IMPROVEMENTS ON SPEECH RECOGNITION FOR FAST TALKERS

Author :M. Richardson,M.Hwang,A. Acero, and X.D. Huang

Professor:陳嘉平
Repotor:葉佳璋

outline

- Introduction
- System Description and speech corpora Support
- Speak rate Determination
- The CLN Algorithm
- Experimental results
- Conclusion

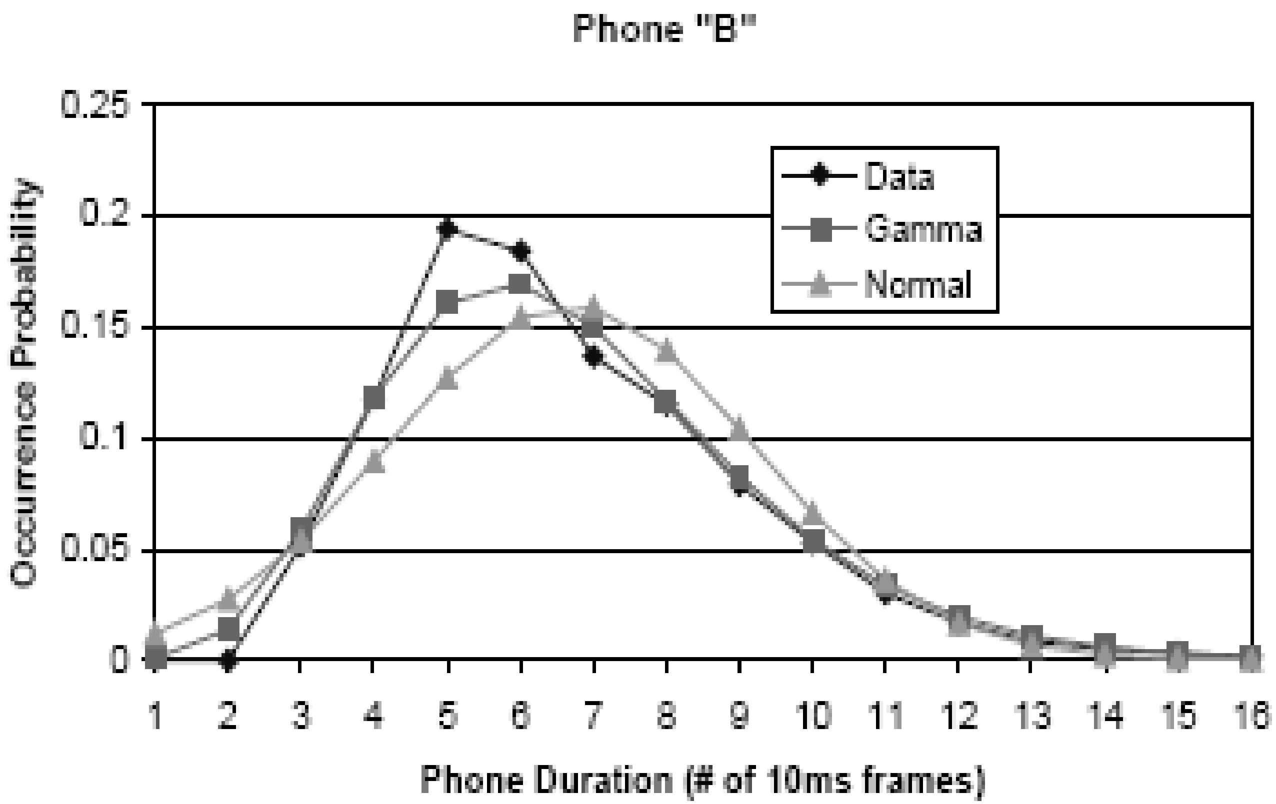
Introduce

- Cepstrum length normalization(CLN)
- Improvements made by CLN and Maximum Likelihood Linear Regression
- Improvement by using shorter window shift in computing cepstra

System Description and speech corpora

- The speaker independent system built here consists of 6000 gender-dependent context.
- The feature used were 12 mel-frequency cepstrum coefficients(MFCC).
- Log energy and their fist and second order differences in 10ms time frames.
- The speaker independent acoustic training corpus comes from the 284-speaker(SI-284)

