Deep Learning for Statistical Parametric Speech Synthesis

Zhen-Hua Ling
National Engineering Laboratory for Speech and Language Information Processing
University of Science and Technology of China, Hefei, China

Oct. 17, 2016

Tutorial @ ISCSLP 2016, Tian Jin

Outline

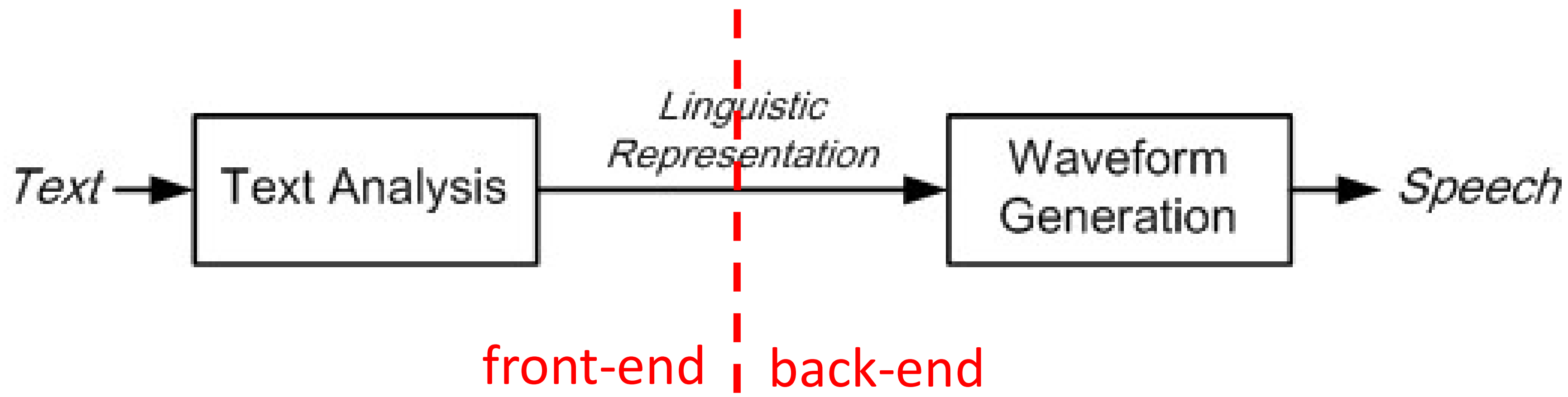
- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussion & Summary

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussion & Summary

Speech Synthesis

- Speech synthesis
  - Artificial production of human speech
- Text-to-speech (TTS)
  - To convert normal language text to speech



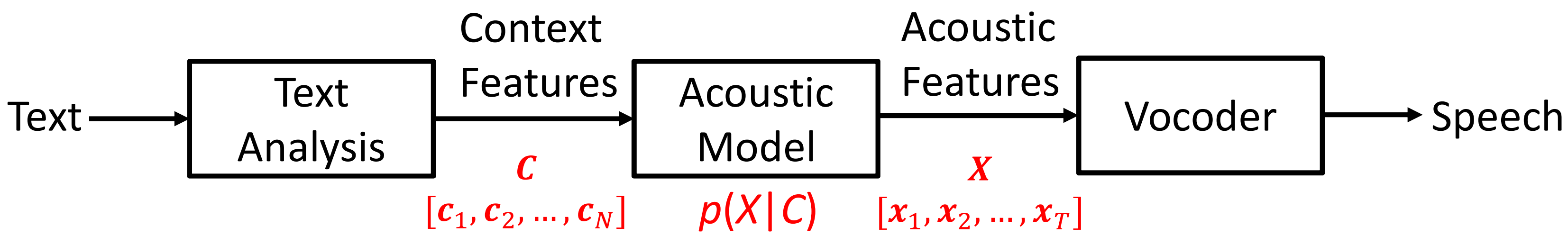
Speech Synthesis Methods (1/2)

- Rule-based, *formant synthesis* (~ '90s)
  - Hand-crafting each phonetic units by rules
  - Base on source-filter model
    - DECtalk [Klatt 1982]

- Corpus-based, *concatenative synthesis* ('90s~)
  - Concatenate speech units (waveform) from a database
  - Single inventory: diphone synthesis [Moulines 1990]
  - Multiple inventory: unit selection synthesis (USS) [Sagisaka 1992], [Hunt 1996]

Speech Synthesis Methods (2/2)

- Corpus-based, *statistical parametric synthesis*
  - Proposed in mid-’90s, becomes popular since mid-’00s



- Statistical
  - Statistical acoustic model based prediction from context features to acoustic features
- Parametric
  - speech vocoder based acoustic feature extraction and waveform reconstruction

Speech Synthesis Methods (2/2)

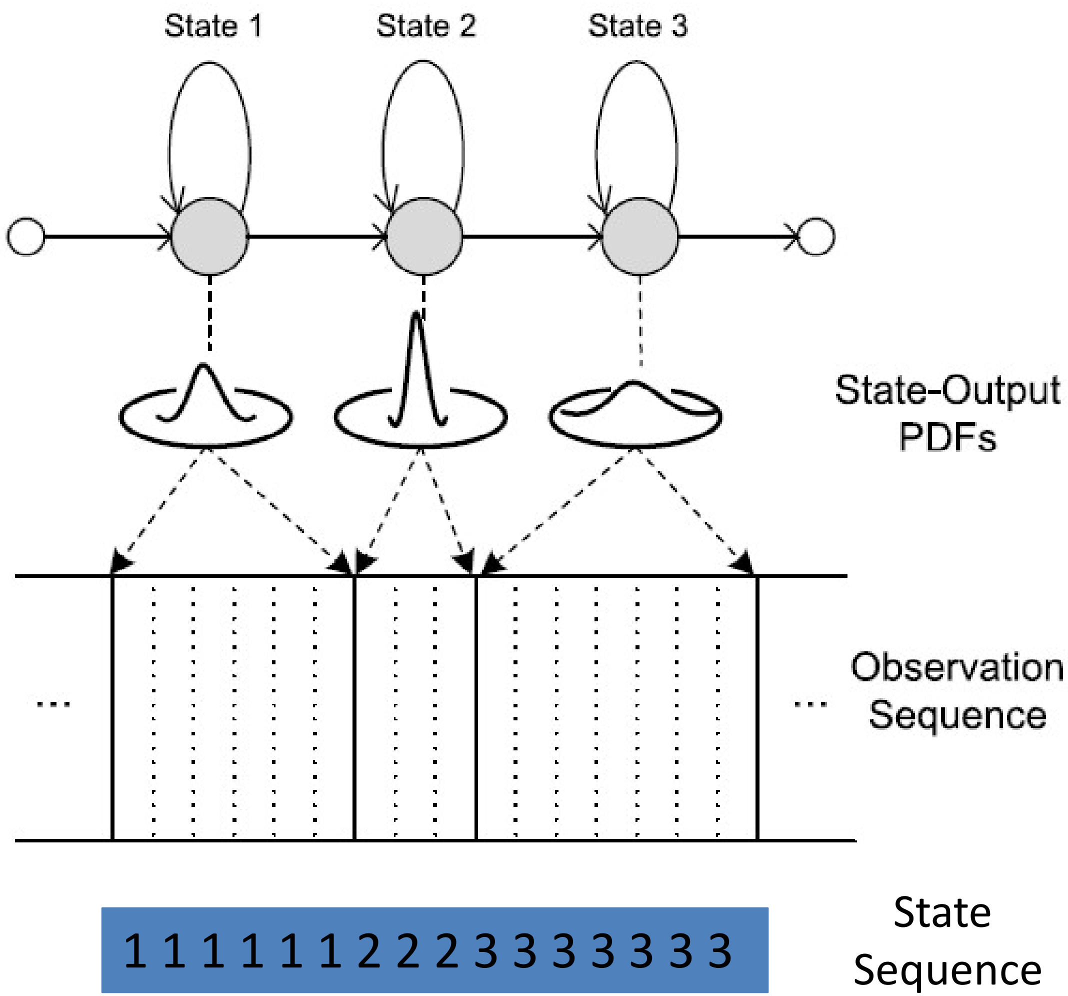
- Corpus-based, *statistical parametric synthesis*
  - Corpus + automatic training
    - ⇒ Automatic voice building
  - Source-filter model + statistical acoustic model
    - ⇒ Flexible to change its voice characteristics
  - HMM as its statistical acoustic model
    - ⇒ HMM-based Speech Synthesis System (HTS)
      [Yoshimura 1999]

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Summary

Hidden Markov model (HMM)

- Generate an observation sequence using a discrete and hidden state sequence

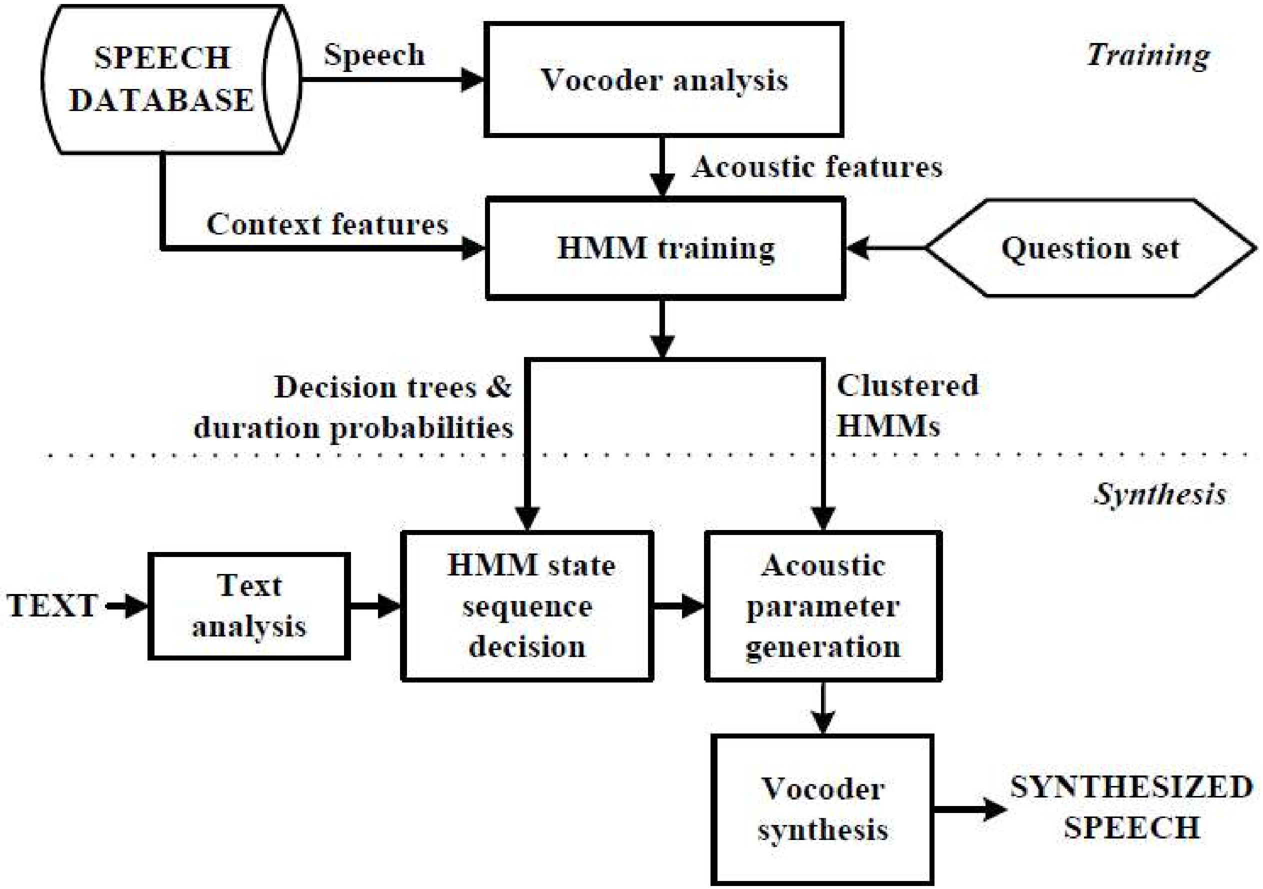


a_{ij} : State transition probability

$b_q(\boldsymbol{o}_t)$: Output probability

HMM-based Speech Synthesis (HTS)

- Framework



HMM-based Speech Synthesis (HTS)

- How to represent $p(X|C)$
  - Context-dependent phoneme HMMs [Yoshimura 1999]

联合国…

Text

Manual labeling
Text analysis

- ID of current/ surrounding phoneme
- Tones of current/surrounding syllables
- # of phonemes at current/ surrounding syllable
- Position of current syllable in current word
- …

Context features of each phoneme

```
XX-sil+l/A
sil-l+ian/A:XX_2@1/B:SH_H@H$H#A/C:8_8@1$1#1/D:3_3@1/V:0_1@1$0
l-ian+h/A:XX_2@1/B:SH_H@H$H#A/C:8_8@1$1#1/D:3_3@1/V:1_1@0$0
ian-h+e/A:2_1@2/B:WM_M@H$H#A/C:8_8@1$1#1/D:3_3@1/V:1_0@1$0
h-e+g/A:2_1@2/B:WM_M@H$H#A/C:8_8@1$1#1/D:3_3@1/V:0_1@0$0
e-g+uo/A:1_2@4/B:WT_T@H$H#A/C:8_8@1$1#1/D:3_3@1/V:1_0@1$0
g-uo+m/A:1_2@4/B:WT_T@H$H#A/C:8_8@1$1#1/D:3_3@1/V:0_1@1$0
...... ............ ......
...... ............ ......
```

Context-dependent phonemes

- Construct sentence HMM by concatenating phoneme HMMs

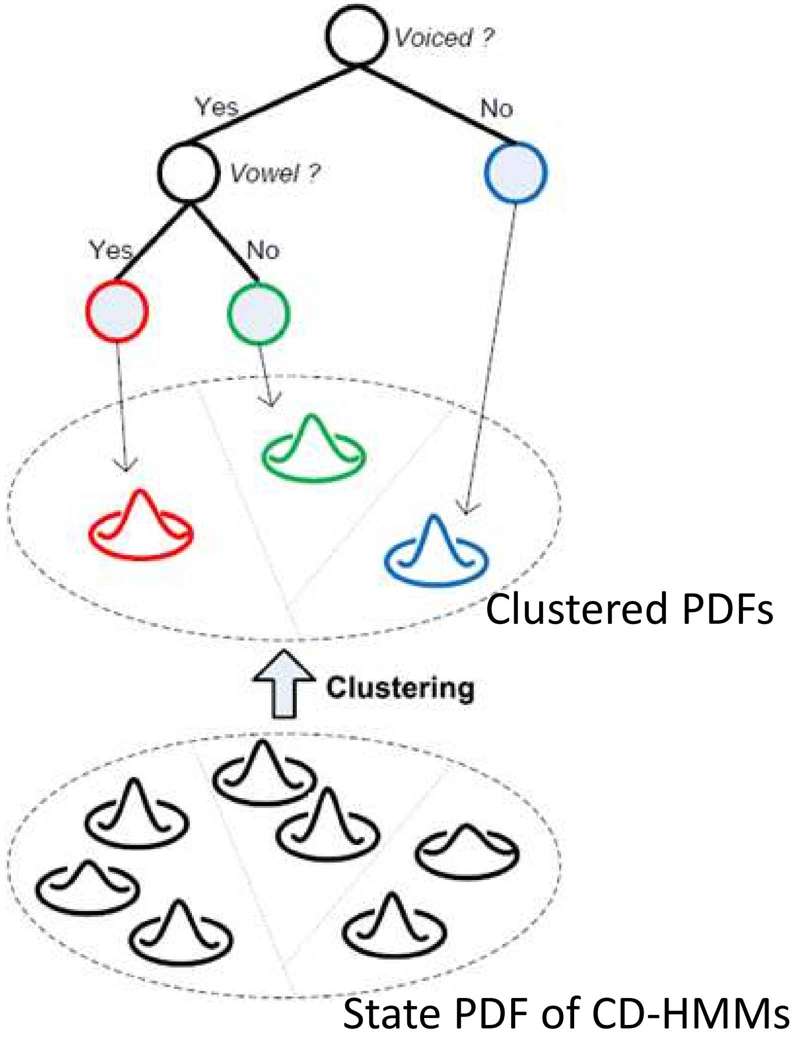
HMM-based Speech Synthesis (HTS)

- Model training
  - Maximum likelihood estimation using training database

$$p(\boldsymbol{X}|\boldsymbol{C}) = \sum_{\boldsymbol{q}} p(\boldsymbol{X}, \boldsymbol{q}|\boldsymbol{C}) = \sum_{\boldsymbol{q}} p(\boldsymbol{q}|\boldsymbol{C}) \prod_{t=1}^{T} \underbrace{p(\boldsymbol{x}_t|q_t)}_{\text{Gaussian Distribution}}$$

$$b_j(\boldsymbol{x}_t) = \mathcal{N}(\boldsymbol{x}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$$

  - Decision tree clustering [Shinoda 2000]
  - To train context-dependent state duration models



HMM-based Speech Synthesis (HTS)

- Parameter generation
  - To maximize $p(X|C)$ given the text analysis output C
  - Two steps

$$\boldsymbol{q}^* = \arg\max_{\boldsymbol{q}} p(\boldsymbol{q}|\boldsymbol{C})$$ ← State duration PDFs

$$\boldsymbol{X}^* = \arg\max_{\boldsymbol{X}} p(\boldsymbol{X}|\boldsymbol{q}^*, \boldsymbol{C})$$ ← Clustered HMM state PDFs

  - To generate smooth trajectories by introducing dynamic acoustic features and considering the constraints between static and dynamic features during parameter generation [Tokuda 2000]

Limitations

- Degraded quality of synthetic speech
- Three factors [Zen *et al.* 2009]
  - Limitations of the vocoder

  → e.g. STRAIGHT [Kawahara 1999]

  - Inadequacy of acoustic modeling

  → e.g. trajectory HMM [Zen 2007], MGE training [Wu 2006]

  - Over-smoothing effect of parameter generation

  → e.g. global variance [Toda 2007], minimum KLD [Ling 2012], modulation spectrum [Takamichi 2015]

How can deep learning techniques cope with these limitations?

