UNIVERSITY OF CAMBRIDGE

Joint Optimisation of Tandem Systems using Gaussian Mixture Density Neural Network Discriminative Sequence Training

Chao Zhang and Phil Woodland

March 8, 2017

Cambridge University Engineering Department

Introduction

Tandem Systems as Mixture Density Neural Networks (MDNNs)

- Tandem systems model features produced by DNN using GMMs
- A bottleneck (BN) DNN and GMMs combine to form an MDNN

Importance of Tandem Systems

- A general framework for modelling non-Gaussian distributions
- Can apply GMM techniques (e.g., adaptation) to improve MDNNs
- Tandem and hybrid systems produce complementary errors

Weakness of Conventional Tandem Systems

- GMMs and DNN are independently estimated→suboptimal

Introduction

Can Tandem and Hybrid Systems Have Comparable WERs?

Improved Training of Tandem Systems

- Jointly optimise tandem system with MPE or other discriminative sequence criteria
- Can be viewed as MDNN hybrid system MPE training

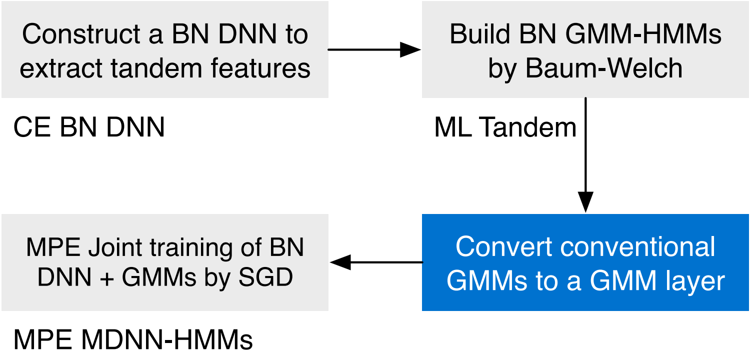
Proposed Methods

- Adapt extended Baum-Welch (EBW) based GMM MPE training to use stochastic gradient descent (SGD)
- Propose a set of methods to improve joint optimisation stability

Methodology

System Construction Procedure

- Convert GMMs to an MDNN GMM output layer for joint training



Construct a BN DNN to extract tandem features

CE BN DNN

Build BN GMM-HMMs by Baum-Welch

ML Tandem

Convert conventional GMMs to a GMM layer

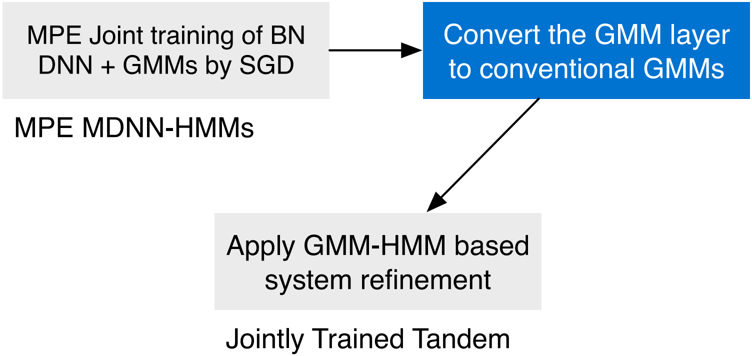
MPE Joint training of BN DNN + GMMs by SGD

MPE MDNN-HMMs

Methodology

System Refinement and Decoding

- GMM layer is converted back to GMMs to reuse existing facilities



MPE Joint training of BN DNN + GMMs by SGD

MPE MDNN-HMMs

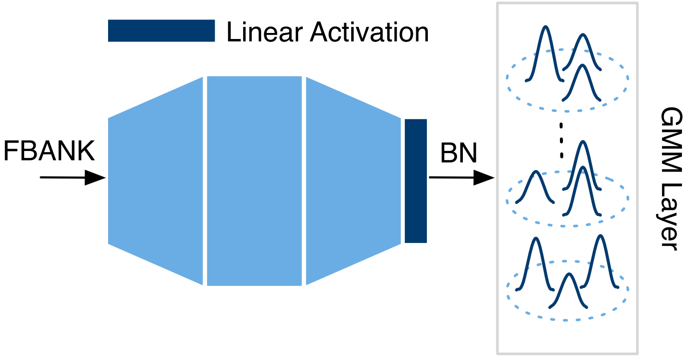
Convert the GMM layer to conventional GMMs

