{1}{2}\left(O_t - \mu_i\right)^T \Sigma_i^{-1} \left(O_t - \mu_i\right)\right) \quad (29)$$

where μ_i and Σ_i are the mean vector and covariance matrix of the *i*-th component. O_t denotes the *t*-th observation in the training corpus. The parameters $\theta = [\omega_i, \mu_i, \Sigma_i]$, $i = 1..M$ can be estimated iteratively using the EM algorithm for mixture *i*

$$\hat{\omega}_i = \frac{1}{N} \sum_{t=1}^{N} P\left(i \mid O_t, \lambda\right) \quad (30)$$

$$\hat{\mu}_i = \frac{\sum_{t=1}^{N} P\left(i \mid O_t, \lambda\right) O_t}{\sum_{t=1}^{N} P\left(i \mid O_t, \lambda\right)} \quad (31)$$

$$\hat{\Sigma}_i = \frac{\sum_{t=1}^{N} P\left(i \mid O_t, \lambda\right)\left(O_t - \hat{\mu}_i\right)\left(O_t - \hat{\mu}_i\right)^T}{\sum_{t=1}^{N} P\left(i \mid O_t, \lambda\right)} \quad (32)$$

Where $P(i \mid O_t, \lambda) = \dfrac{P(O_t \mid \lambda)\omega_i}{\sum_{j=1}^{W} P(O_t \mid \lambda)\omega_j}$ and N denote the total number of feature observations.

3.3.4. Potential interruption point extraction

Based on the assumption that IP appears generally at the boundary of two successive words, we can remove the detected IPs that do not appear in the word boundary. After the removal of unlikely IPs, the remaining IPs will be kept for further processing. Since the word graph or word lattice is obtained from speech recognition module, every path in the word graph or word lattice form its potential IP set for an input utterance.

3.4. Lingusitic processing for edit disfluency correction

In previous section, potential IPs has been detected from the acoustic features. However, correcting edit disfluency using the linguistic features is, in fact, one of the keys for rich transcription. In this section, the edit disfluency is detected by maximizing the likelihood of the language model for the cleaned-up utterances and the word correspondence between the deletable region and the correction given the position of the IP. Consider the word sequence W^* in the word lattice generated by the speech recognition engine. We can model the word string W^* using a log linear mixture model in which language model and alignment are both included.

$$\begin{aligned} W^* &= \arg\max_{W,IP} P(W;IP) \\ &= \arg\max_{W,IP} P(w_1, w_2, ...w_t, w_{t+1}, ...w_n, w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N; IP) \\ &= \arg\max_{W,n,t} \begin{pmatrix} P(w_1, w_2, ...w_t, w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N)^{\alpha} \\ \times P(w_{t+1}, ...w_n \mid w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N)^{(1-\alpha)} \end{pmatrix} \\ &= \arg\max_{W,n,t} \begin{pmatrix} \alpha \log\left(P(w_1, w_2, ...w_t, w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N)\right) \\ +(1-\alpha)\log\left(P(w_{t+1}, ...w_n \mid w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N)\right) \end{pmatrix} \end{aligned} \tag{33}$$

where α and $1-\alpha$ are the combination weight for cleanup language model and alignment model. *IP* means the interruption point obtained from the IP detection module and n is the position of the potential IP.

3.4.1. Language model of cleanup utterance

In the past, statistical language models have been applied to speech recognition and have achieved significant improvement in the recognition results. However, probability estimation of word sequences can be expensive and always suffers from the problem of data sparseness. In practice, the statistical language model is often approximated by the class-

based *n*-gram model with modified Kneser-Ney discounting probabilities for further smoothing.

$$\begin{aligned} &P\left(w_1, w_2, ...w_t, w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N\right) \\ &= \prod_{i=1}^{t} P\left(w_i \mid Class\left(w_1^{i-1}\right)\right) P\left(w_{n+1} \mid Class\left(w_1^{t}\right)\right) \prod_{j=n+2}^{N} P\left(w_j \mid Class\left(w_1^{t} w_{n+1}^{j-1}\right)\right) \end{aligned} \tag{34}$$

Where $Class(\cdot)$ means the conversion function that translates a word sequence into a word class sequence. In this section, we employ two word classes: semantic class and parts-of-speech (POS) class. A semantic class, such as the synsets in WordNet (http://wordnet.princeton.edu/) or concepts in the UMLS (http://www.nlm.nih.gov/research/umls/), contains the words that share a semantic property based on semantic relations, such as hyponym and hypernym. POS is called syntactic or grammatical categories defined as the role that a word plays in a sentence such as noun, verb, adjective… etc.

The other essential issue of *n*-gram model for correcting edit disfluency is the number of orders in Markov model. Since IP is the point at which the speaker breaks off the deletable region and the correction consists of the portion of the utterance that has been repaired by the speaker and can be considered fluent. By removing part of the word string will lead to a shorter string and result in the condition that higher probability is obtained for shorter word string. As a result, short word string will be favored. To deal with this problem, we can increase the order to constrain the perplexity and normalize the word length by aligning the deletable region and the correction.