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussion & Summary

What is Deep Learning ?

- **One of the various definitions**: A class of machine learning techniques that exploit many layers of non-linear information processing for supervised or unsupervised feature extraction and transformation, and for pattern analysis and classification.

Key Techniques of DL

- Modeling joint distribution, i.e., p(x) or p(x,y)
  - Restricted Boltzmann Machine (RBM)
  - Deep Belief Network (DBN)

- Modeling conditional distribution, i.e., p(y|x)
  - Deep Neural Network (DNN)
  - Recurrent Neural Network (RNN)

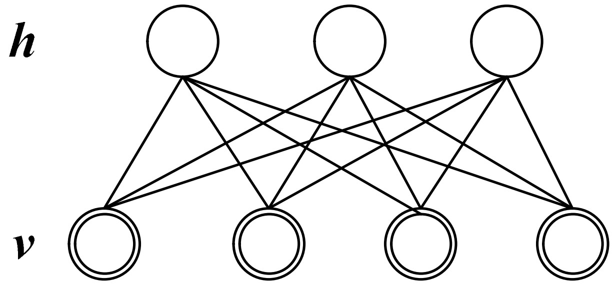
Key Techniques of DL

- Modeling joint distribution, i.e., p(x) or p(x,y)
  - Restricted Boltzmann Machine (RBM)
  - Deep Belief Network (DBN)

- Modeling conditional distribution, i.e., p(y|x)
  - Deep Neural Network (DNN)
  - Recurrent Neural Network (RNN)

Restricted Boltzmann Machines

- Model structure
  - two-layer undirected graphical model without within-layer connections [Smolensky 1986]
  - binary/real-valued visible units
    $$\boldsymbol{v} = [v_1, v_2, \dots, v_V]^{\mathsf{T}}$$
  - binary hidden units
    $$\boldsymbol{h} = [h_1, h_2, \dots, h_H]^{\mathsf{T}}$$
  - energy function of the state $\{\boldsymbol{v}, \boldsymbol{h}\}$



Bernoulli-Bernoulli RBM

$$E(\boldsymbol{v}, \boldsymbol{h}) = -\sum_{i=1}^{V} a_i v_i - \sum_{j=1}^{H} b_j h_j - \sum_{i=1}^{V}\sum_{j=1}^{H} w_{ij} v_i h_j$$

Gaussian-Bernoulli RBM

$$E(\boldsymbol{v}, \boldsymbol{h}) = \sum_{i=1}^{V} \frac{(v_i - a_i)^2}{2} - \sum_{j=1}^{H} b_j h_j - \sum_{i=1}^{V}\sum_{j=1}^{H} w_{ij} v_i h_j$$

Restricted Boltzmann Machines

- As a density model
  - joint distribution over the visible and hidden units

$$P(\boldsymbol{v},\boldsymbol{h}) = \frac{1}{Z}\exp\big(-E(\boldsymbol{v},\boldsymbol{h})\big)$$

  where partition function Z can be estimated using the annealed importance sampling (AIS) method [Salakhutdinov 2009]

  - marginal distribution over the visible units

$$P(\boldsymbol{v}) = \frac{1}{Z}\sum_{\boldsymbol{h}}\exp\big(-E(\boldsymbol{v},\boldsymbol{h})\big)$$

density model describing the distribution of vector $\boldsymbol{v}$

  - Estimate model parameters $\{\boldsymbol{W},\boldsymbol{a},\boldsymbol{b}\}$ by ML learning using the contrastive divergence (CD) algorithm [Hinton 2002]

Restricted Boltzmann Machines

- As a density model
  - Gaussian-Bernoulli RBM

$$P(\boldsymbol{v}) = \frac{1}{Z}\sum_{\boldsymbol{h}} \exp\big(-E(\boldsymbol{v},\boldsymbol{h})\big) = \frac{1}{Z}\sum_{\boldsymbol{h}} \exp\left(-\sum_{i=1}^{V}\frac{(v_i - a_i)^2}{2} + \boldsymbol{b}^{\mathsf{T}}\boldsymbol{h} + \boldsymbol{v}^{\mathsf{T}}\boldsymbol{W}\boldsymbol{h}\right)$$

Product of Experts model $\quad \frac{1}{Z}\prod_{i=1}^{V}\exp\left(-\frac{(v_i - a_i)^2}{2}\right)\prod_{j=1}^{H}\left(1 + \exp(b_j + \boldsymbol{v}^{\mathsf{T}}\boldsymbol{w}_j)\right)$

- elements in the first product represent single-variable experts
- elements in the second product represent constraints between the input variables

GMM $\quad \frac{1}{Z}\exp\left(-\sum_{i=1}^{V}\frac{(v_i - a_i)^2}{2}\right)\prod_{j=1}^{H}\left(1 + \exp(b_j + \boldsymbol{v}^{\mathsf{T}}\boldsymbol{w}_j)\right)$

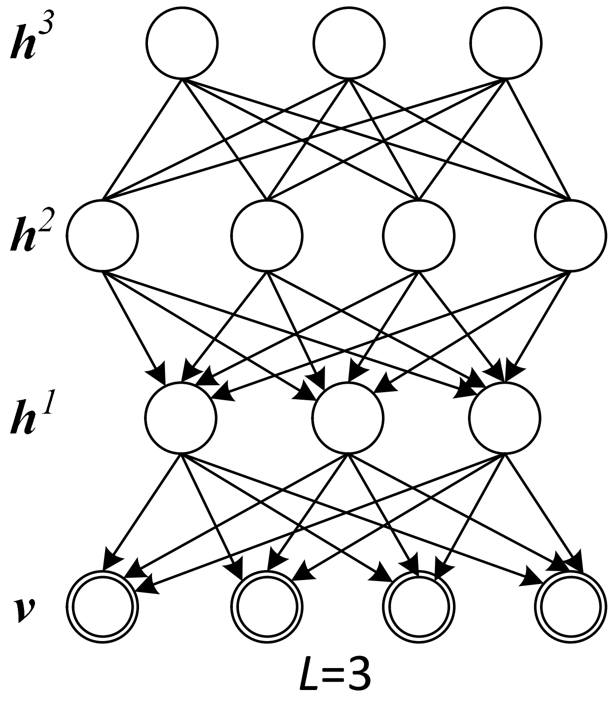
- 2^H mixtures
- structured mean vectors $\quad \boldsymbol{a}\ (H = \boldsymbol{0}) \rightarrow \{\boldsymbol{a}, \boldsymbol{a} + \boldsymbol{w_1}\}\ (H = \boldsymbol{1})$
- shared identity covariance matrices

Restricted Boltzmann Machines

- As a density model — better than GMM
  - Capable of modeling high dimensional features
    - Visible units are conditional independent on each other
    - Weights can capture cross dimensional correlations
  - RBM can model more patterns than GMM
    - A GMM with 2^H mixtures
  - RBM can model shaper distributions
    - Product of experts
  - Better generalization and less over-fitting
    - Binary hidden units create a information bottleneck and act as an effective regularizer

Deep Belief Networks

- Model structure
  - a graphical model with multi-layer hidden units [Hinton 2006]
  - real-valued visible units and binary hidden units
  - $P(\boldsymbol{h}^{L-1}, \boldsymbol{h}^{L})$ is represented by an RBM $\{\boldsymbol{W}^{L}, \boldsymbol{a}^{L}, \boldsymbol{b}^{L}\}$
  - $P(\boldsymbol{v}|\boldsymbol{h}^{1})$ and $\mathrm{P}\left(\boldsymbol{h}^{l-1}|\boldsymbol{h}^{l}\right)$, $l \in \{2,3,\dots,L-1\}$ are represented by sigmoid belief networks [Neal 1992]



$$P(\boldsymbol{v}|\boldsymbol{h}^{1}) = \mathcal{N}\left(\boldsymbol{v}; \boldsymbol{W}^{1\mathsf{T}}\boldsymbol{h}^{1} + \boldsymbol{a}^{1}, \mathbf{I}\right)$$

$$P\left(h_{\boldsymbol{i}}^{\boldsymbol{l}-1} = 1|\boldsymbol{h}^{l}\right) = g\left(a_i^l + \sum_j w_{ij}^l h_j^l\right) \qquad g(x) = 1/(1 + \exp(-x))$$

Deep Belief Networks

- Popularly used for pre-training of DNNs [Hinton 2006]
- As a density model
  - joint distribution over the visible and all hidden units

$$P(\boldsymbol{v}, \boldsymbol{h}^1, \dots, \boldsymbol{h}^L) = \underbrace{P(\boldsymbol{v}|\boldsymbol{h}^1)P(\boldsymbol{h}^1|\boldsymbol{h}^2)\cdots P(\boldsymbol{h}^{L-2}|\boldsymbol{h}^{L-1})}_{\text{SBN}}\underbrace{P(\boldsymbol{h}^{L-1}, \boldsymbol{h}^L)}_{\text{RBM}}$$

  - marginal distribution over the visible units

$$P(\boldsymbol{v}) = \sum_{\boldsymbol{h}^1} \cdots \sum_{\boldsymbol{h}^L} P(\boldsymbol{v}, \boldsymbol{h}^1, \dots, \boldsymbol{h}^L)$$

- Model training
  - difficult to estimate the model parameters directly under ML criterion
  - Greedy learning using a stack of RBMs

Key Techniques of DL

- Modeling joint distribution, i.e., p(x) or p(x,y)
  - Restricted Boltzmann Machine (RBM)
  - Deep Belief Network (DBN)

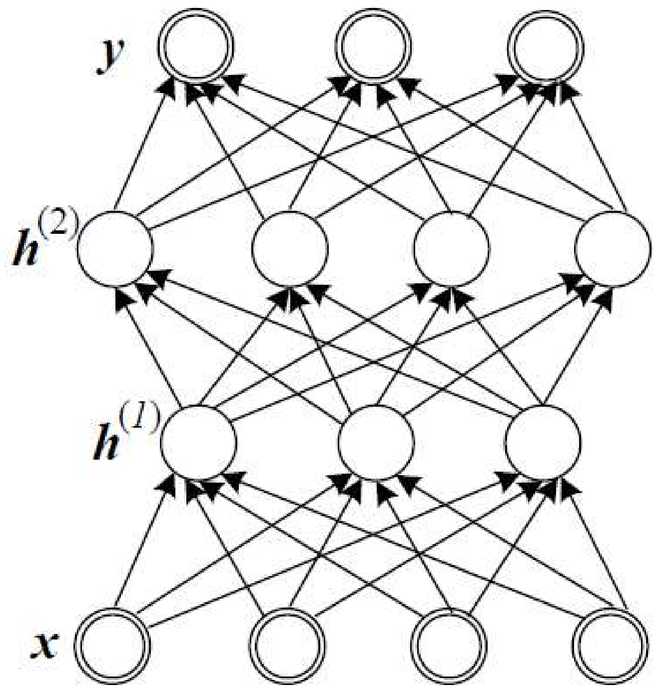
- Modeling conditional distribution, i.e., p(y|x)
  - Deep Neural Network (DNN)
  - Recurrent Neural Network (RNN)

Deep Neural Networks

- Model structure
  - a feed-forward, artificial neural network with than one layer of hidden units between input and output layers [Hinton 2006]
  - non-linear activation function at hidden units

$$h_j^{(l)} = g\left(b_j^{(l)} + \sum_i h_i^{(l-1)} w_{ij}^{(l)}\right)$$

    - $h_i^{(0)} = x_i$
    - Sigmoid / ReLU …



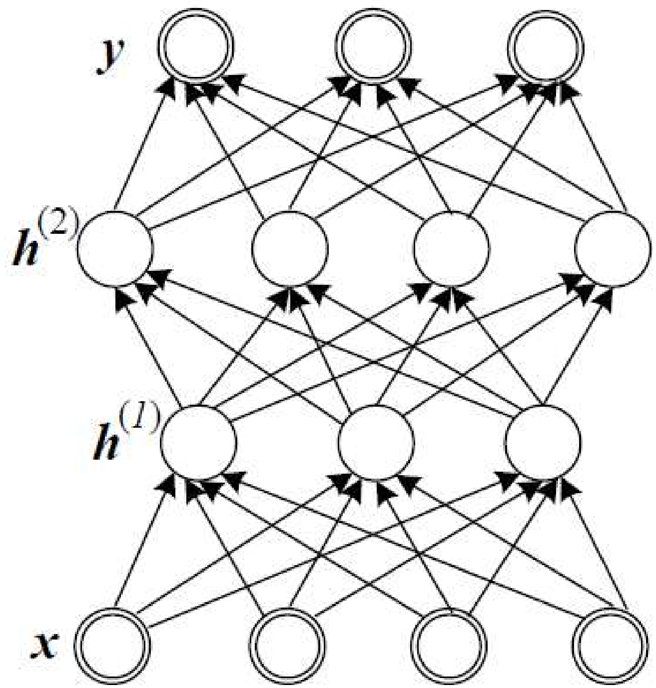
Deep Neural Networks

- Model structure
  - Output layer
    - Softmax function for classification

$$\tilde{y}_j = \frac{\exp\left\{b_j^{(L+1)} + \sum_i h_i^{(L)} w_{ij}^{(L+1)}\right\}}{\sum_k \exp\left\{b_k^{(L+1)} + \sum_i h_i^{(L)} w_{ik}^{(L+1)}\right\}}$$

    - Linear function for regression

$$\tilde{y}_j = b_j^{(L+1)} + \sum_i h_i^{(L)} w_{ij}^{(L+1)}$$



  - Parameter set

$$\lambda = \{\boldsymbol{b}^{(1)}, \boldsymbol{W}^{(1)}, \ldots, \boldsymbol{b}^{(L+1)}, \boldsymbol{W}^{(L+1)}\}$$

Deep Neural Networks

- Model training
  - Loss function
    - Cross entropy for classification

$$\mathcal{L}(\boldsymbol{y}, \tilde{\boldsymbol{y}}; \lambda) = -\sum_j y_j \log(\tilde{y}_j)$$

    - Mean square error for regression

$$\mathcal{L}(\boldsymbol{y}, \tilde{\boldsymbol{y}}; \lambda) = \sum_j (y_j - \tilde{y}_j)^2$$

  - Parameter estimation
    - Back-propagation [Rumelhart 1985]
    - Momentum / Weight decay
    - Pre-training using DBNs (stack RBMs), DAEs (deep auto-encoders)

Deep Neural Networks

- Consider a DNN for regression as a probabilistic model
  - a conditional PDF of y given x

Gaussian distribution

$$p(\boldsymbol{y}|\boldsymbol{x},\lambda) = \mathcal{N}(\boldsymbol{y};\tilde{\boldsymbol{y}}(\boldsymbol{x},\lambda),\boldsymbol{I})$$

Observed output

Observed input

Nonlinear transform from input using λ

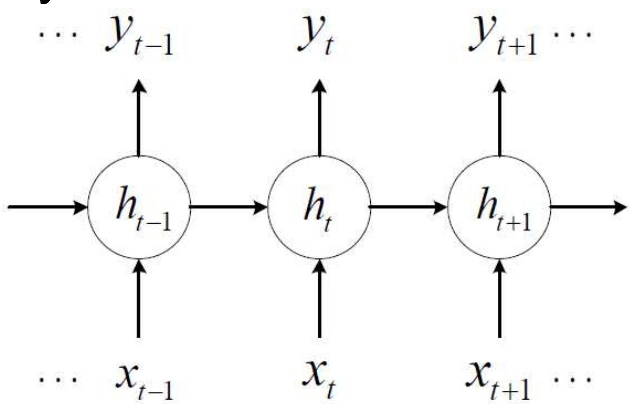
  - minimizing the mean square error between $\tilde{\boldsymbol{y}}$ and $\boldsymbol{y}$ with respect to λ is equivalent to the ML estimation of λ

Recurrent Neural Networks

- Model structure
  - a dynamic neural network where there are cyclical connections among hidden nodes [Hopfield 1982]
  - provide better ability of processing dynamic and temporal information
  - e.g. a regression RNN with one hidden layer

$$\boldsymbol{h}_t = \mathcal{H}\left(\boldsymbol{W}_{xh}\boldsymbol{x}_t + \boldsymbol{W}_{hh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_h\right)$$

$$\boldsymbol{y}_t = \boldsymbol{W}_{hy}\boldsymbol{h}_t + \boldsymbol{b}_y$$



  - stacking multiple recurrent hidden layers to build a deep RNN
  - unidirectional vs. bidirectional