Apply GMM-HMM based system refinement

Jointly Trained Tandem

ML Tandem System Construction

- monophone BN GMM-HMMs → initial triphone BN GMM-HMMs → HMM state clustering → final triphone BN GMM-HMMs



SGD based GMM-HMM Training

GMM Parameter Update Values

- Calculate the partial derivatives of $\mathcal{F}$ w.r.t. each GMM parameter and input value
- For SGD, Gaussian component weight and std. dev. values are transformed so constraints satisfied

Speed Up

- Rearrange mean and std. dev. from of Gaussians as matrices
- Speed up GMM calculations by highly optimised general matrix multiplication (GEMM) functions in the BLAS library

MPE Training for GMM-HMMs using SGD

Regularisation

- Parameter smoothing
  - I-smoothing with $\mathcal{F}^{\text{ML}}$: data dependent coeff. $\tau^{\text{ML}}(s, g)$
  - H-criterion with $\mathcal{F}^{\text{MMI}}$: fixed coeff. τ^{MMI} (H-criterion)
- L2 regularisation: $\lambda \cdot \theta^2/2$
- Composite objective function

$$\mathcal{F}^{\text{MPE}} + \tau^{\text{MMI}}(\mathcal{F}^{\text{MMI}} + \tau^{\text{ML}}(s, g)\mathcal{F}^{\text{ML}}) + \lambda\,\theta^2/2$$

Percentile based Variance Floor

- Modified to find the flooring threshold more efficiently to apply frequently in SGD

Tandem System Joint Optimisation

Linear to ReLU Activation Function Conversion

- Observe instability issue when averaged partial derivatives w.r.t. linear BN features shifting from positive to negative
- To avoid negative values, modify BN layer bias to equivalently use ReLU by

$$b^{\text{bn}} - \mu^{\text{bn}} + 6\,\sigma^{\text{bn}}$$

Amplified GMM Learning

- GMMs have a rather different functional form than DNN layers
- Learning rates and L2 reg. coeff. are amplified for GMMs by α

Tandem System Joint Optimisation

Relative Update Value Clipping

- To avoid setting a specific threshold for each type of parameter
- Assuming values are Gaussian distributed, compute thresholds of Θ based on stats. in *n*th mini-batch by

$$\mu_{\Theta}[n] + m\,\sigma_{\Theta}[n]$$

Parameter Update Schemes

- Update GMMs and hidden layers in an interleaved manner
- Update all parameters concurrently without any restriction
- Update all parameters concurrently, then update the GMMs only

Experimental Setup

Data

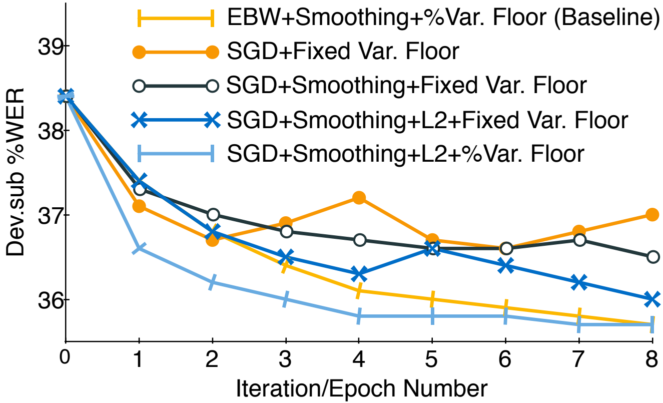
- 50h and 200h data from ASRU 2015 MGB challenge
- A trigram word level LM with a 160k word dictionary
- **dev.sub** test set contains 5.5h data with reference segmentation and 285 automatic speaker clusters