3.4.2. Alignment model between the deletable region and the correction

In conversational speech, the structural pattern of a deletable region is usually similar to that of the correction. Sometimes, the deletable region appears as a substring of the correction. Accordingly, we can find the structural pattern in the starting point of the correction which generally follows the IP. Then, we can take the potential IP as the center and align the word string before and after it. Since the correction is used for replacing the deletable region and ending the utterance, there exists a correspondence between the words in the deletable region and the correction. We may, therefore, model the alignment assuming the conditional probability of the correction given the possible deletable region. According to this observation, class-based alignment is proposed to clean up edit disfluency. The alignment model can be described as

$$\begin{aligned} &P\left(w_{n+1}, ...w_{2n-t}, w_{2n-t+1}, ...w_N \mid w_{t+1}, ...w_n\right) \\ &= \prod_{k=t+1}^{n} \left(P\left(f_k \mid Class\left(w_k\right)\right) \prod_{l=1}^{f_k} P\left(Class\left(w_l\right) \mid Class\left(w_k\right)\right) \right) \prod_{k,l,m} P\left(l \mid k, m\right) \end{aligned} \tag{35}$$

where fertility f_k means the number of words in the correction corresponding to the word w_k in the deletable region. k and l are the positions of the words w_k and w_l in the

deletable region and the correction, respectively. m denotes the number of words in the deletable region. The alignment model for cleanup contains three parts: fertility probability, translation or corresponding probability and distortion probability. The fertility probability of word w_k is defined as

$$P\left(f_k \mid Class\left(w_k\right)\right) = \frac{\sum_{w_{i \in Class\left(w_k\right)}} \delta\left(f_i = f_k\right)}{\sum_{p=0}^{N} \sum_{w_{j \in Class\left(w_k\right)}} \delta\left(f_j = p\right)} \tag{36}$$

where $\delta(\cdot)$ is an indicator function and N means the maximum value of fertility. The translation or corresponding probability is measured according to (Wu et al. 1994).

$$P\left(Class\left(w_l\right) \mid Class\left(w_k\right)\right) = \frac{2 \times Depth\left(LCS\left(Class\left(w_l\right), Class\left(w_k\right)\right)\right)}{Depth\left(Class\left(w_l\right)\right) + Depth\left(Class\left(w_k\right)\right)} \tag{37}$$

where $Depth(\cdot)$ denotes the depth of the word class and $LCN(\cdot)$ denotes the lowest common subsumer of the words. The distortion probability $P(l \mid k, m)$ is the mapping probability of the word sequence between the deletable region and the correction.

3.5. Experimental results and discussion

To evaluate the performance of the proposed approach, a transcription system for spontaneous speech with edit dsifluencies in Mandarin was developed. A speech recognition engine using Hidden Markov Model Toolkit (HTK) was constructed as the syllable recognizer using 8 states (3 states for initial, and 5 states for final in Mandarin).

3.5.1. Experimental data

The Mandarin Conversational Dialogue Corpus (MCDC), collected from 2000 to 2001 at the Institute of Linguistics of Academia Sinica, Taiwan, consists of 30 digitized conversational dialogues of a total length of 27 hours. 60 subjects were randomly chosen from daily life in Taiwan area. It was annotated according to (Yeh et al. 2006) that gives concise explanations and detailed operational definitions of each tag in Mandarin. Corresponding to SimpleMDE, direct repetitions, partial repetitions, overt repairs and abandoned utterances are taken as edit disfluency in MCDC. The dialogs tagged as number 01, 02, 03 and 05 are used as the test corpus. For training the parameters in the speech recognizer, MAT Speech Database, TCC-300 and MCDC were employed.

3.5.2. Potential interruption point detection

According to the observation of the MCDC, the probability density function (pdf) of the duration of the silences with or without IPs is obtained. The average duration of the silences

with IP is larger than that of the silences without IP. According to this result, we can estimate the posterior probability of silence duration using a GMM for IP detection. For hypothesis testing, an anti-IP GMM is also constructed.

Since IP detection can be regarded as a position determination problem, an observation window over several syllables is adopted. In this observation window, the values of pitch and energy of the syllables just before an IP are usually larger than that after the IP. This phenomenon means the pitch reset and energy reset co-occur with IP in the edit disfluency. This generally happens in the syllables of the first word just after the IP. The pitch reset event is very obvious when the disfluency type is repair. Similar to the pitch, energy plays the same role when edit disfluency appears, but the effect is not so obvious compared to the pitch. The filler words or phrase after IP will be lengthened to strive for the time for the speaker to construct the correction and attract the listener to pay attention to. This factor can achieve significant improvement in IP detection rate.

The hypothesis testing, combined with the GMM model with four mixture components using the syllable features, will determine if the silence contains the IP. The parameter γ should be determined to achieve a better result. The overall IP error rate defined in RT'04F will be simply the average number of missed IP detections and falsely detected IPs per reference IP:

$$Error_{IP} = \frac{n_{M-IP} + n_{FA-IP}}{n_{IP}} \tag{38}$$

Where n_{M-IP} and n_{FA-IP} denote the numbers of missed and false alarm IPs respectively. n_{IP} means the number of reference IPs. We can adjust the threshold γ for n_{M-IP} and n_{FA-IP}.

Since the goal of the IP detection module is to detect the potential IPs, false alarm for IP detection is not a serious problem compared to miss error. That is to say, we want to obtain high recall rate without much increase in false alarm rate. Finally, the threshold γ was set to 0.25. Since the IP always appears in word boundary, this constraint can be used to remove unlikely IPs.