Recurrent Neural Networks

- Consider a RNN as a conditional PDF
  - Unidirectional

$$p(\boldsymbol{y}_t|\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_t, \lambda)$$

  - Bidirectional

$$p(\boldsymbol{y}_t|\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_t, \dots, \boldsymbol{x}_T, \lambda)$$

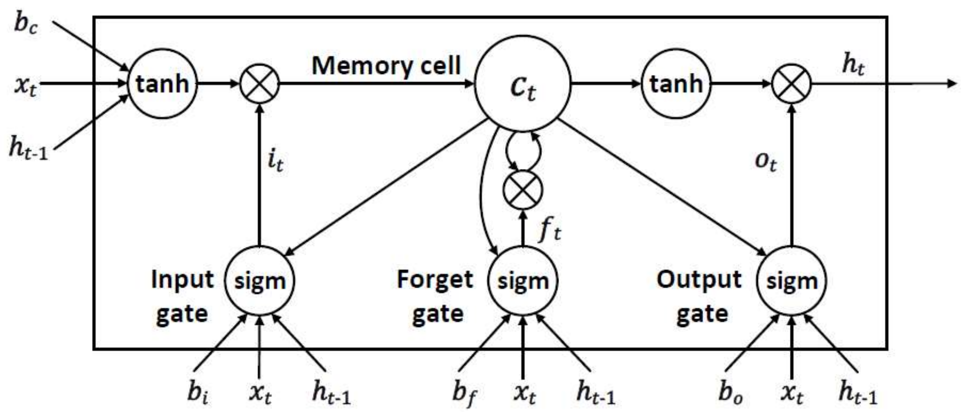
- Model training
  - Back-propagation through time (BPTT) [Werbos 1990]
  - Training difficulty: exploding and vanishing gradients

$\longrightarrow$ Long Short-Term Memory (LSTM) cell

Long-Short Term Memory (LSTM)

- An LSTM cell [Hochreiter 1997]
  - a complex hidden unit with gating structure
  - the information flow transmitting iteratively through the network is controlled by the input gate , forget gate, output gate and the cell memory state

$$
\begin{aligned}
\boldsymbol{i}_t &= \sigma(\boldsymbol{W}_{xi}\boldsymbol{x}_t + \boldsymbol{W}_{hi}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{ci}\boldsymbol{c}_{t-1} + \boldsymbol{b}_i) \\
\boldsymbol{f}_t &= \sigma(\boldsymbol{W}_{xf}\boldsymbol{x}_t + \boldsymbol{W}_{hf}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{cf}\boldsymbol{c}_{t-1} + \boldsymbol{b}_f) \\
\boldsymbol{c}_t &= \boldsymbol{f}_t * \boldsymbol{c}_{t-1} + \boldsymbol{i}_t * \tanh(\boldsymbol{W}_{xc}\boldsymbol{x}_t + \boldsymbol{W}_{hc}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c) \\
\boldsymbol{o}_t &= \sigma(\boldsymbol{W}_{xo}\boldsymbol{x}_t + \boldsymbol{W}_{ho}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{co}\boldsymbol{c}_t + \boldsymbol{b}_o) \\
\boldsymbol{h}_t &= \boldsymbol{o}_t * \tanh(\boldsymbol{c}_t)
\end{aligned}
$$



  - capable of remembering information from a long span of time steps
  - success in speech recognition [Graves 2013a], handwriting generation [Graves 2013b], etc.

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for SPSS
- Discussion & Summary

Limitations of HMM-Based AMs

- Input-to-Cluster mapping using decision trees
  - Inefficient for expressing complex context dependencies, e.g. XOR

    Overfitting to the training data due to the data partitioning issue

- Cluster-to-feature mapping using Gaussians
  - Difficulty in estimating full covariance matrices

    Using low-dimensional spectral parameters (mel-cepstra / LSPs)

    Detailed characteristics of the raw spectra are lost
  - Averaged model means by ML training

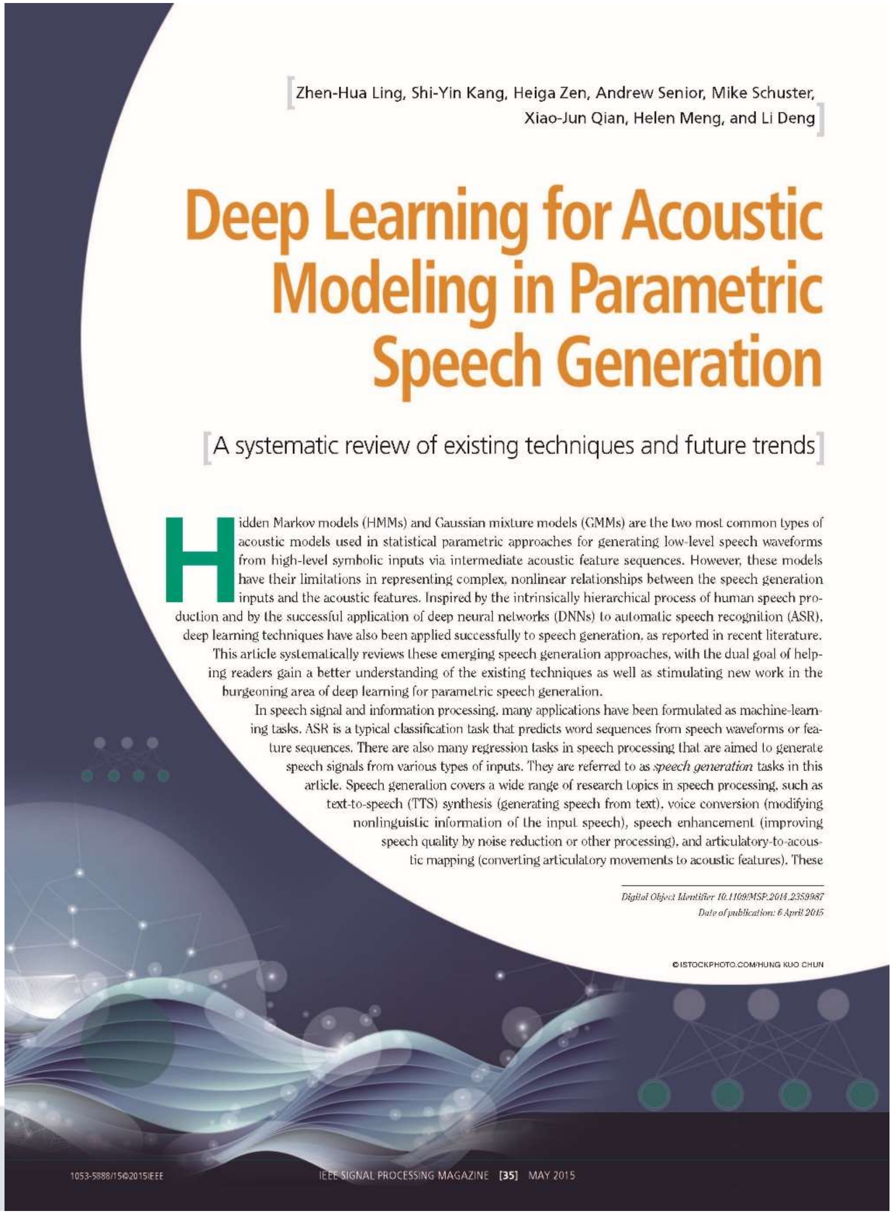
    Outputs of MLPG distribute near the modes (means) of Gaussians

    The generated spectral features are over-smoothed

Need better models for acoustic modeling of SPSS !

DL-Based Acoustic Modeling for SPSS

- Since 2013
- Three different strategies
  - Cluster-to-feature mapping using RBMs (USTC & Microsoft)
  - Input-to-feature mapping using DBNs (CUHK)
  - Input-to-feature mapping using deep-structured NNs (Google)
- A survey paper @ IEEE SPM



Zhen-Hua Ling, Shi-Yin Kang, Heiga Zen, Andrew Senior, Mike Schuster, Xiao-Jun Qian, Helen Meng, and Li Deng

Deep Learning for Acoustic Modeling in Parametric Speech Generation

A systematic review of existing techniques and future trends

Hidden Markov models (HMMs) and Gaussian mixture models (GMMs) are the two most common types of acoustic models used in statistical parametric approaches for generating low-level speech waveforms from high-level symbolic inputs via intermediate acoustic feature sequences. However, these models have their limitations in representing complex, nonlinear relationships between the speech generation inputs and the acoustic features. Inspired by the intrinsically hierarchical process of human speech production and by the successful application of deep neural networks (DNNs) to automatic speech recognition (ASR), deep learning techniques have also been applied successfully to speech generation, as reported in recent literature. This article systematically reviews these emerging speech generation approaches, with the dual goal of helping readers gain a better understanding of the existing techniques as well as stimulating new work in the burgeoning area of deep learning for parametric speech generation.

In speech signal and information processing, many applications have been formulated as machine-learning tasks. ASR is a typical classification task that predicts word sequences from speech waveforms or feature sequences. There are also many regression tasks in speech processing that are aimed to generate speech signals from various types of inputs. They are referred to as *speech generation* tasks in this article. Speech generation covers a wide range of research topics in speech processing, such as text-to-speech (TTS) synthesis (generating speech from text), voice conversion (modifying nonlinguistic information of the input speech), speech enhancement (improving speech quality by noise reduction or other processing), and articulatory-to-acoustic mapping (converting articulatory movements to acoustic features). These

Digital Object Identifier 10.1109/MSP.2014.2359987
Date of publication: 6 April 2015

Cluster-to-feature mapping using RBMs

Framework

- Motivation
  - The advantages of RBMs in describing the distribution of high-dimensional observations with cross-dimension correlations
- Method [Ling 2013]
  - Features