Speak rate Determination



Stretch the utterance

- Phone-by-phone Length Stretching

- Sentence-by-Sentence Length Streching

Phone-by-phone Length Stretching

$$\Gamma(x,\alpha,\beta)=\frac{\beta^{\alpha}x^{\alpha-1}e^{-x\beta}}{\Gamma(\alpha)}$$

Gamna distribution $\Gamma(x,\alpha_i,\beta_i)$

Mean $\mu_i=\dfrac{\alpha_i}{\beta_i}$ variance $\dfrac{\alpha_i}{\beta_i^2}$

Length-stretching factor $\rho_i=\dfrac{peak_i}{l_i}$

$$peak_i=\frac{\alpha_i-1}{\beta_i}$$

Sentence-by-Sentence Length Stretching

$$\tilde{\rho} = \underset{\rho}{\operatorname{argmax}} \{ p(\rho l_1 \mid \Gamma_1) p(\rho l_2 \mid \Gamma_2) \ldots p(\rho l_n \mid \Gamma_n) \}$$

$$\rho = \frac{\sum_{i=1}^{n} \alpha_i}{\sum_{i=1}^{n} \beta_i l_i}$$

Average peak $\rho = \frac{1}{n} \sum_{i=1}^{n} \rho_i$

The CLN Algorithm

- Inserting/dropping frames uniformly in the speech segment.
- Repeating/deleting represent the steady state of each phone segment.
- Creating new frames by interpolating neighboring frames

Experimental resuts

- CLN on the test data of fast Speech

- CLN on the test data of normal speech

CLN on the test data of fast Speech

Training data \ test data	Original	Interpolation
Original	16.64%	13.90%

Table 1. Word error rates on dev-fast with and without MFCC interpolation.

CLN on the test data of normal Speech

Regular		*H1dev94*	
Original MFCCs	MFCC Interpolation	Original MFCCs	MFCC Interpolation
8.36%	8.20%	8.71%	8.78%

Table 2: Word error rates on the *regular* and *h1dev94* data sets.

Evaluation and MLLR

Original MFCC	MFCC Interpolation	MLLR on Gaussian Means	MFCC Interpolation + MLLR
18.34%	15.91%	16.03%	14.03%

Table 3: Word error rates on the *eval-fast* set. Combining MFCC interpolation and MLLR speaker adaptation yielded 23.5 % error rate reduction.

figure2

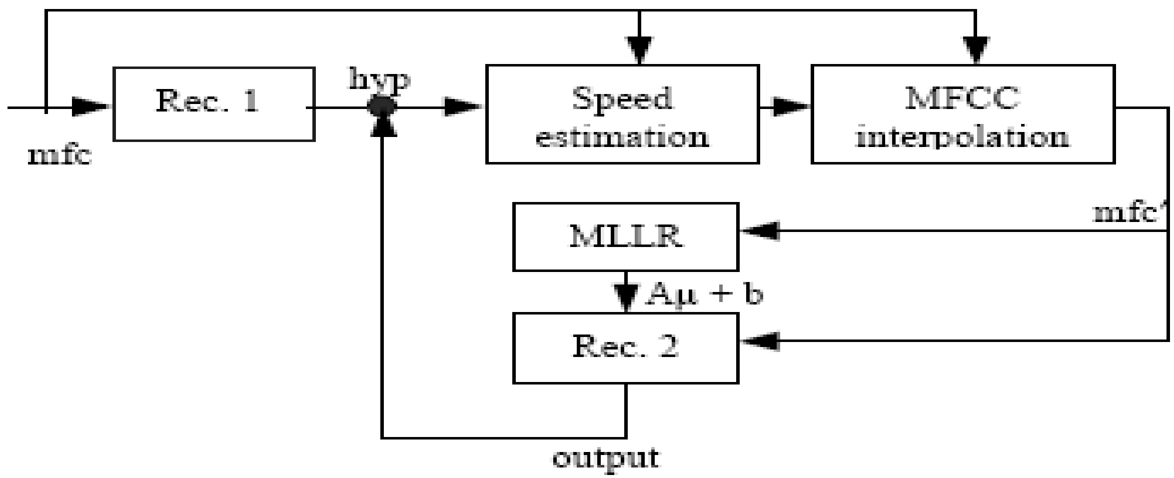


Figure 2: Combination of MFCC interpolation and MLLR adaptation.

Shringking Hamming Window Shift

Use a smaller window shift in generating the cepstrum

$$s' = \frac{s}{\rho}$$

New window shift is inversely proportional to speed factor