3.5.3. Clean-up disfluency using linguistic information

For evaluating the edit disfluency correction model, two different types of transcriptions were used: human generated transcription (REF) and speech-to-text recognition output (STT). Using the reference transcriptions provides the best case for the evaluation of the edit disfluency correction module because there are no word errors in the transcription. For practicability, the syllable lattice from speech recognition is fed to the edit disfluency correction module for performance assessment.

For class-based approach, part of speech (POS) and semantic class are employed as the word class. Herein, semantic class is obtained based on Hownet (http://www.keenage.com/) that

defines the relation "IS-A" as the primary feature. There are 26 and 30 classes in POS class and semantic class respectively. By this, we can categorize the words according to their hypernyms or concepts, and every word can map to its own semantic class.

The edit word detection (EWD) task is to detect the regions of the input speech containing the words in the deletable regions. One of the primary metrics for edit disfluency correction is to use the edit word detection method defined in RT'04F (Chen et al. 2002), which is similar to the metric for IP detection shown in Eq. (38).

Due to the lack of structural information, unigram does not obtain any improvement. Bigram provides more significant improvement combined with POS class-based alignment than semantic class-based alignment. Using 3-gram and semantic class-based alignment outperforms other combinations. The reason is that 3-gram with more strict constraints can reduce the false alarm rate for edit word detection. In fact, we also tried using 4-gram to gain more improvement than 3-gram, but the excess computation makes the light improvement not conspicuous as we expected. Besides, the statistics of 4-gram is too spare compared to 3-gram model. The best combination in edit disfluency correction module is 3-gram and semantic class.

According to the analysis of the results shown in Table 2, we can find the values of the probabilities of the *n*-gram model are much smaller than that of the alignment model. Since the alignment can be taken as the penalty for edit words, we should balance the effects between the 3-gram and the alignment with semantic class using a log linear combination weight α. For optimizing the performance, we estimate α empirically based on the minimization of the edit word errors.

	Human generated transcription (REF)			**Speech-to-text recognition output (STT)**		
	$\frac{n_{M-EWD}}{n_{EWD}}$	$\frac{n_{FA-EWD}}{n_{EWD}}$	$Error_{EWD}$	$\frac{n_{M-EWD}}{n_{EWD}}$	$\frac{n_{FA-EWD}}{n_{EWD}}$	$Error_{EWD}$
1-gram+alignment[1]	0.15	0.17	0.32	0.58	0.65	1.23
1-gram+alignment[2]	0.23	0.12	0.35	0.62	0.42	1.04
2-gram+alignment[1]	0.09	0.15	0.24	0.46	0.43	0.87
2-gram+alignment[2]	0.10	0.11	0.21	0.38	0.36	0.74
3-gram+alignment[1]	0.12	0.04	0.16	0.39	0.23	0.62
3-gram+alignment[2]	**0.11**	**0.04**	**0.15**	0.36	0.24	0.60

[1]: word class based on the part of speech (POS) [2]: word class based on the semantic class

Table 2. Results (%) of linguistic module with equal weight $\alpha = (1-\alpha) = 0.5$ for edit word detection on REF and STT conditions

3.6. Conclusion and future work

This investigation has proposed an approach to edit disfluency detection and correction for rich transcription. The proposed theoretical approach, based on a two stage process, aims to model the behavior of edit disfluency and cleanup the disfluency. IP detection module using hypothesis testing from the acoustic features is employed to detect the potential IPs. Word-based linguistic module consists of a cleanup language model and an alignment model is used for verifying the position of the IP and therefore correcting the edit disfluency. Experimental results indicate that the IP detection mechanism is able to recall IPs by adjusting the threshold in hypothesis testing. In an investigation of the linguistic properties of edit disfluency, the linguistic module was explored for correcting disfluency based on the potential IPs. The experimental results indicate a significant improvement in performance was achieved. In the future, this framework will be extended to deal with the problem resulted from subword to improve the performance of the rich transcription system.

4. Speech recognition in multilingual environment

This section presents an approach to generating phonetic units for mixed-language or multilingual speech recognition. Acoustic and contextual analysis is performed to characterize multilingual phonetic units for phone set creation. Acoustic likelihood is utilized for similarity estimation of phone models. The hyperspace analog to language (HAL) model is adopted for contextual modeling and contextual similarity estimation. A confusion matrix combining acoustic and contextual similarities between every two phonetic units is built for phonetic unit clustering. Multidimensional scaling (MDS) method is applied to the confusion matrix for reducing dimensionality.

4.1. Introduction

In multilingual speech recognition, it is very important to determine a global phone inventory for different languages. When an authentic multilingual phone set is defined, the acoustic models and pronunciation lexicon can be constructed (Chen et al. 2002). The simplest approach to phone set definition is to combine the phone inventories of different languages together without sharing the units across the languages. The second one is to map language-dependent phones to the global inventory of the multilingual phonetic association based on phonetic knowledge to construct the multilingual phone inventory. Several global phone-based phonetic representations such as International Phonetic Alphabet (IPA) (Mathews 1979), Speech Assessment Methods Phonetic Alphabet (Wells 1989) and Worldbet (Hieronymus 1993) are generally used. The third one is to merge the language-dependent phone models using a hierarchical phone clustering algorithm to obtain a compact multilingual inventory. In this approach, the distance measure between acoustic models, such as Bhattacharyya distance (Mak et al. 1996) and Kullback-Leibler (KL) divergence (Goldberger et al. 2005), is employed to perform the bottom-up clustering. Finally, the multilingual phone models are generated with the use of a phonetic top-down clustering procedure (Young et al. 1994).