    High level spectral parameters

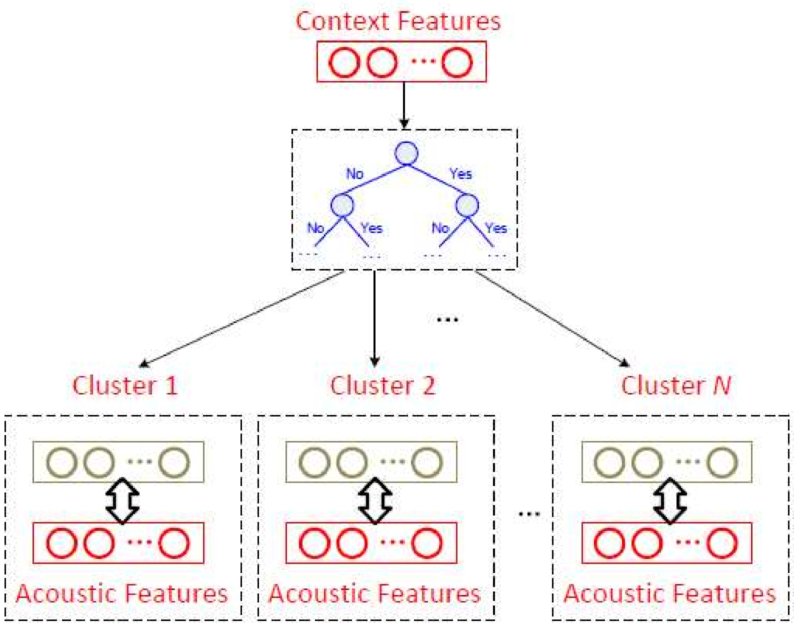
    ⇩

    Low level spectral envelopes
  - State PDFs

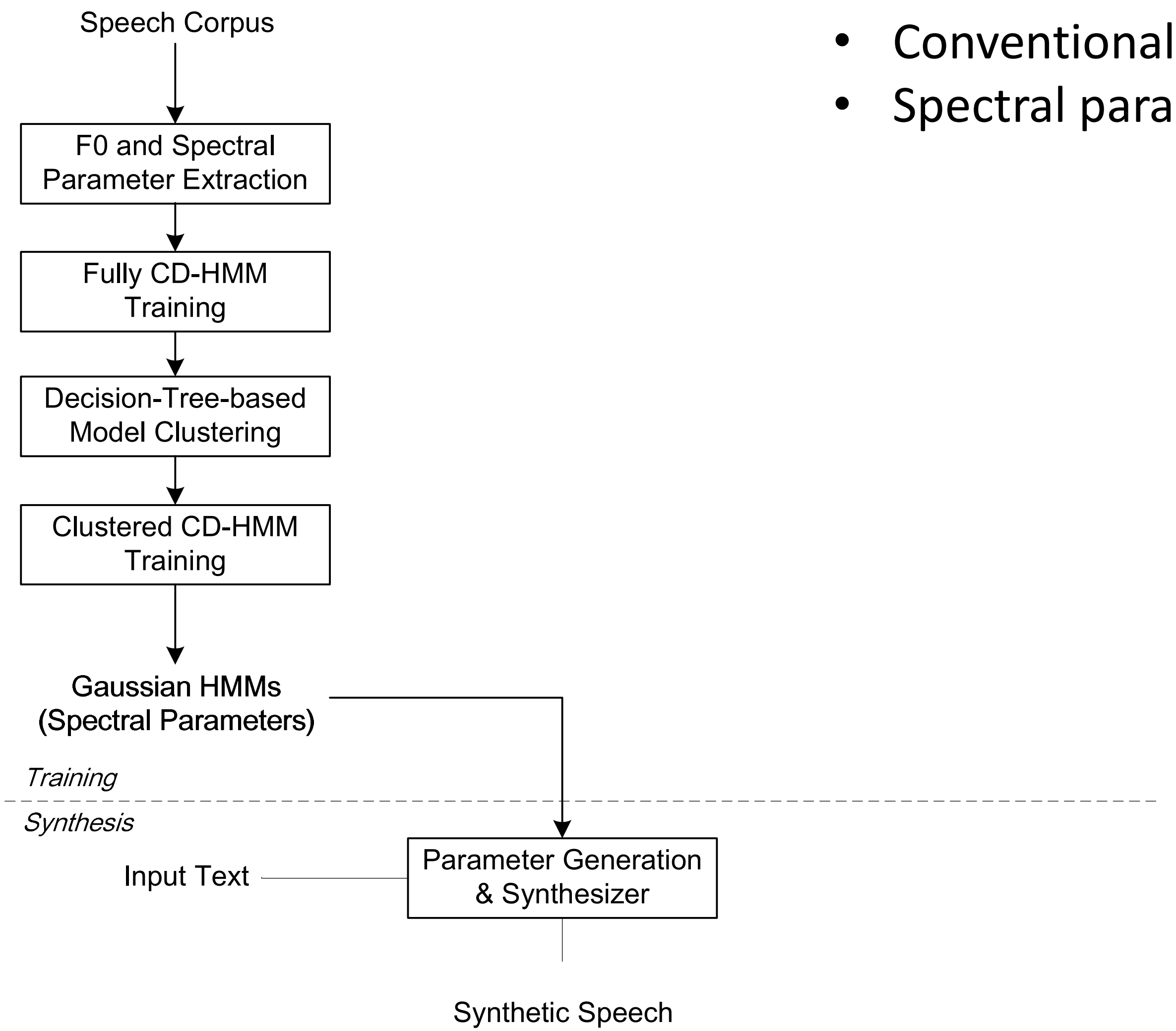
    Gaussian distributions

    ⇩

    RBMs

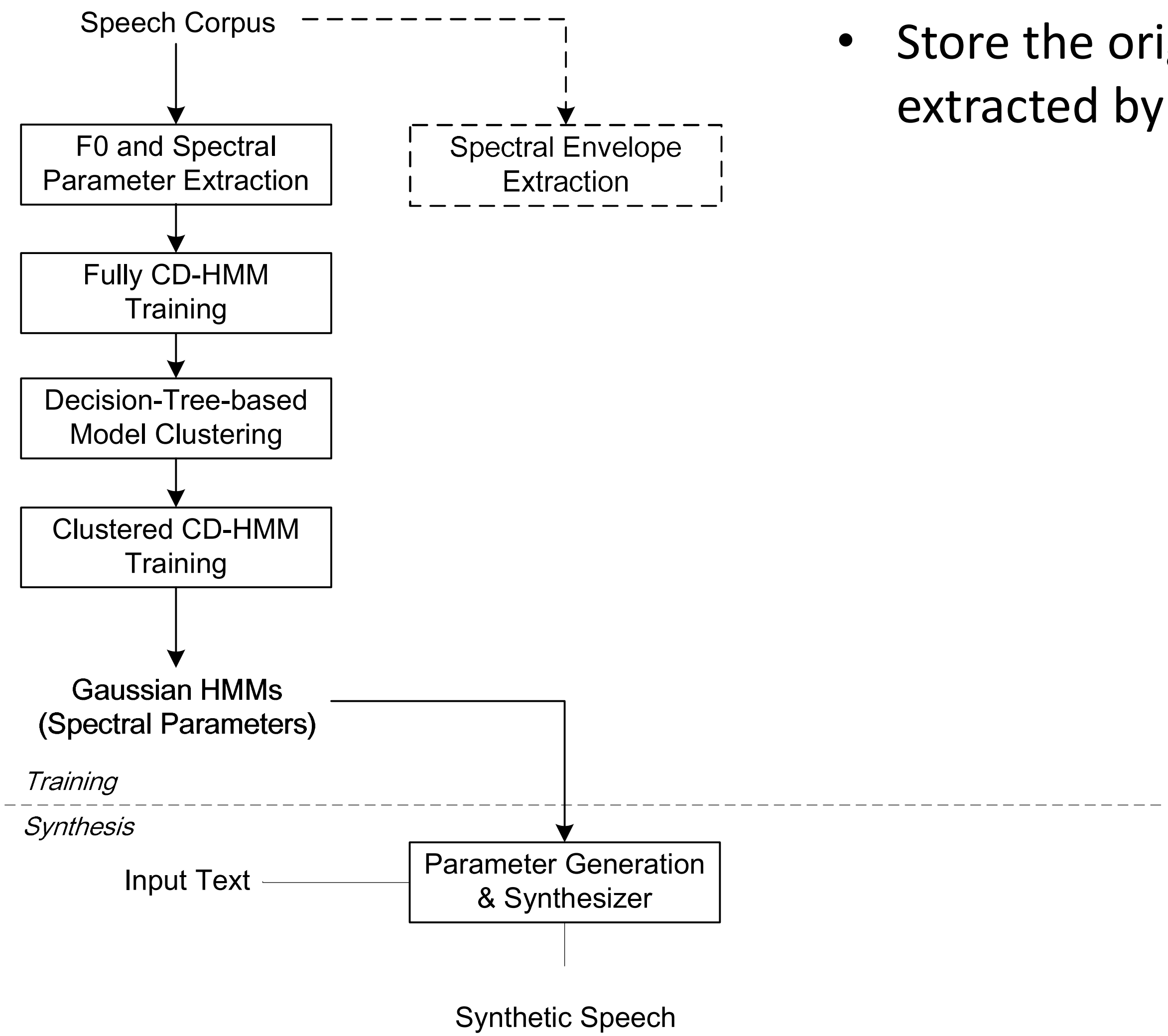


Implementation



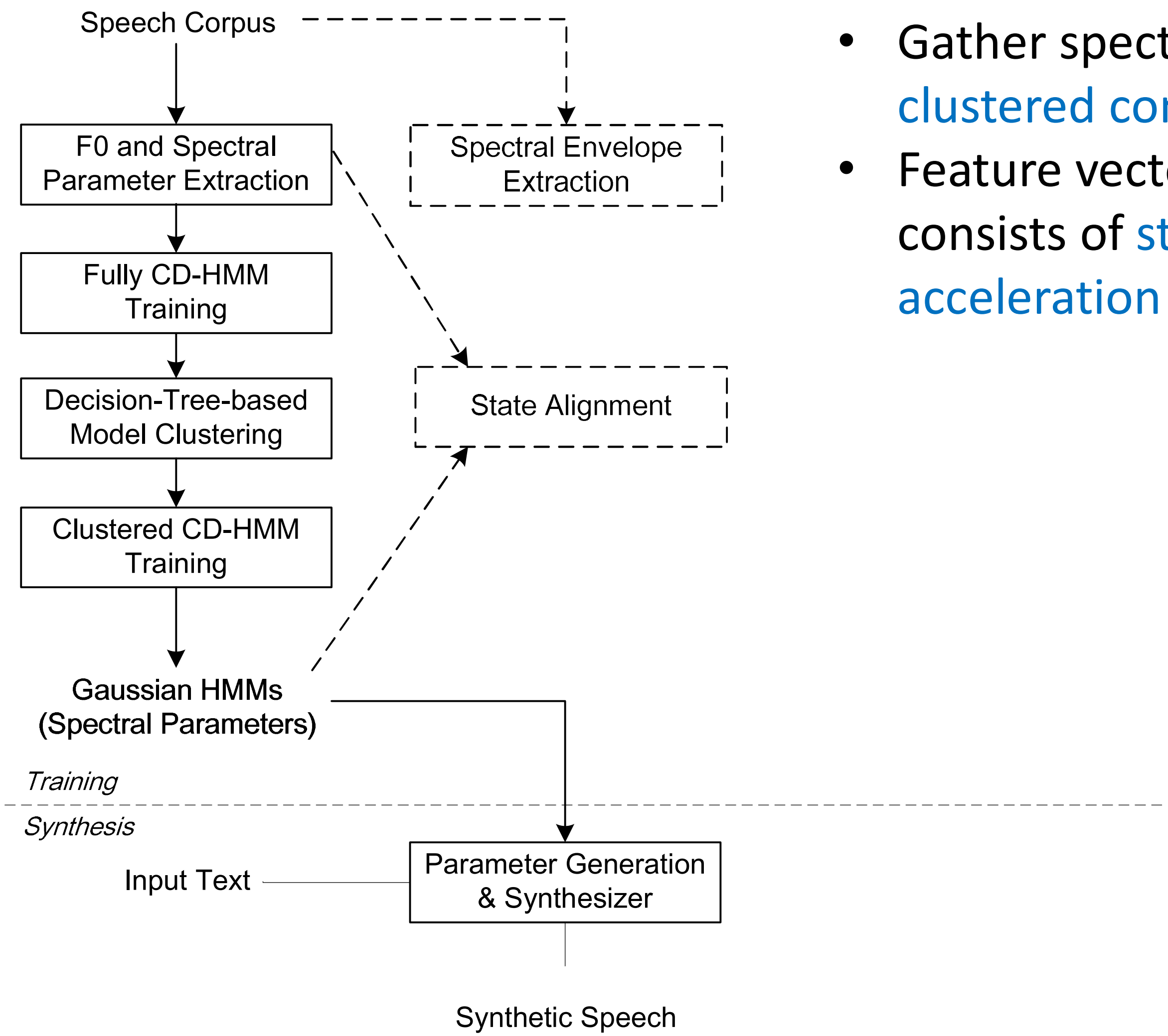
- Conventional HTS model training
- Spectral parameters (mel-cepstra/LSPs)

Implementation



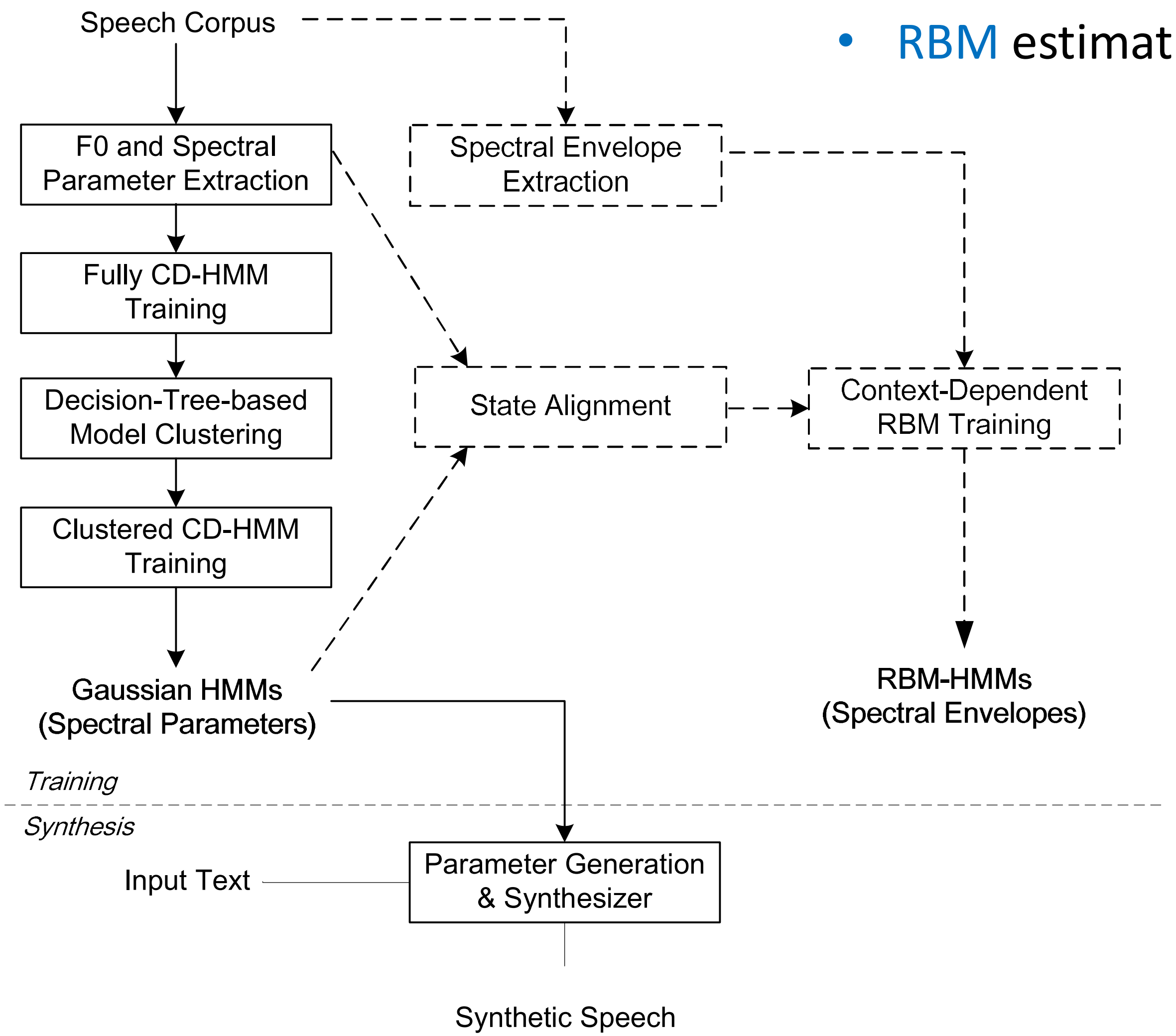
- Store the original spectral envelopes extracted by STRAIGHT

Implementation



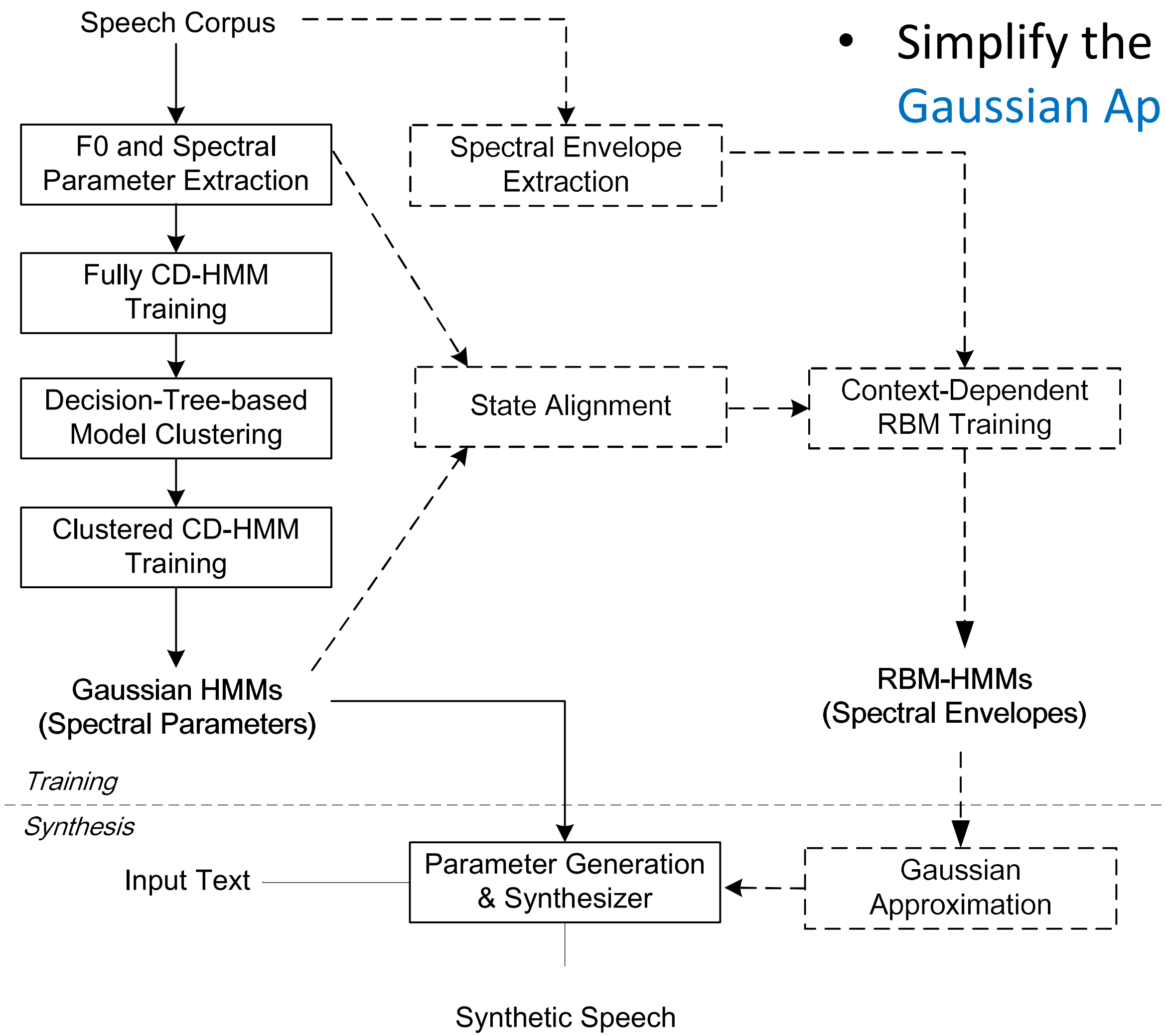
- Gather spectral envelopes for each clustered context-dependent state
- Feature vector of spectral envelopes consists of static / velocity / acceleration components

Implementation



- RBM estimation for each state

Implementation



- Simplify the generation problem by Gaussian Approximation

Gaussian Approximation

Gaussian distribution

$\mathcal{N}(\boldsymbol{v}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \longrightarrow P(\boldsymbol{v})$

RBM at each HMM state

$\boldsymbol{\mu} = \arg\max_{\boldsymbol{v}} \log P(\boldsymbol{v})$

sample covariances & diagonal

Mode Estimation [Ling 2013]

- RBM no close-form solution → gradient descent updating

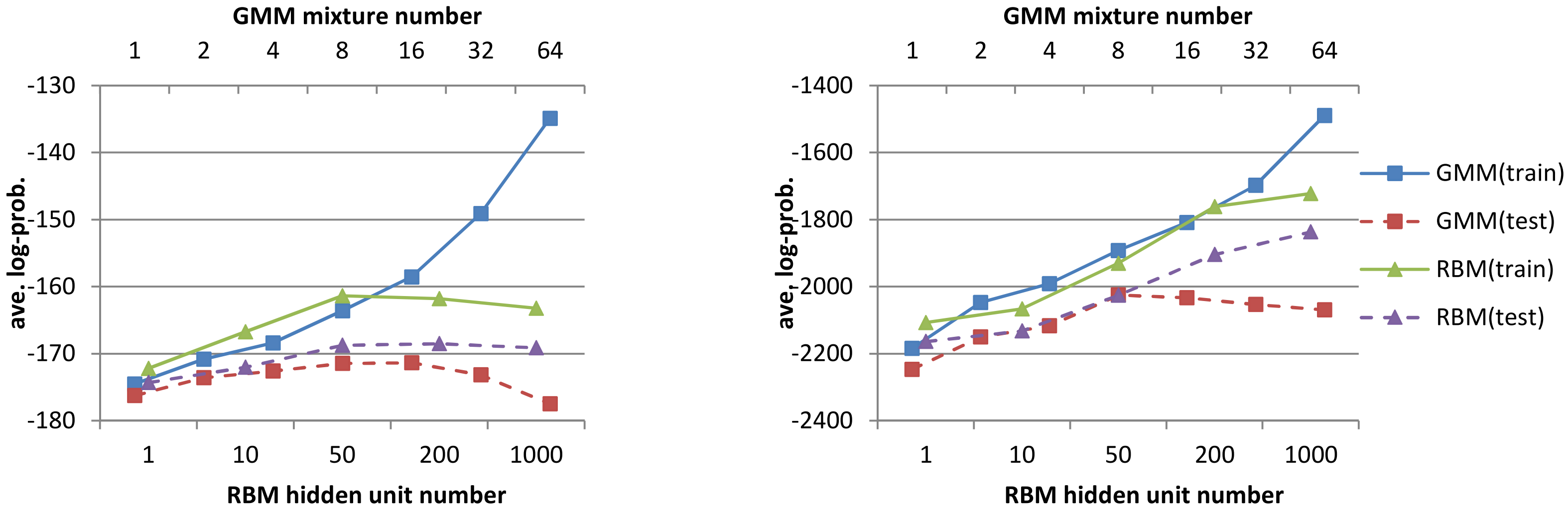
$$\frac{\partial \log P(\mathbf{v})}{\partial \mathbf{v}} = -(\mathbf{v} - \mathbf{a}) + \sum_{j=1}^{H} \frac{\exp(b_j + \mathbf{v}^\top \mathbf{w}_j)}{1 + \exp(b_j + \mathbf{v}^\top \mathbf{w}_j)} \mathbf{w}_j$$