Systems

- All experiments were conducted with HTK 3.5
- 40-dim log-Mel filter bank features with their Δ coefficients
- DNN structure $720 \times 1000^5 \times \{4000, 6000\}$
  BN DNN structure $720 \times 1000^4 \times 39 \times 1000 \times \{4000, 6000\}$
- Each GMM has 16 Gaussians (`sil`/`sp` has 32 Gaussians)

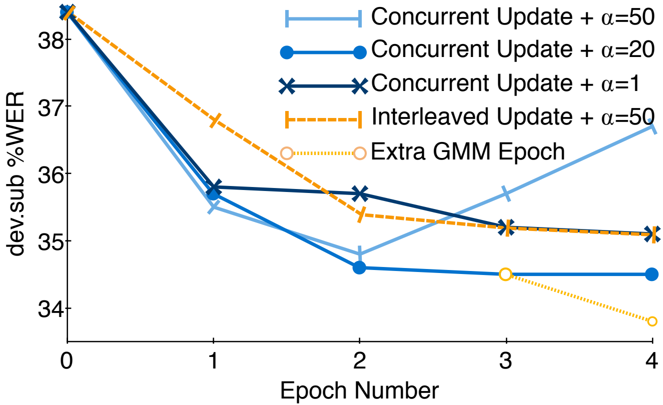
Experimental Results

Comparison of EBW and SGD GMM Training (50h)



Experimental Results

Joint Training Experiments with Different α (50h)



Experimental Results

Comparisons Among Various 50h Systems

- T_2^{50h} is comparable to hybrid MPE systems (H_1^{50h}&H_2^{50h}) in both WER and #parameters, and is useful for hybrid system (H_4^{50h})

ID	System	WER%
T_0^{50h}	ML BN-GMM-HMMs	38.4
T_1^{50h}	MPE BN-GMM-HMMs	36.1
T_2^{50h}	MPE MDNN-HMMs	33.8
H_0^{50h}	CE DNN-HMMs	36.9
H_1^{50h}	MPE DNN-HMMs	34.2
H_2^{50h}	MPE DNN-HMMs+H_1^{50h} align.	33.7
H_3^{50h}	MPE DNN-HMMs+T_2^{50h} align.	33.6
H_4^{50h}	MPE DNN-HMMs+T_2^{50h} align. & tree	33.2

Experimental Results

Comparisons Among Various 200h Systems

- MLLR and joint decoding still improve system performance

ID	System	WER%
T_0^{200h}	ML BN-GMM-HMMs	33.7
T_1^{200h}	MPE MDNN-HMMs	29.8
T_2^{200h}	MPE MDNN-HMMs+MLLR	28.6
H_0^{200h}	CE DNN-HMMs	31.9
H_1^{200h}	MPE DNN-HMMs	29.6
H_2^{200h}	MPE DNN-HMMs+T_1^{200h} align. & tree	29.0
J_1^{200h}	$T_1^{200h} \otimes H_2^{200h}$ joint decoding	28.3
J_2^{200h}	$T_2^{200h} \otimes H_2^{200h}$ joint decoding	27.4

Conclusions

Main Contributions Include

- EBW based GMM-HMM MPE training is extended to SGD
- MDNN discriminative sequence training is studied as tandem system joint optimisation
- A set of methods are modified/proposed to improve training that result in an 6.4% rel. WER reduction over MPE tandem systems

The Jointly Trained Tandem System

- is comparable to MPE hybrid systems in WER and #parameters
- is useful for hybrid system construction and system combination
- can also benefit from existing GMM approaches (e.g., MLLR)

Thanks for listening!