4.2. Multilingual phone set definition

From the viewpoint of multilingual speech recognition, a phonetic representation is functionally defined by the mapping of the fundamental phonetic units of languages to describe the corresponding pronunciation. In this section, IPA-based multilingual phone definition is suitable and consistent for phonetic representation. Using phonetic representation of the IPA, the recognition units can be effectively reduced for multilingual speech recognition. Considering the co-articulated pronunciation, context-dependent triphones are adopted in the expansion of IPA-based phonetic units.

In multilingual speech recognition, misrecognition generally results from incorrect pronunciation or confusable phonetic set. For examples, in Mandarin speech, the "ei_M" and "zh_M" is usually pronounced as "en_M" and "z_M", respectively. In this section, statistical methods are proposed to deal with the problem of misrecognition caused by the confusing characteristics between phonetic units in multilingual speech recognition. Based on the analysis of confusing characteristics, confusing phones due in part to the confusable phonetic representation are redefined to alleviate the misrecognition problem.

4.2.1. Acoustic likelihood

For the estimation of the confusion between two phone models, the posterior probabilities obtained from the phone-based hidden Markov model (HMM) are employed. Given two phone models, ω_k and ω_l, trained with the corresponding training data, x_i^k, $1 \le i \le I$ and x_j^l, $1 \le j \le J$, the symmetric acoustic likelihood (ACL) between two phone models, ω_k and ω_l, are estimated as follows.

$$a_{k,l} = \frac{\sum_{i=1}^{I} P(x_i^l \mid \omega_k) + \sum_{j=1}^{J} P(x_j^k \mid \omega_l)}{I + J} \tag{39}$$

where I and J represent the number of training data for phone models, ω_k and ω_l, respectively. The acoustic confusing matrix $\mathrm{A} = (a_{k,l})_{N \times N}$ is obtained from the pairwise similarities between every two phone models, and N denotes the number of phone models.

4.2.2. Contextual analysis

A co-articulation pattern can be considered as a semantically plausible combination of phones. This section presents a text mining framework to automatically induce co-articulation patterns from a mixed-language or a multilingual corpus. A crucial step to induce the co-articulation patterns is to represent speech intonation as well as combination of phones. To achieve this goal, the hyperspace analog to language (HAL) model constructs a high-dimensional contextual space for the mixed-language or multilingual corpus. Each context-dependent triphone in the HAL space is represented as a vector of its context

phones, which represents that the sense of a phone can be co-articulated through its context phones. Such notion is derived from the observation of articulation behavior. Based on the co-articulation behavior, if two phones share more common context, they are more similarly articulated.

The HAL model represents the multilingual triphones based on a vector representation. Each dimension of the vector is a weight representing the strength of association between the target phone and its context phone. The weights are computed by applying an observation window of length ℓ over the corpus. All phones within the window are considered as the co-articulated pronunciation with each other. For any two phones of distance d within the window, the weight between them is defined as $\ell - d + 1$. After moving the window by one phone increment over the sentence, the HAL space $\mathrm{G} = (g_{k,l})_{N \times N}$ is constructed. The resultant HAL space is an $N \times N$ matrix, where N is the number of triphones.

Table 3 presents the HAL space for the example of English and Mandarin mixed sentence "查一下<look up> (CH A @ I X I A) Baghdad (B AE G D AE D)." For each phone in Table 3, the corresponding row vector represents its left contextual information, i.e. the weights of the phones preceding it. The corresponding column vector represents its right contextual information. $w_{k,l}$ indicates the *k*-th weight of the *l*-th triphone φ_l. Furthermore, the weights in the vector are re-estimated as described as follows.

$$\overline{w}_{k,l} = w_{k,l} \times \log \frac{N}{N_l} \tag{40}$$

where *N* denotes the total number of phone vectors and N_l represents the number of vectors of phone φ_l with nonzero dimension. After each dimension is re-weighted, the HAL space is transformed into a probabilistic framework, and thus each weight can be redefined as

$$\hat{w}_{k,l} = \frac{\overline{w}_{k,l}}{\sum_{k=1}^{N} \overline{w}_{k,l}} \tag{41}$$

To generate a symmetric matrix, the weight is averaged as

$$g_{k,l} = \frac{\hat{w}_{k,l} + \hat{w}_{l,k}}{2}, \quad 1 \le k, l \le N$$

4.2.3. Fusion of confusing matrices and dimensional reduction

The multidimensional scaling (MDS) method is used to project multilingual triphones to the orthogonal axes where the ranking distance relation between them can be estimated using Euclidean distance. MDS is generally a procedure which characterizes the data in terms of a matrix of pairwise distances using Euclidean distance estimation. One of the purposes of

MDS is to reduce the data dimensionality into a low-dimensional space. The IPA-based phone alphabet is 55 for English and Mandarin. This makes around 166,375 ($55 \times 55 \times 55$) triphone numbers. When the number of target languages is increased, the dimension of the confusing matrix becomes huge. Another purpose of multidimensional scaling is to project the elements in the matrix to the orthogonal axes where the ranking distance relation between elements in the confusion matrix can be estimated. Compared to the hierarchical clustering method (Mak et al. 1996), this section applies MDS to the global similarity measure of multilingual triphones.