Experiments

- Experimental Conditions
  - 1-hour Chinese speech database; female speaker; 16kHz/16bits
  - 800 utterances for training / 200 utterances for test
  - Low-level spectral features: STRAIGHT spectral envelopes (513)
  - High-level spectral features: mel-cepstra (41)
  - Context-dependent HMM training using mel-cepstra
    - MDL-based DT clustering: 1,612 states for spectral stream
  - RBM training
    - CD with 1-step Gibbs Sampling
    - learning rate = 0.0001; batch size = 10; epoch = 200

Experiments

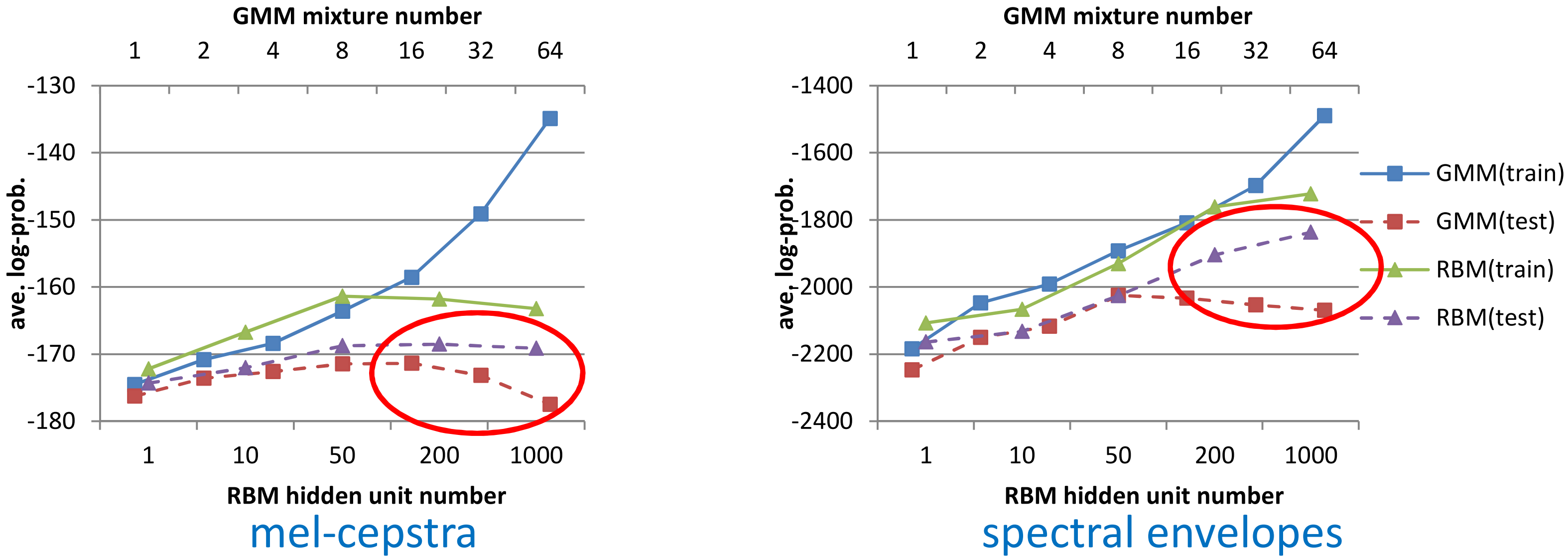
- Comparison between GMMs and RBMs as state PDFs



  - average log-prob. on the training and test sets when modeling the mel-cepstra (left) and the spectral envelopes (right)
  - a state with 650 training frames and 130 test frames
  - GMM mixture number: 1~64
  - RBM hidden unit number: 1~1,000

Experiments

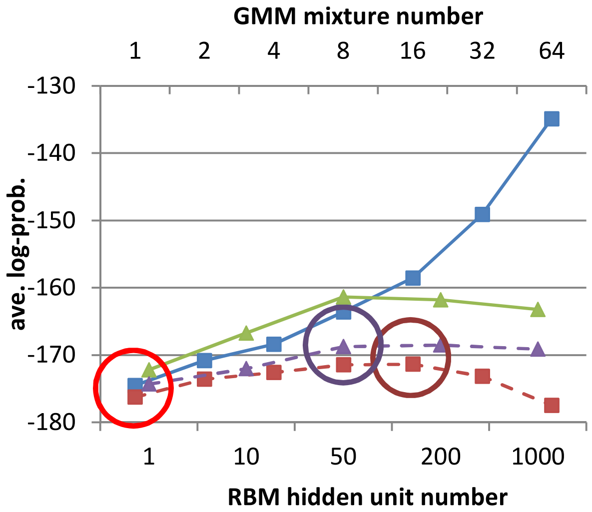
- Comparison between GMMs and RBMs as state PDFs



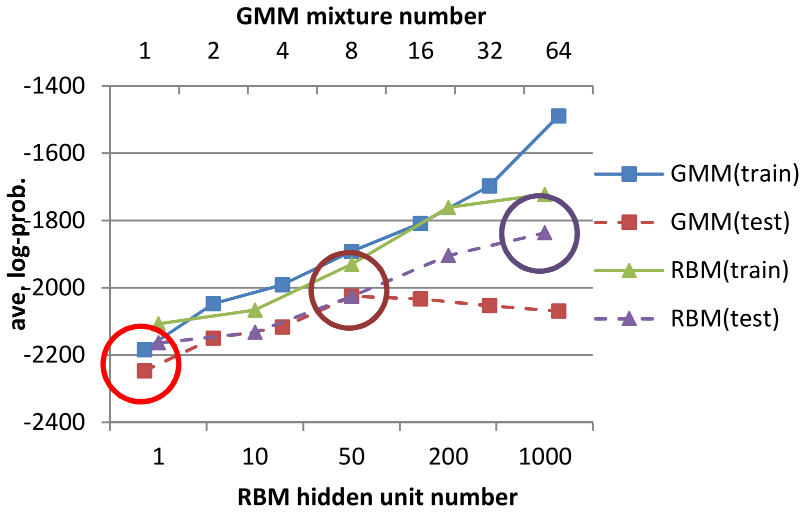
  - GMMs have a clear tendency of over-fitting with the increasing of model complexity
  - RBM shows consistently good generalization ability with the increasing of the number of hidden units

Experiments

- Comparison between GMMs and RBMs as state PDFs



mel-cepstra



spectral envelopes

- Mel-cepstra
  - the gain of using the density models more complex than a single Gaussian distribution are relatively small ← decorrelation processing of cepstral analysis
- Spectral envelopes
  - the gain becomes much more significant for both GMMs and RBMs
  - RBMs can give much higher test log-prob. than GMMs

Experiments

- System construction

System	Spectral Features	State PDF
Baseline	mel-cepstra	single Gaussian
GMM(1)	spectral envelopes	single Gaussian
GMM(8)	spectral envelopes	GMM, 8 mixtures
RBM(10)	spectral envelopes	RBM, 10 hidden units
RBM(50)	spectral envelopes	RBM, 50 hidden units

Experiments

- Subjective preference scores

Baseline	*GMM(8)*	*RBM(10)*	*RBM(50)*	*N/P*	*p*
18.67	**48.00**	-	-	33.33	0.0014
12.00	-	**50.67**	-	37.33	0.00
5.33	-	-	**70.67**	24.00	0.00
-	16.00	-	**69.33**	14.67	0.00
-	-	9.33	**37.33**	53.33	0.00

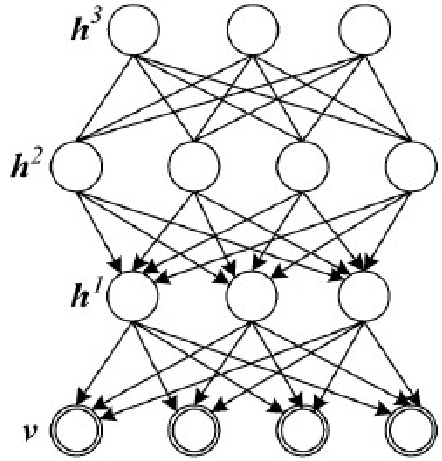
  - *Baseline* and *GMM(1)* have very similar synthetic results
  - GMMs and RBMs are significantly better than single Gaussian when modeling spectral envelopes
  - superiority of RBM over GMM in modeling the spectral envelopes
  - performance of the RBM-based systems is influenced by the number of hidden units used in the model

Demos

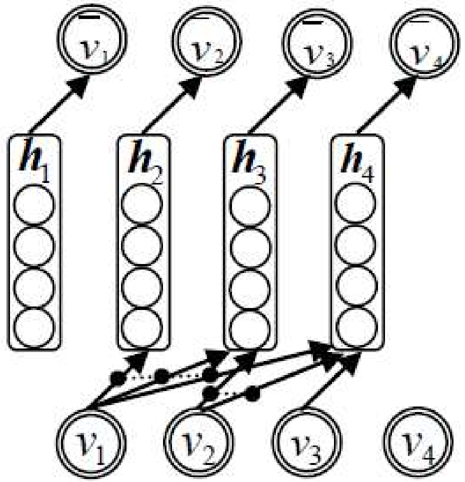
System	Spectral Features	State PDF	Demo
Baseline	mel-cepstra	single Gaussian	
GMM(1)	spectral envelopes	single Gaussian	
GMM(8)	spectral envelopes	GMM, 8 mixtures	
RBM(10)	spectral envelopes	RBM, 10 hidden units	
RBM(50)	spectral envelopes	RBM, 50 hidden units	

Extensions

- Other generative models
  - Deep Belief Network (DBN)
    [Ling 2013]

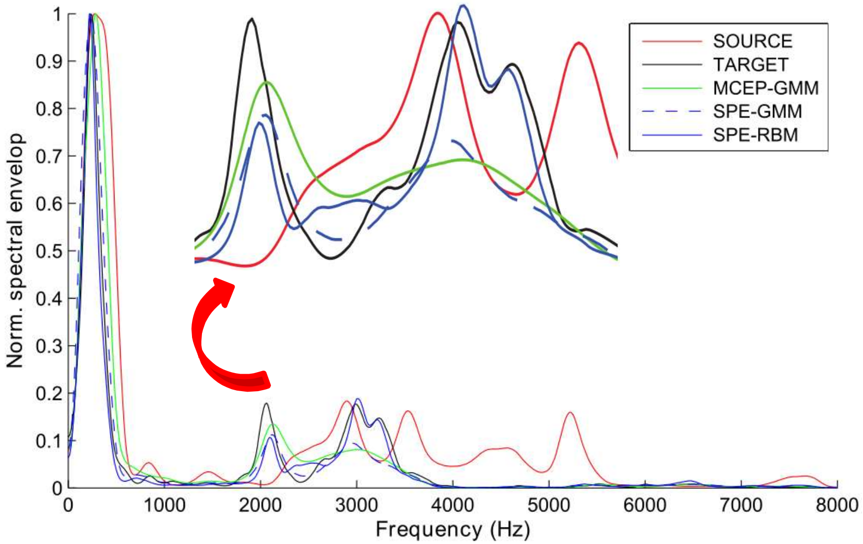


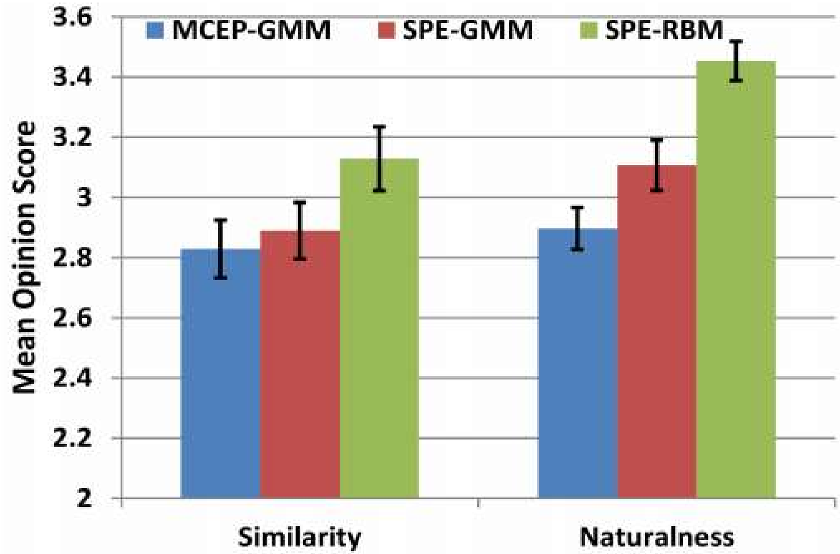
  - Neural Autoregressive Distribution Estimator (NADE) [Yin 2014]



Extensions

- Other applications
  - Voice conversion [Chen 2013]





Input-to-feature mapping using DBNs

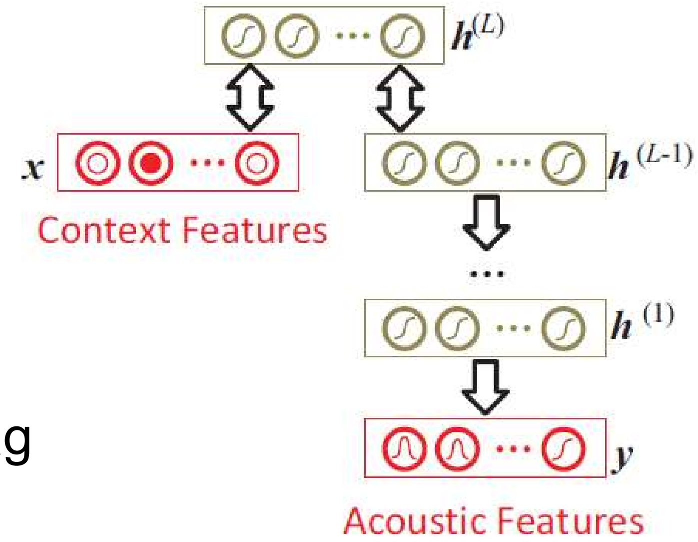
Framework

- Motivation
  - To model all data in a centralized network and avoid data partitioning
  - To model spectral coefficients without independence assumptions

- Method
  - Model the joint distribution p(x,y) using a single DBN
    - x input context features
    - y output acoustic features

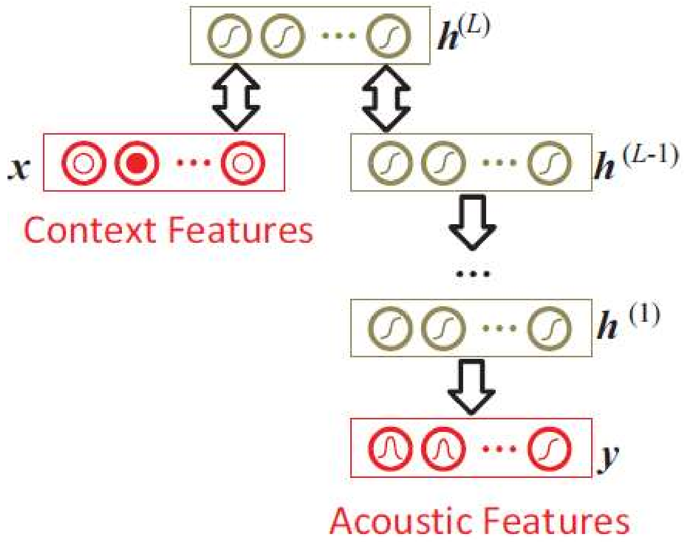
Implementation

- Mandarin Chinese speech synthesis with MD-DBN [Kang 2013]
  - Input context features
    - 1-of-k code of tonal syllables
  - Output acoustic features
    - Syllable-level spectrum and excitation features
    - MGCs / log energy / log F0 / UV flag
  - Multi-distribution DBN
    - Different types of distribution units in the visible layer (Gaussian/Bernoulli)



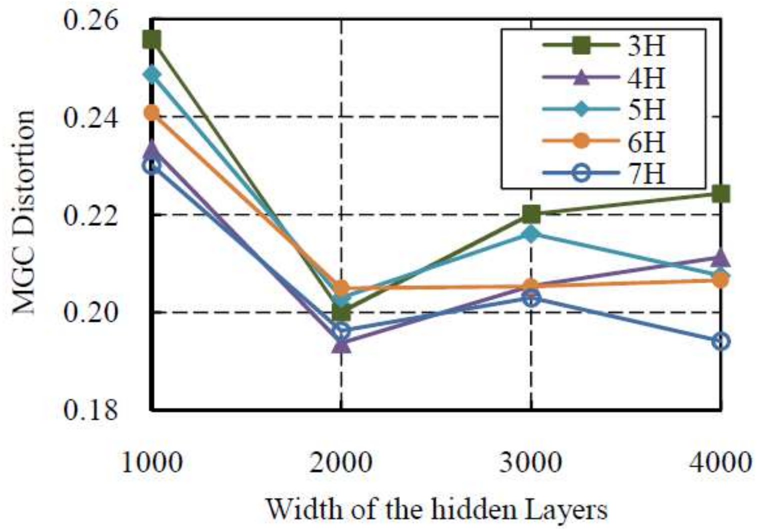
Implementation

- Mandarin Chinese speech synthesis with MD-DBN [Kang 2013]
  - Model training
    - Stacking up RBMs
    - Extend the (L-1)-th layer with context features
  - Synthesis
    - $\boldsymbol{x} \rightarrow \boldsymbol{h}^{(L-1)}$
      - Gibbs sampling between $[\boldsymbol{x}, \boldsymbol{h}^{(L-1)}]$ and $\boldsymbol{h}^{(L)}$ with $\boldsymbol{x}$ clamped
    - $\boldsymbol{h}^{(L-1)} \longrightarrow \cdots \longrightarrow \boldsymbol{h}^{(1)} \longrightarrow \boldsymbol{y}$
      - Using the mean value of $\Pr\left(\boldsymbol{h}^{(l-1)} \middle| \boldsymbol{h}^{(l)}\right)$ and $p\left(\boldsymbol{y} \middle| \boldsymbol{h}^{(1)}\right)$
    - Frame interpolation



Experiments

- Mandarin corpus ~80min
- Objective evaluation
  - HMM baseline = 0.223
- Subjective evaluation
  - outperform HMM baseline for modeling and predicting spectral features
  - the low-dimensional F0 features are not well modeled



System	MOS
HMM	2.86
DBN	2.88
MIX: DBN MGCs + HMM Log-F0	3.09

Table 1. MOS test result.