	CH	A	@	I	X	B	AE	G	D
CH									
A	3			4	1				
@	2	3							
I	1	2	4		3				
X		1	2	3					
B		3		2	1				
AE		2		1		3		2	3
G		1				2	3		
D						1	5	4	

Table 3. Example of multilingual sentence "查一下<look up> (CH A @ I X I A) Baghdad (B AE G D AE D)" in HAL space

In this section, the multidimensional scaling method suitable to represent the high dimensionality relation is adopted to project the confusing characteristic of multilingual triphones onto a lower-dimensional space for similarity estimation. Multidimensional scaling approach is similar to the principal component analysis (PCA) method. The difference is that MDS focuses on the distance relation between any two variables and PCA focuses on the discriminative principal component in variables. MDS is applied for estimating the similarity of pairwise triphones. The similarity matrix $\mathrm{V} = (v_{k,l})_{N \times N}$ contains pairwise similarities between every two multilingual triphones. The element of row k and column l in the similarity matrix is computed as

$$v_{k,l} = -(\alpha \times \log(a_{k,l}) + (1-\alpha) \times \log(g_{k,l})) \qquad 1 \le k,l \le N \tag{42}$$

where α denotes the combination weight. The sum rule of data fusion is indicated to combine acoustic likelihood (ACL) and contextual analysis (HAL) confusing matrices as shown in Figure 5.

MDS is then adopted to project the triphones onto the orthogonal axes where the ranking distance relation between triphones can be estimated based on the similarity matrices of triphones. The first step of MDS is to obtain the following matrices

$$\mathrm{B} = \mathrm{HSH} \tag{43}$$

where $\mathrm{H} = \mathrm{I} - \frac{1}{n}11'$ is the centralized matrix. I indicates the diagonal matrix and 1 means the indicator vector. The elements in matrix B is computed as

$$b_{kl} = s_{kl} - \bar{s}_{k\bullet} - \bar{s}_{\bullet l} - \bar{s}_{\bullet\bullet} \tag{44}$$

where

$$\bar{s}_{k\bullet} = \sum_{l=1}^{N} \frac{s_{kl}}{N} \tag{45}$$

is the average similarity values over the k^{th} row,

$$\bar{s}_{\bullet l} = \sum_{k=1}^{N} \frac{s_{kl}}{N} \tag{46}$$

denotes the average similarity values over the l^{th} column, and

$$\bar{s}_{\bullet\bullet} = \sum_{k=1}^{N}\sum_{l=1}^{N} \frac{s_{kl}}{N^2} \tag{47}$$

are the average similarity values over all rows and columns of the matrix B. The eigenvector analysis is applied to matrix B to obtain the axis of each triphone in a low dimension. The singular value decomposition (SVD) is applied to solve the eigenvalue and eigenvector problems. Afterwards, the first z nonzero eigenvalues for each phone in a descending order, i.e. $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_z > 0$, is obtained. The corresponding ordered eigenvectors are denoted as u. Then, each triphone is represented by a projected vector as

$$\mathrm{Y} = [\sqrt{\lambda_1}\mathrm{u}_1, \sqrt{\lambda_2}\mathrm{u}_2, \ldots, \sqrt{\lambda_z}\mathrm{u}_z] \tag{48}$$

4.2.4. Phone clustering

This section presents how to cluster the triphones with similar acoustic and contextual properties into a multilingual triphone cluster. Cosine measure between triphones Y_k and Y_l is adopted as follows.

$$C(\mathrm{Y}_k, \mathrm{Y}_l) = \frac{\overline{y_k} \bullet \overline{y_l}}{\left\|\overline{y_k}\right\| \cdot \left\|\overline{y_l}\right\|} = \frac{\sum_{i=1}^{z} y_{k,i} \times y_{l,i}}{\sqrt{\sum_{i=1}^{z} y_{k,i}^2} \times \sqrt{\sum_{i=1}^{z} y_{l,i}^2}} \tag{49}$$

where $y_{k,i}$ and $y_{l,i}$ are the element of the triphone vectors Y_k and Y_l. The modified *k*-means (MKM) algorithm is applied to cluster all the triphones into a compact phonetic set. The convergence of closeness measure is determined by a pre-set threshold.

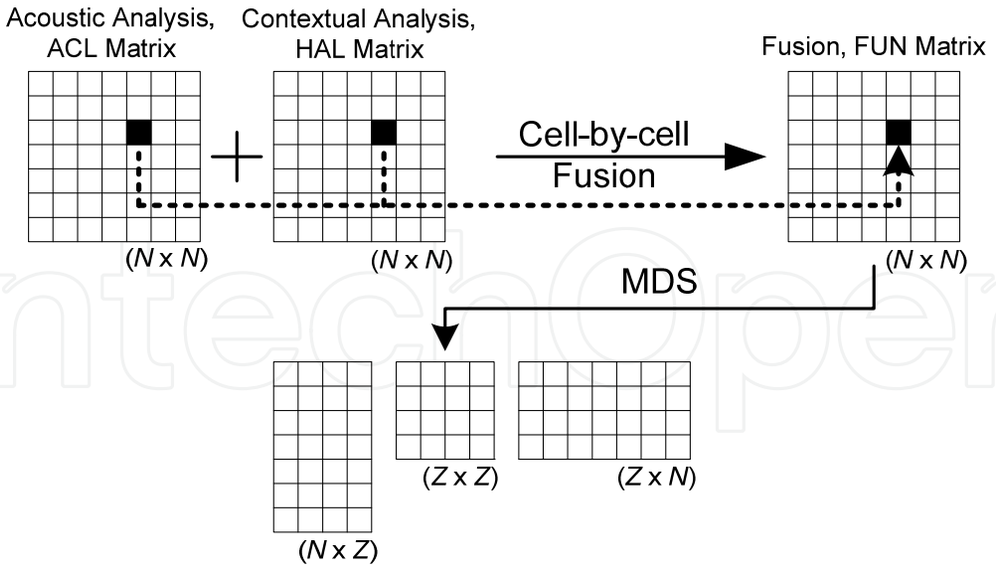

Figure 5. An illustration of fusion of acoustic likelihood (ACL) and contextual analysis (HAL) confusing matrices for the MDS process

4.3. Experimental evaluations

For evaluation, an in-house multilingual speech recognizer was implemented and experiments were conducted to evaluate the performance of the proposed approach on an English-Mandarin multilingual corpus.

4.3.1. Multilingual database

In Taiwan, English and Mandarin are popular in conversation, culture, media, and everyday life. For bilingual corpus collection, the English across Taiwan (EAT) project (EAT [online] http://www.aclclp.org.tw/) sponsored by National Science Council, Taiwan prepared 600 recording sheets. Each sheet contains 80 reading sentences, including English long sentences, English short sentences, English words and mixed English and Mandarin sentences. Each sheet was used for speech recording individually for English-major students and non-English-major students. Microphone corpus was recorded as sound files with 16 kHz sampling rate and 16 bit sample resolution. The summarized recording information of EAT corpus is shown in Table 4. In this section, we applied mixed English-Mandarin sentences in microphone application. The average sentence length is around 12.62 characters.

	English-Major		Non-English-Major	
	male	female	male	female
No. of Sentences	11,977	30,094	25,432	15,540
No. of Speakers	166	406	368	224

Table 4. EAT-MIC Multilingual Corpus Information

4.3.2. Evaluation of the phone set generation based on acoustic and contextual analysis

In this section, the phone recognition rate was adopted for the evaluation of acoustic modeling accuracy. Three classes of speech recognition errors, including insertion errors (*Ins*), deletion errors (*Del*) and substitution errors (*Sub*), were considered. This section applied the fusion of acoustic and contextual analysis approaches to generating the multilingual triphone set. Since the optimal clustering number of acoustic models was unknown, several sets of HMMs were produced by varying the MKM convergence threshold during multilingual triphone clustering. There are three different approaches including acoustic likelihood (ACL), contextual analysis (HAL) and fusion of acoustic and contextual analysis (FUN). It is evident that the proposed fusion method achieves a better result than individual ACL or HAL methods. The comparison of acoustic analysis and contextual analysis, HAL achieves a higher recognition rate than ACL. It denotes that contextual analysis is more significant than acoustic analysis for multilingual confusing phone clustering. The curves shows that phone accuracy will increase with the increase in state number, and finally decrease due to the confusing triphone definition and the requirement of a large size of multilingual training corpus. The proposed multilingual phone generation approach can get an improved performance than the ordinary multilingual triphone sets. In this section, the English and Mandarin triphone sets is defined based on the expansion of the IPA definition. The multilingual speech recognition system for English and Mandarin contains 924 context-dependent triphone models. The best phone recognition accuracy was 67.01% for the HAL window size = 3. Therefore, this section applied this setting in the following experiments.

4.3.3. Comparison of acoustic and language models for multilingual speech recognition

Table 5 shows the comparisons on different acoustic and language models for multilingual speech recognition. For the comparison of monophone and triphone-based recognition, different phone inventory definitions including direct combination of language-dependent phones (MIX), language-dependent IPA phone definition (IPA), tree-based clustering procedure (TRE) (Mak et al. 1996) and the proposed methods (FUN) were considered. The phonetic units of Mandarin can be represented as 37 fundamental phones and English can be represented as 39 fundamental phones. The phone set for the direct combination of English and Mandarin is 78 phones with two silence models. The phone set for IPA definition of English and Mandarin contains 55 phones.

	Monophone		**Triphone**	
	MIX	IPA	TRE	FUN
Phone models	78	55	1172	924
With language model	45.81%	66.05%	76.46%	78.18%
Without language model	32.58%	51.98%	65.32%	67.01%

Table 5. Comparison of acoustic and language models for multilingual speech recognition

In acoustic comparison, multilingual context-independent (MIX and IPA) and context-dependent (TRE and FUN) phone sets were investigated. With the language model of English and Mandarin, the approach based on MIX achieved 45.81% phone accuracy and the IPA method achieved 66.05% phone accuracy. The IPA performance is evidently better than MIX approach. TRE method achieved 76.46% phone accuracy and our proposed approach achieved 78.18%. It is obvious that triphone models achieved better performance than monophone models. There is around 2.25% relative improvement from 76.46% accuracy for the baseline system based on TRE to 78.18% accuracy for the approach using acoustic and contextual analysis.