[Kang 2013]

Extensions

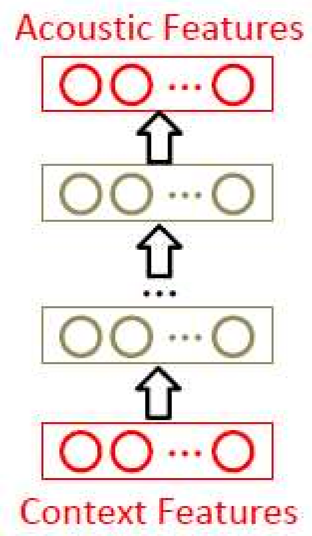
- Visual Speech Synthesis [Liu 2015]
  - 2D image-based approach
  - HMM-based lip movement generation
  - Using RBM/DBN to model visual features for HMM states
    - PCA coefficients or raw pixels as visual features
    - RBM for each HMM state
    - DBN for joint modeling of context features and visual features

baseline RBM-PCA RBM-PXL DBN-PXL DBN-PXL

Input-to-feature mapping using deep-structured NNs

Framework

- Motivation
  - To better describe the complex dependency between input context features and output acoustic
- Method
  - Model the conditional distribution p(y | x) directly using deep conditional models, e.g. DNNs or RNNs
    - x input context features
    - y output acoustic features



History

- Application of NNs in speech synthesis since 1980's

(1986)
Terrence J. Sejnowski and Charles R. Rosenberg

NETtalk: a parallel network that learns to read aloud
The Johns Hopkins University Electrical Engineering and Computer Science Technical Report
JHU/EECS-86/01, 32 pp.

- Popularity of DNN-based acoustic modeling for speech recognition since 2009
- The first attempt of DNN-based acoustic modeling for speech synthesis at ICASSP 2013 [Zen 2013]

STATISTICAL PARAMETRIC SPEECH SYNTHESIS USING DEEP NEURAL NETWORKS

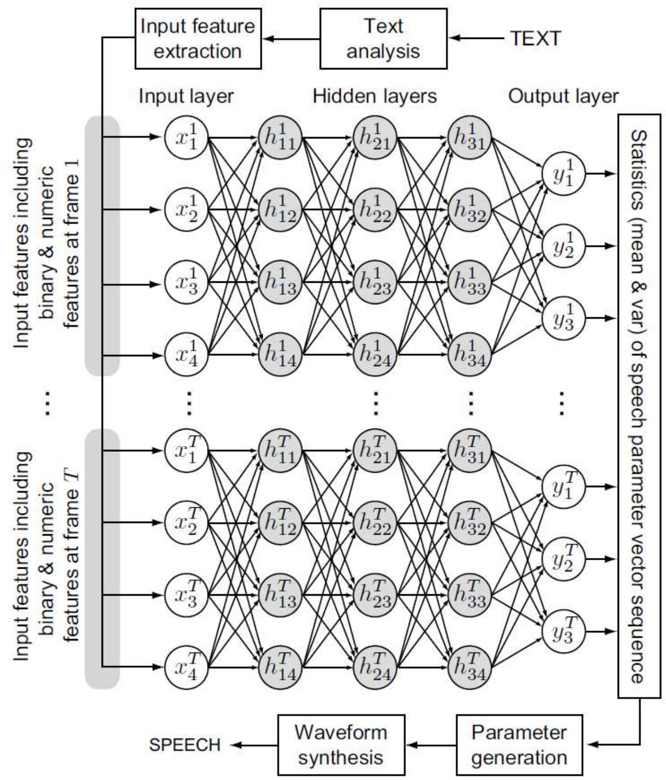
Heiga Zen, Andrew Senior, Mike Schuster

Google

{heigazen,andrewsenior,schuster}@google.com

Implementation

- Input linguistic features
  - frame-level
    - binary answers to questions about contexts
    - numeric context descriptors
    - position of current frame within a segment
    - segment durations
  - HMM-based alignment is necessary
- Output acoustic features
  - frame-level (static+dynamic)
    - MCC
    - logF0
    - excitation aperiodicity
    - voiced/unvoiced flag

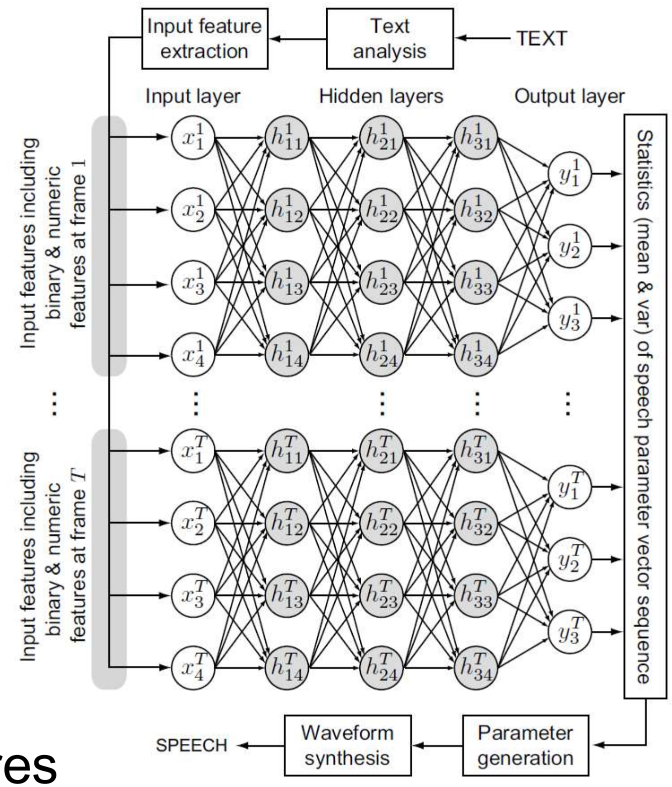


[Zen 2013]

Implementation

- Model training
  - sigmoid activation function
  - {input, output} pairs from training data
  - minimize mean square error
  - random initialization / BP training

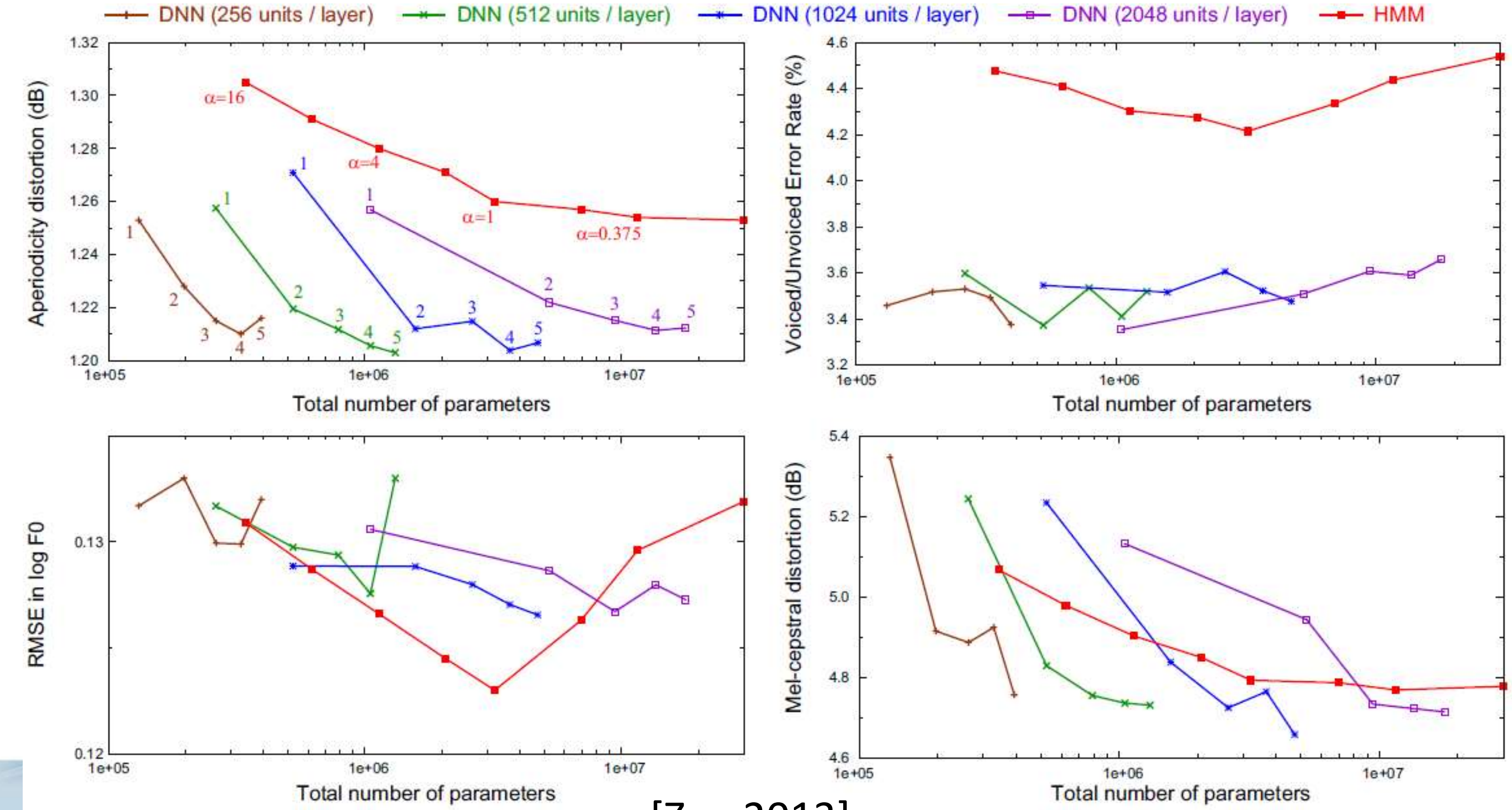
- Synthesis
  - text analysis
  - duration prediction
  - compose frame-level linguistic features
  - predict acoustic features using DNN
  - parameter generation with dynamic features
    - predicted output acoustic features as mean vectors
    - frame-independent variances of all training data



[Zen 2013]

Experiments

- Database
  - a US English female voice of 33,000 utterances
- Objective evaluation



[Zen 2013]

Experiments

- Subjective evaluation

HMM (α)	DNN (#layers $\times$ #units)	Neutral	p value	z value
15.8 (16)	**38.5** (4×256)	45.7	$< 10^{-6}$	-9.9
16.1 (4)	**27.2** (4×512)	56.8	$< 10^{-6}$	-5.1
12.7 (1)	**36.6** ($4 \times 1\,024$)	50.7	$< 10^{-6}$	-11.5

[Zen 2013]

  – The DNN-based system achieved better naturalness than the HMM-based one with similar number of parameters

Variations

- Model structure
- Representation of input features
- Representation of output features
- Training Criterion
- Other topics

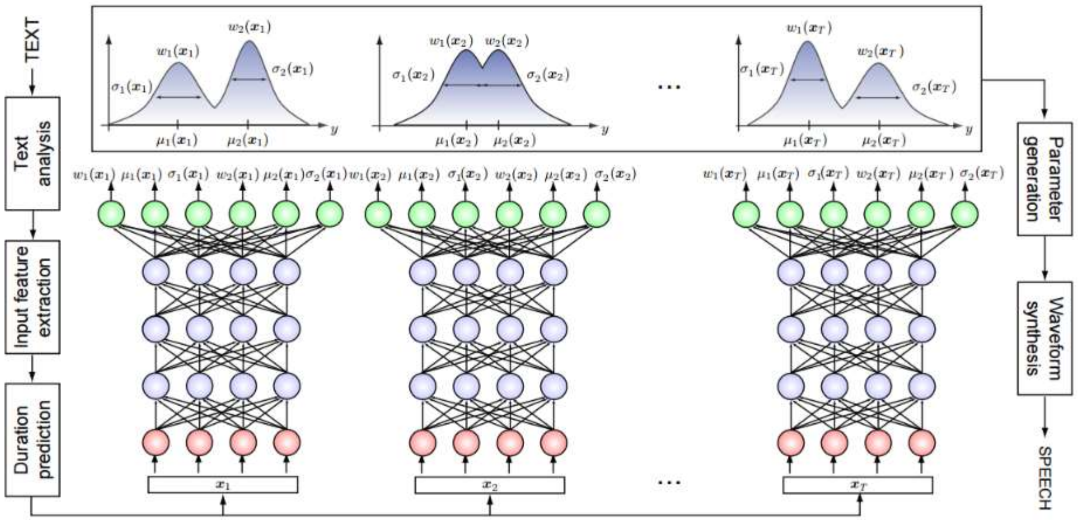
Variations

- Model structure
- Representation of input features
- Representation of output features
- Training Criterion
- Other topics

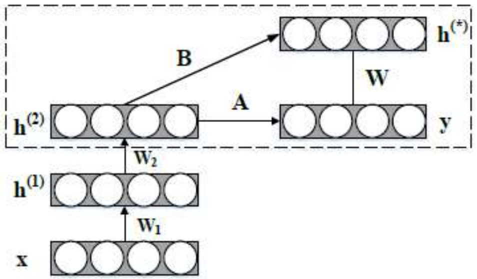
Variations

- Model structure
  - DNN→DMDN [Zen 2014] →DCRBM [Yin 2016a]
    - Provide better modelling ability of p(y|x)

	DNN	DMDN	DCRBM
p(y|x)	single Gaussian	GMM	RBM



Deep Mixture Density Network

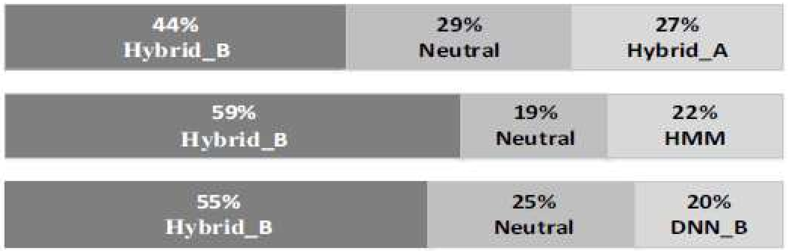


HMM-Baseline	*RBM-HMM*	*DNN-Baseline*	*DMDN*	*DCRBM*	N/P
15.00	–	–	–	74.38	10.62
–	30.62	–	–	58.75	10.63
–	–	18.75	–	69.38	11.87
–	–	–	21.88	63.75	9.38

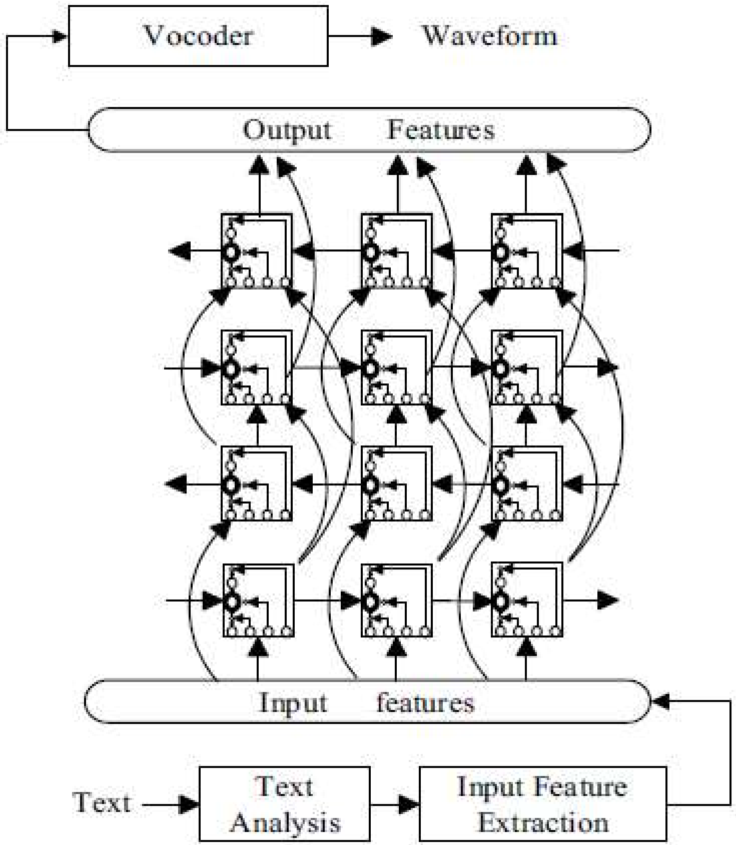
Deep Conditional Restricted Boltzmann Machine