In order to evaluate the acoustic modeling performance, the experiments were conducted without using language model. Without the language model, the MIX approach achieved 32.58%, IPA method achieved 51.98%, TRE method achieved 65.32%, and the proposed approach achieved 67.01% phone accuracies. In conclusion, multilingual speech recognition can obtain the best performance using FUN approach for the context-dependent phone definition with language model.

4.3.4. Comparison of monolingual and multilingual speech recognition

In this experiment, the utterances of English word and English sentence in the EAT corpus were collected for the evaluation of monolingual speech recognition. A comparison of monolingual and multilingual speech recognition using EAT corpus was shown in Table 6. Totally, 2496 English words, 3072 English sentences and 5884 mixed English and Mandarin utterances were separately used for training. Other 200 utterances were applied for evaluation. In the context-dependent without language model condition, the performance of monolingual English word achieved 76.25% which is higher than 67.42% for monolingual English sentences. The phone recognition accuracy of monolingual English sentences is 67.42% slightly better than 67.01% for mixed English and Mandarin sentences.

	Monolingual		**Multilingual**
	English word	English sent.	English and Mandarin mixed sent.
Training corpus	2496	3072	5884
Phone recognition accuracy	76.25%	67.42%	67.01%

Table 6. Comparison of monolingual and multilingual speech recognition

4.4. Conclusions

In this section, the fusion of acoustic and contextual analysis is proposed to generate phonetic units for mixed-language or multilingual speech recognition. The context-dependent triphones are defined based on the IPA representation. Furthermore, the confusing characteristics of multilingual phone sets are analyzed using acoustic and contextual information. From the acoustic analysis, the acoustic likelihood confusing matrix

is constructed by the posterior probability of triphones. From the contextual analysis, the hyperspace analog to language (HAL) approach is employed. Using the multidimensional scaling and data fusion approaches, the combination matrix is built and each phone is represented as a vector. Furthermore, the modified *k*-means algorithm is used to cluster the multilingual triphones into a compact and robust phone set. Experimental results show that the proposed approach gives encouraging results.

5. Conclusions

In this chapter speech recognition techniques in adverse environments are presented. For speech recognition in noisy environments, two approaches to cepstral feature enhancement for noisy speech recognition using noise-normalized stochastic vector mapping are described. Experimental results show that the proposed approach outperformed the SPLICE-based approach without stereo data on AURORA2 database. For speech recognition in disfluent environments, an approach to edit disfluency detection and correction for rich transcription is presented. The proposed theoretical approach, based on a two stage process, aims to model the behavior of edit disfluency and cleanup the disfluency. Experimental results indicate that the IP detection mechanism is able to recall IPs by adjusting the threshold in hypothesis testing. For speech recognition in multilingual environments, the fusion of acoustic and contextual analysis is proposed to generate phonetic units for mixed-language or multilingual speech recognition. The confusing characteristics of multilingual phone sets are analyzed using acoustic and contextual information. The modified k-means algorithm is used to cluster the multilingual triphones into a compact and robust phone set. Experimental results show that the proposed approach improves recognition accuracy in multilingual environments.

Author details

Chung-Hsien Wu* and Chao-Hong Liu
Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, Taiwan, R.O.C.

Acknowledgement

This work was partially supported by NCKU Project of Promoting Academic Excellence & Developing World Class Research Centers.

6. References

Bear, J., J. Dowding and E. Shriberg (1992). *Integrating multiple knowledge sources for detection and correction of repairs in human-computer dialog*. *Proc. of ACL*. Newark, Deleware, USA, Association for Computational Linguistics: 56-63.

* Corresponding Author

Benveniste, A., M. Métivier and P. Priouret (1990). *Adaptive Algorithms and Stochastic Approximations*. *Applications of Mathematics*. New York, Springer. 22.

Boll, S. (1979). Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. 27. No. 2. pp. 113-120.

Charniak, E. and M. Johnson (2001). *Edit detection and parsing for transcribed speech*. *Proc. of NAACL*, Association for Computational Linguistics: 118-126.

Chen, Y. J., C. H. Wu, Y. H. Chiu and H. C. Liao (2002). Generation of robust phonetic set and decision tree for Mandarin using chi-square testing. *Speech Communication*, Vol. 38. No. 3-4. pp. 349-364.

Deng, L., A. Acero, M. Plumpe and X. Huang (2000). Large-vocabulary speech recognition under adverse acoustic environments. *Proc. ICSLP-2000*, Beijing, China.

Deng, L., J. Droppo and A. Acero (2003). Recursive estimation of nonstationary noise using iterative stochastic approximation for robust speech recognition. *Speech and Audio Processing, IEEE Transactions on*, Vol. 11. No. 6. pp. 568-580.

Furui, S., M. Nakamura, T. Ichiba and K. Iwano (2005). Analysis and recognition of spontaneous speech using Corpus of Spontaneous Japanese. *Speech Communication*, Vol. 47. No. 1-2. pp. 208-219.

Gales, M. J. F. and S. J. Young (1996). Robust continuous speech recognition using parallel model combination. *IEEE Transactions on Speech and Audio Processing*, Vol. 4. No. 5. pp. 352-359.

Goldberger, J. and H. Aronowitz (2005). *A distance measure between gmms based on the unscented transform and its application to speaker recognition*. *Proc. of EUROSPEECH*. Lisbon, Portugal: 1985-1988.