Variations

- Model structure
  - DNN→RNN [Fan 2014]
    - Better capture temporal information for sequence transformation
    - Bidirectional Deep RNN
    - LSTM units



Hybrid_A	3 FF + 1 BLSTM
Hybrid_B	2 FF + 2 BLSTM



- A investigation on the effects of LSTM gate [Wu 2016]
  - The forget gate is the only critical component

Variations

- Model structure
- Representation of input features
- Representation of output features
- Training Criterion
- Other topics

Variations

- Representation of input features
  - Vector space representation of linguistic contexts [Lu 2013]
    - gather co-occurrence statistics of words/letters
    - derive low-dimensional representation of words/letters by SVD
    - only orthographic information (graphemes) used
    - require no language knowledge to build a model

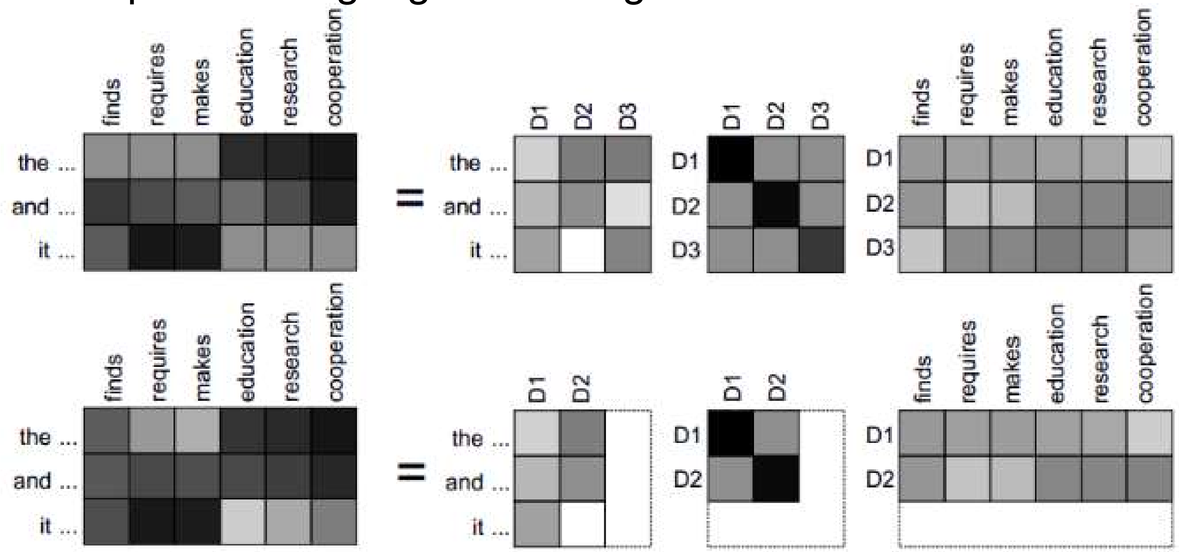
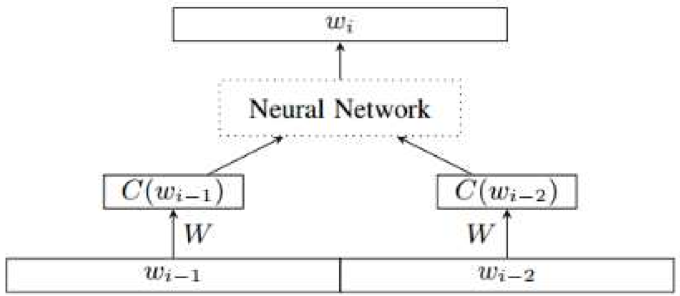


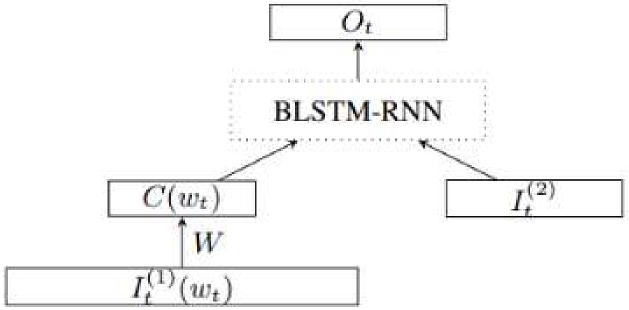
Figure 4.1: *Graphical toy example of the induction of word representations via singular value decomposition (a logarithmic grey-scale is used).* [Watts 2013]

Variations

- Representation of input features
  - Word embedding for RNN-based TTS [Wang 2015]
    - Word embedding: low-dimensional continuous-valued vector for words
    - Achieve word embeddings using neural network language model (NNLM)



structure of NNLM

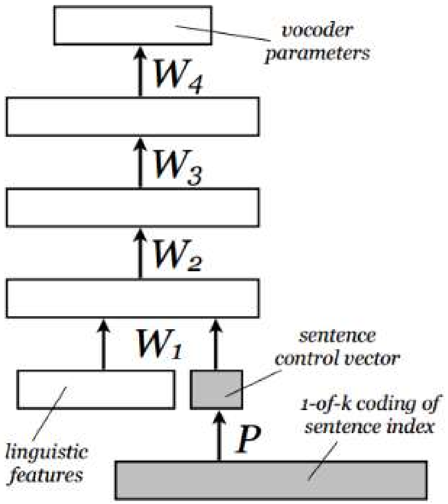


structure of BLSTM-RNN with word embeddings

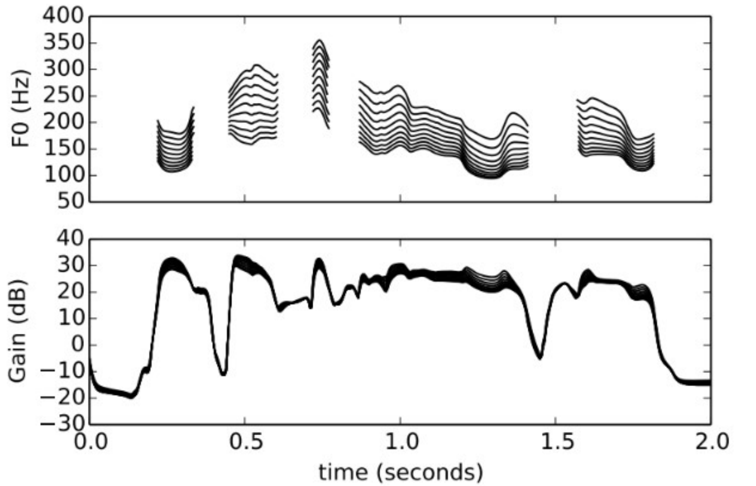
    - significantly improve the performance of the baseline system without using TOBI and POS as input features
    - still has a gap to the upper bound system, which uses manually labeled POS and TOBI as input features for both training and testing

Variations

- Representation of input features
  - Sentence-level control vector [Watts 2015]
    - Use a low-dimensional vector representation of sentence acoustics to control the output of DNNs
    - Learn sentence vectors together with other model parameters
    - Control the global prosodic characteristics of synthetic speech using sentence vectors at run time



model architecture



variation in synthetic F0 and gain controlled by sentence vectors

Variations

- Representation of input features
  - Speaker code for DNN-based speech synthesis [Hojo 2016]
    - To utilize multi-speaker corpus
    - To achieve speaker-adaptation under DNN framework

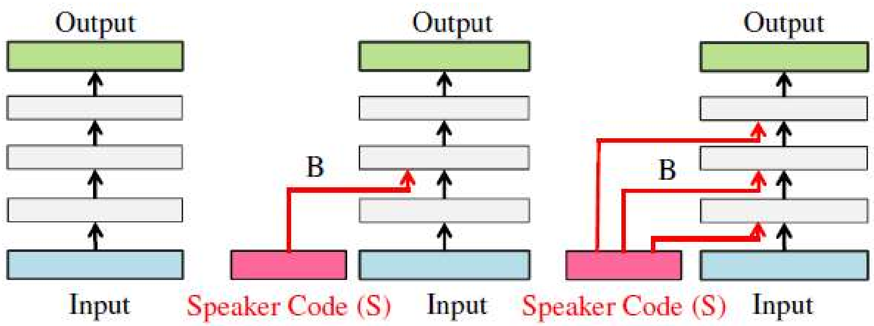


Figure 1: The architecture of DNNs. (left: the conventional model, middle: the proposed model using a single hidden layer, right: the proposed model using all hidden layers)

    - produce more natural speech than the speaker-dependent method
    - adaptation using speaker codes can achieve quality comparable to or better than the conventional HMM-based methods

Variations

- Model structure
- Representation of input features
- Representation of output features
- Training Criterion
- Other topics

Variations

- Representation of output features
  - Low-dimensional spectral features, e.g.
    - mel-cepstral coefficients [Zen 2013]
    - line spectral pairs [Fan 2014]
  - Raw spectral envelopes extracted by STRAIGHT [Yin 2016a]
  - Complex-valued spectral features [Hu 2016]
  - Speech waveforms [Tokuda 2016]

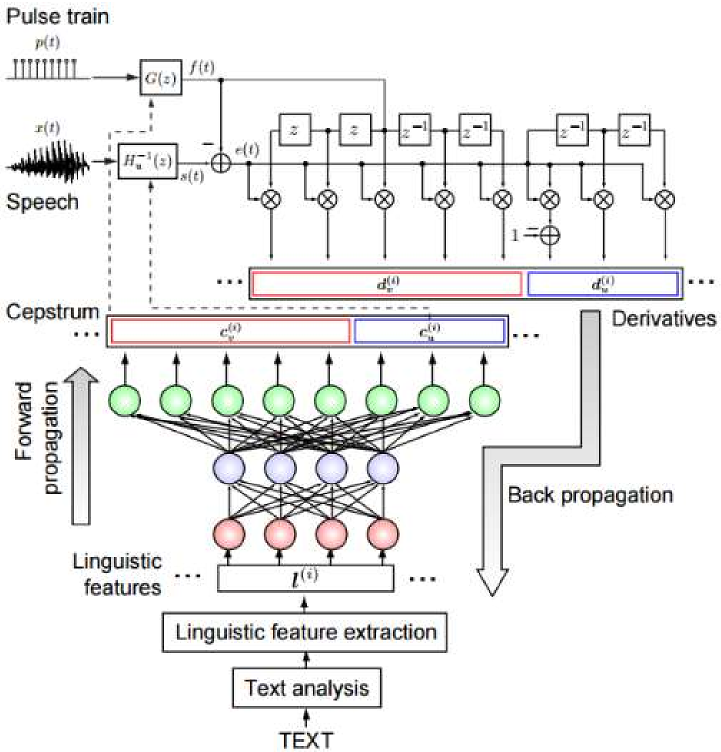


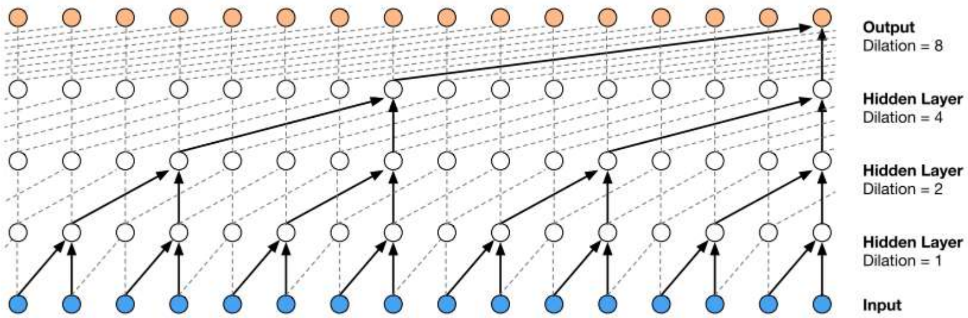
diagram of waveform-based framework

Variations

- Representation of output features
  - WaveNet by DeepMind [van den Oord 2016]
    - Model the joint probability of a waveform using a product of conditional PDFs

$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t \mid x_1, \ldots, x_{t-1})$$

    - The conditional PDF is modelled by a stack of convolutional layers



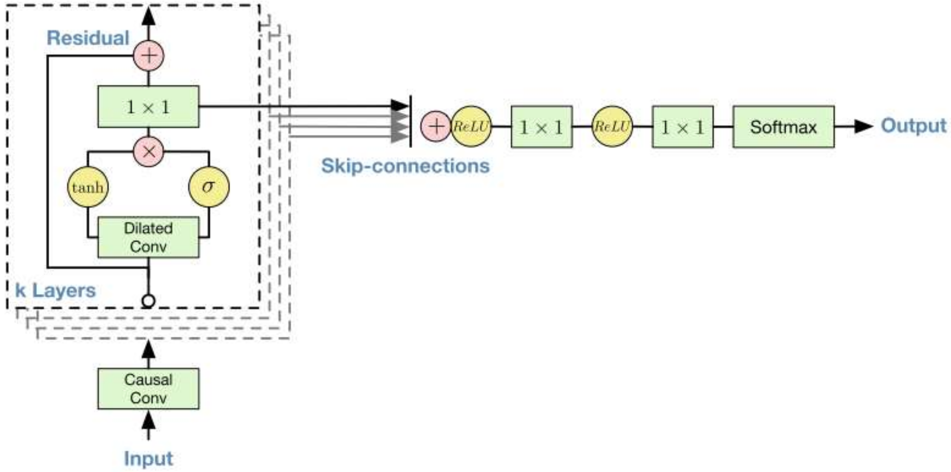
dilated causal convolutional layers to increase receptive field

    - Gated convolution: works better than ReLU

$$\mathbf{z} = \tanh(W_{f,k} * \mathbf{x}) \odot \sigma(W_{g,k} * \mathbf{x})$$

Variations

- Representation of output features
  - WaveNet by DeepMind [van den Oord 2016]
    - Softmax at output layer
      - μ-law companding, 16bit → 8bit, 65536 → 256
    - Residual and skip connections for entire architecture



    - Conditional WaveNet for integrating linguistic features for TTS

$$\mathbf{z} = \tanh\left(W_{f,k} * \mathbf{x} + V_{f,k} * \mathbf{y}\right) \odot \sigma\left(W_{g,k} * \mathbf{x} + V_{g,k} * \mathbf{y}\right)$$

transformed from linguistic features

Variations

- Representation of output features
  - WaveNet by DeepMind [van den Oord 2016]
    - Performance

Speech samples	Subjective 5-scale MOS in naturalness: North American English	Mandarin Chinese
LSTM-RNN parametric	3.67 ± 0.098	3.79 ± 0.084
HMM-driven concatenative	3.86 ± 0.137	3.47 ± 0.108
WaveNet (L+F)	$\mathbf{4.21} \pm 0.081$	$\mathbf{4.08} \pm 0.085$
Natural (8-bit μ-law)	4.46 ± 0.067	4.25 ± 0.082
Natural (16-bit linear PCM)	4.55 ± 0.075	4.21 ± 0.071

- unified NN structure for acoustic modeling + vocoder
- nonlinear adaptive filtering
- key points: wide receptive field + softmax output
- issues: prosodic modeling; efficiency at synthesis time

Variations

- Model structure
- Representation of input features
- Representation of output features
- Training Criterion
- Other topics

Variations

- Training criterion
  - Minimum perceptual error training [Valentini-Botinhao 2015]
    - Spectro-Temporal Excitation Pattern (STEP) domain for cost calculation