Hain, T., P. C. Woodland, G. Evermann, M. J. F. Gales, X. Liu, G. L. Moore, D. Povey and L. Wang (2005). Automatic transcription of conversational telephone speech. *IEEE Transactions on Speech and Audio Processing*, Vol. 13. No. 6. pp. 1173-1185.

Heeman, P. A. and J. F. Allen (1999). Speech repairs, intonational phrases, and discourse markers: modeling speakers' utterances in spoken dialogue. *Computational Linguistics*, Vol. 25. No. 4. pp. 527-571.

Hermansky, H. and N. Morgan (1994). RASTA processing of speech. *IEEE Transactions on Speech and Audio Processing*, Vol. 2. No. 4. pp. 578-589.

Hieronymus, J. L. (1993). ASCII phonetic symbols for the world's languages: Worldbet. *Journal of the International Phonetic Association*, Vol. 23.

Hsieh, C. H. and C. H. Wu (2008). Stochastic vector mapping-based feature enhancement using prior-models and model adaptation for noisy speech recognition. *Speech Communication*, Vol. 50. No. 6. pp. 467-475.

Huang, C. L. and C. H. Wu (2007). Generation of phonetic units for mixed-language speech recognition based on acoustic and contextual analysis. *IEEE Transactions on Computers*, Vol. 56. No. 9. pp. 1225-1233.

Johnson, M. and E. Charniak (2004). *A TAG-based noisy channel model of speech repairs*. *Proc. of ACL*, Association for Computational Linguistics: 33-39.

Kohler, J. (2001). Multilingual phone models for vocabulary-independent speech recognition tasks. *Speech Communication*, Vol. 35. No. 1-2. pp. 21-30.

Liu, Y., E. Shriberg, A. Stolcke and M. Harper (2005). *Comparing HMM, maximum entropy, and conditional random fields for disfluency detection*. *Proc. of Eurospeech*: 3313-3316.

Macho, D., L. Mauuary, B. Noé, Y. M. Cheng, D. Ealey, D. Jouvet, H. Kelleher, D. Pearce and F. Saadoun (2002). Evaluation of a noise-robust DSR front-end on Aurora databases. *Proc. ICSLP-2002*, Denver, Colorado, USA.

Mak, B. and E. Barnard (1996). *Phone clustering using the Bhattacharyya distance*. *Proc. ICSLP*, IEEE. 4: 2005-2008.

Mathews, R. H. (1979). *Mathews' Chinese-English Dictionary*, Harvard university press.

Savova, G. and J. Bachenko (2003). *Prosodic features of four types of disfluencies*. *Proc. of DiSS*: 91–94.

Shriberg, E., L. Ferrer, S. Kajarekar, A. Venkataraman and A. Stolcke (2005). Modeling prosodic feature sequences for speaker recognition. *Speech Communication*, Vol. 46. No. 3-4. pp. 455-472.

Shriberg, E., A. Stolcke, D. Hakkani-Tur and G. Tur (2000). Prosody-based automatic segmentation of speech into sentences and topics. *Speech Communication*, Vol. 32. No. 1-2. pp. 127-154.

Snover, M., B. Dorr and R. Schwartz (2004). *A lexically-driven algorithm for disfluency detection*. *Proc. of HLT/NAACL*, Association for Computational Linguistics: 157-160.

Soltau, H., B. Kingsbury, L. Mangu, D. Povey, G. Saon and G. Zweig (2005). The IBM 2004 conversational telephony system for rich transcription. *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, Philadelphia, USA.

Waibel, A., H. Soltau, T. Schultz, T. Schaaf and F. Metze (2000). Multilingual Speech Recognition. *Verbmobil: foundations of speech-to-speech translation*, Springer-Verlag.

Wells, J. C. (1989). Computer-coded phonemic notation of individual languages of the European Community. *Journal of the International Phonetic Association*, Vol. 19. No. 1. pp. 31-54.

Wu, C. H., Y. H. Chiu, C. J. Shia and C. Y. Lin (2006). Automatic segmentation and identification of mixed-language speech using delta-BIC and LSA-based GMMs. *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14. No. 1. pp. 266-276.

Wu, C. H. and G. L. Yan (2004). Acoustic Feature Analysis and Discriminative Modeling of Filled Pauses for Spontaneous Speech Recognition. *Journal of VLSI Signal Processing Systems*, Vol. 36. No. 2. pp. 91-104.

Wu, J. and Q. Huo (2002). An environment compensated minimum classification error training approach and its evaluation on Aurora2 database. *Proc. ICSLP-2002*, Denver, Colorado, USA.

Wu, Z. and M. Palmer (1994). *Verbs semantics and lexical selection*. *Proc. 32nd ACL*, Association for Computational Linguistics: 133-138.

Yeh, J. F. and C. H. Wu (2006). Edit disfluency detection and correction using a cleanup language model and an alignment model. *IEEE Transactions on Audio, Speech, and Language Processing*, Vol. 14. No. 5. pp. 1574-1583.

Young, S. J., J. Odell and P. Woodland (1994). *Tree-based state tying for high accuracy acoustic modelling*. *Proc. ARPA Human Language Technology Conference*. Plainsboro, USA, Association for Computational Linguistics: 307-312.

IntechOpen