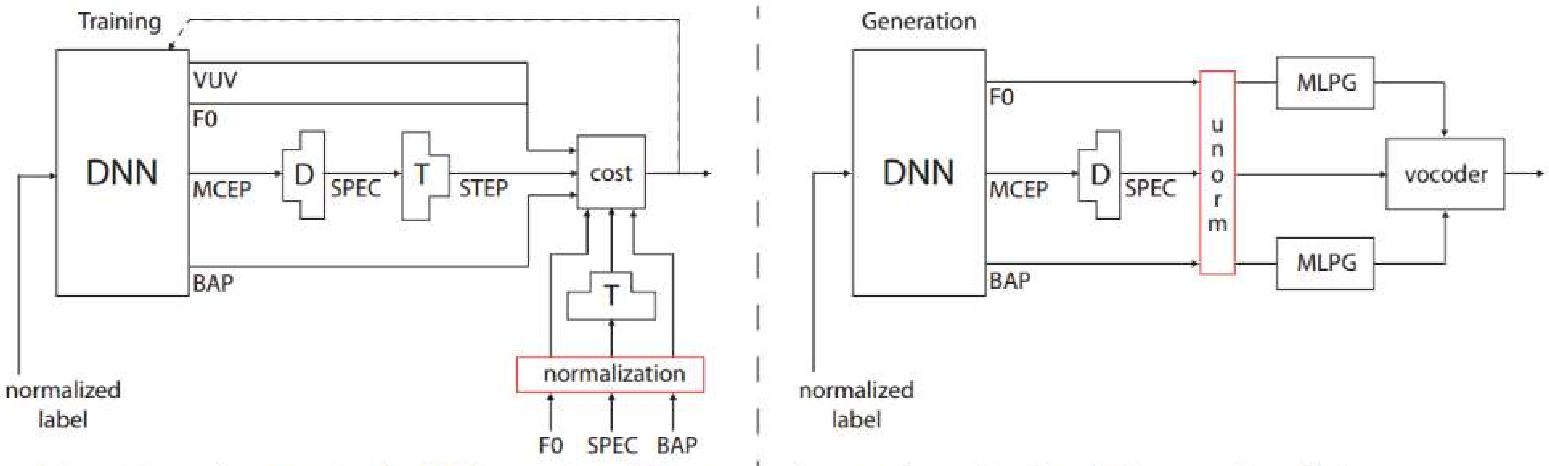
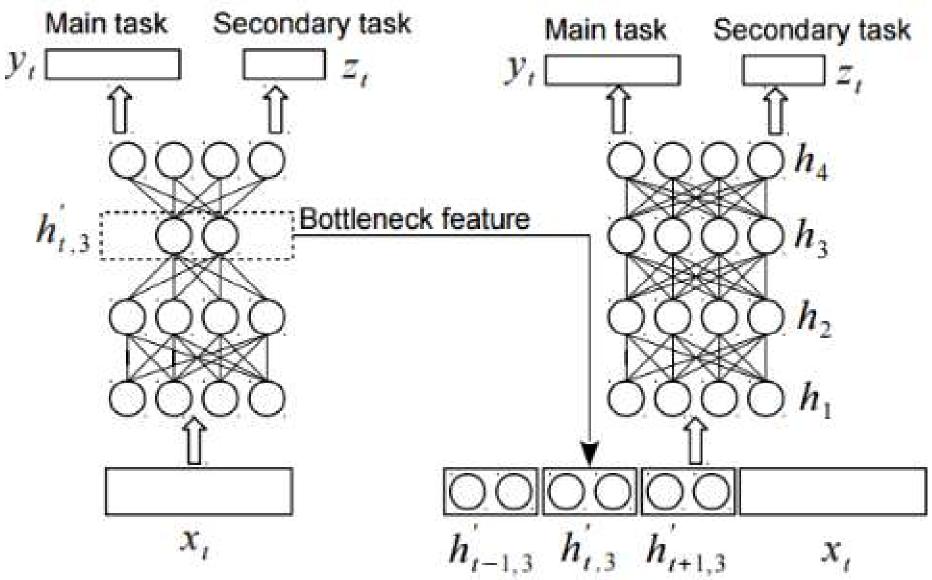


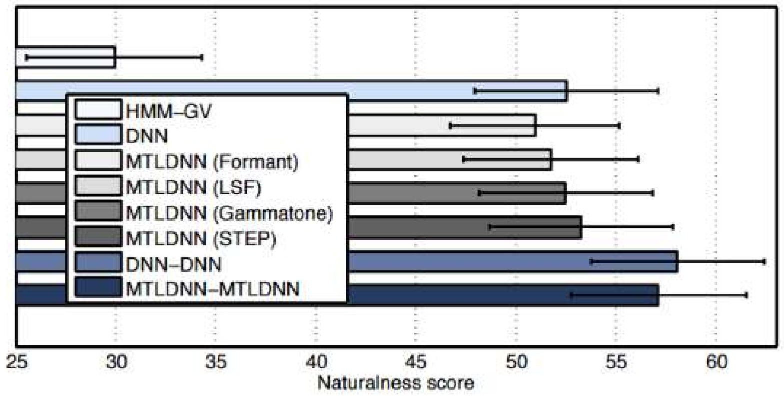
Figure 1: *Training and generation for DNN-step. D and T represent the transformation from Mel cepstral coefficients to spectrum and spectrum to STEP respectively.*

- Experimental results: warped log spectrum > STEP > mel-cepstra

Variations

- Training criterion
  - Multi-task learning and stacked bottleneck features [Wu 2015a]
    - to predict a perceptual representation of the target speech as a secondary task
    - to produce a wide context around the current frame using bottleneck features





MUSHRA evaluation results with 90% confidence interval

Variations

- Training criterion
  - Trajectory training considering global variance [Hashimoto 2016]
    - the inconsistency between training and synthesis criteria → trajectory training
    - the over-smoothing of generated parameter trajectories → GV

conventional objective function

$$\mathcal{L} = P(\boldsymbol{o}|\boldsymbol{\lambda}) = \mathcal{N}(\boldsymbol{o}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \prod_{t=1}^{T} \mathcal{N}(\boldsymbol{o}_t|\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_g)$$

objective function of trajectory training with $\boldsymbol{o} = \boldsymbol{W}\boldsymbol{c}$

$$\mathcal{L}_{Trj} = \frac{1}{Z} P(\boldsymbol{o}|\boldsymbol{\lambda}) = P(\boldsymbol{c}|\boldsymbol{\lambda}) = \mathcal{N}(\boldsymbol{c}|\bar{\boldsymbol{c}}, \boldsymbol{P})$$

objective function of trajectory training considering GV

$$\begin{aligned}\mathcal{L}_{GVTrj} &= P(\boldsymbol{c}|\boldsymbol{\lambda}) P(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{\lambda}, \boldsymbol{\lambda}_v)^{wT} \\ &= \mathcal{N}(\boldsymbol{c}|\bar{\boldsymbol{c}}, \boldsymbol{P}) \mathcal{N}(\boldsymbol{v}(\boldsymbol{c})|\boldsymbol{v}(\bar{\boldsymbol{c}}), \boldsymbol{\Sigma}_v)^{wT}\end{aligned}$$

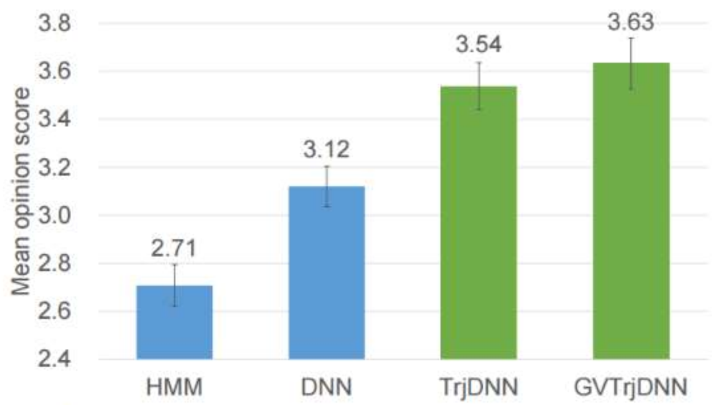


Fig. 2. Mean opinion scores of the four speech synthesis systems.

Variations

- Model structure
- Representation of input features
- Representation of output features
- Training Criterion
- Other topics

Variations

- Other topics
  - Multi-speaker & Multi-lingual
    - Multi-speaker & speaker adaptation [Fan 2015] [Wu 2015b]
    - Multi-lingual multi-speaker acoustic modeling [Li 2016]
    - Cross-lingual learning for low-resource languages [Yu 2016]
  - Modeling excitation features
    - Modeling F0 in hierarchically structured DNNs [Yin *et al.* 2016b]
    - Modeling glottal flow signals using DNNs [Raitio 2014]
    - Modeling SEW/REW components using DNNs [Song *et al.* 2015]
  - Practical implementation
    - Uni-directional LSTM-RNN for low-latency TTS [Zen 2015]
    - LSTM-RNN TTS on mobile devices [Zen 2016]

Outline

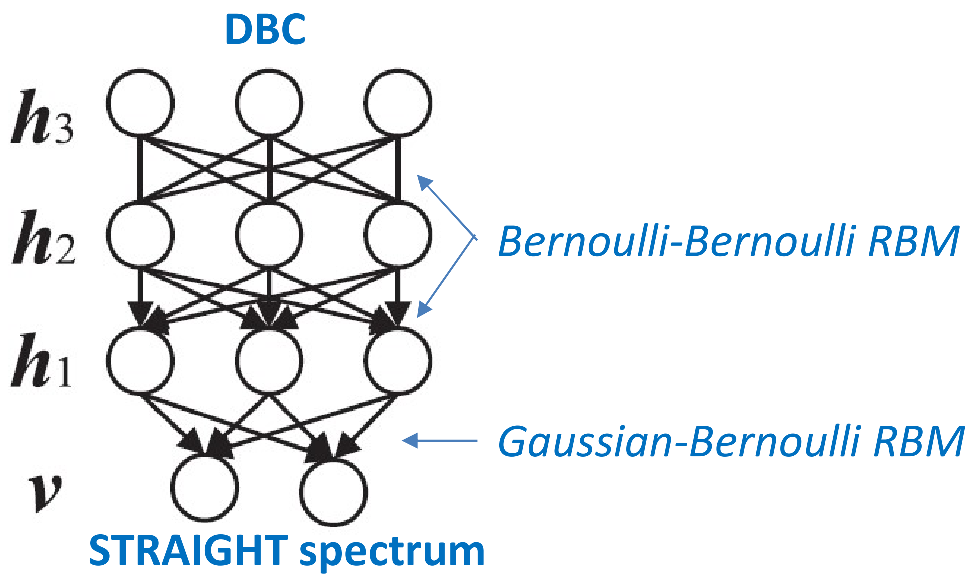
- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussion & Summary

Overview

- Aims
  - To extract spectral features from raw spectral representations for acoustic modeling using deep learning techniques
  - Raw spectral representations
    - Spectral envelope extracted by STRAIGHT [Kawahara 1999]
    - Spectral envelope extracted by WORLD [Morise 2015]
    - FFT spectrum
  - Deep learning techniques
    - DBN
    - Deep Auto-Encoder (DAE)

DBN-base feature extraction [Hu 2016]

- Train a DBN to model STRAIGHT spectral envelope
  - Binary samples are drawn as training data for upper layer RBMs

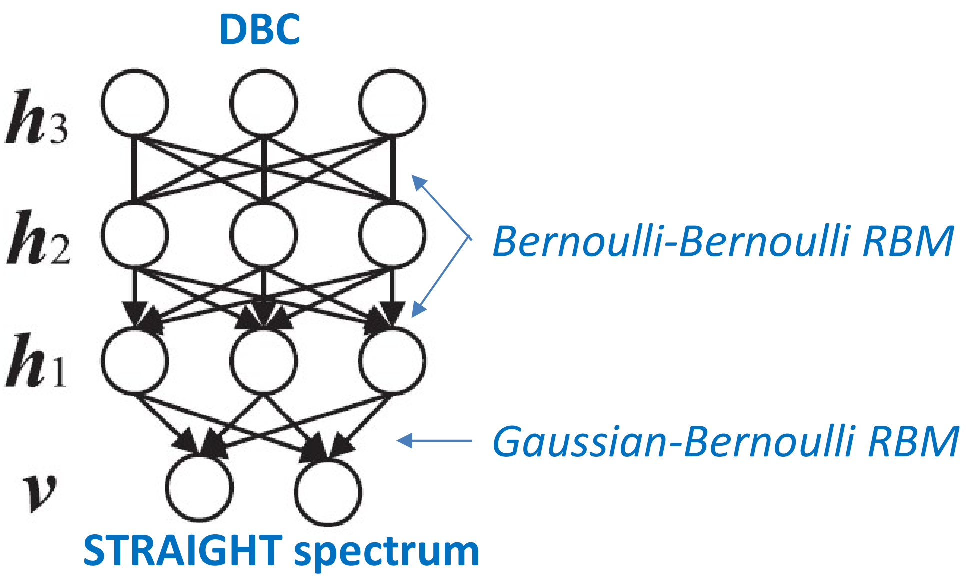


DBN-base feature extraction [Hu 2016a]

- Map STRAIGHT spectrum into binary codes
  - a visible feature vector $\tilde{\boldsymbol{v}}$ → DBN-based binary codes (DBC) $\tilde{\boldsymbol{h}}^L$

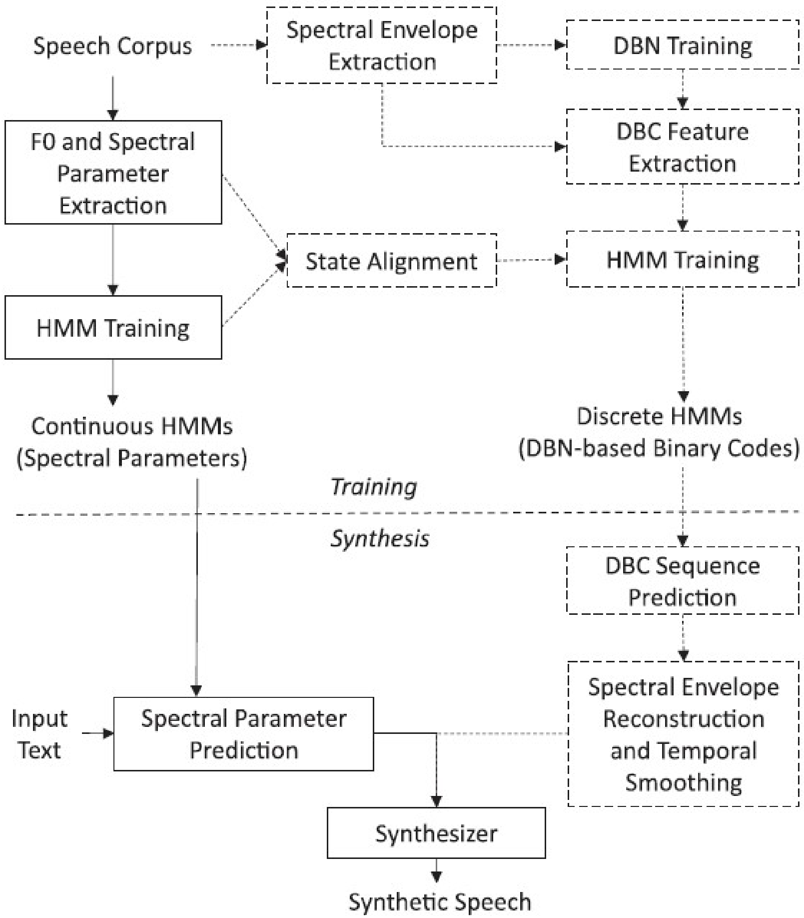
$$\tilde{h}_j^k = p(h_j^k = 1|\tilde{\boldsymbol{h}}^{k-1}) \qquad \tilde{\boldsymbol{h}}^0 = \tilde{\boldsymbol{v}}$$

  - $p(h_j^L = 1|\tilde{\boldsymbol{h}}^{L-1})$ are binarized using a threshold of 0.5 to obtain $\tilde{\boldsymbol{h}}^L$



DBN-base feature extraction [Hu 2016a]

- Use DBC in HMM-based acoustic modeling
  - model clustering / alignment using conventional HMM with mel-cepstra as spectral features
  - model DBCs with Bernoulli distributions at HMM states
  - maximum likelihood training
  - maximum output probability generation

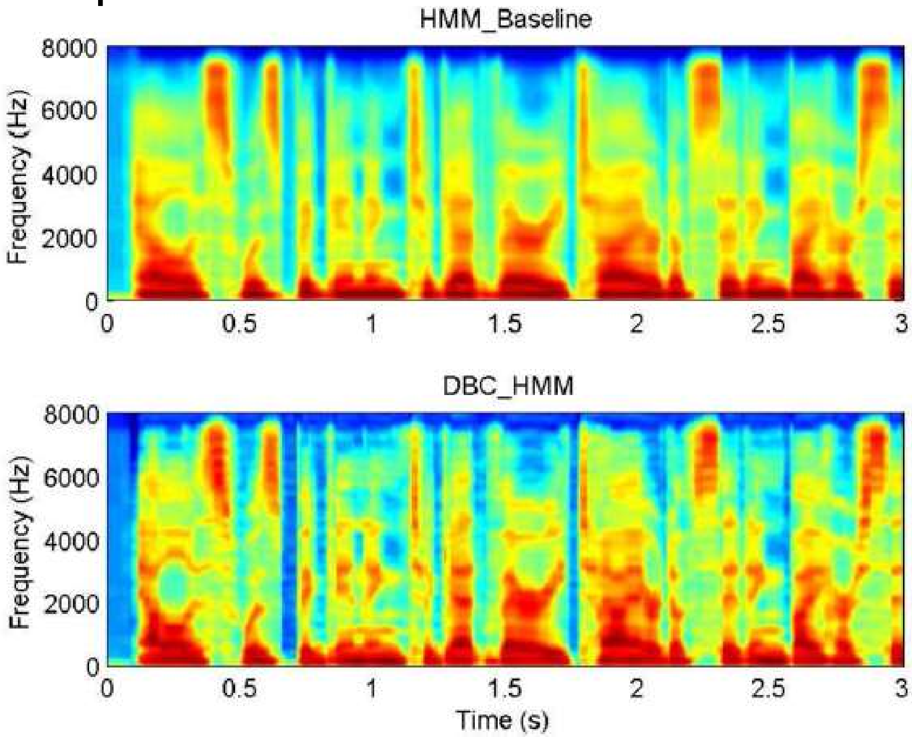


DBN-base feature extraction [Hu 2016a]

- Experiments
  - CMU ARCTIC database / female speaker SLT

DBC-HMM	*HMM-Baseline*	*HMM-GV*	*RBM-HMM*	N/P	*p*
68.3	17.5	-	-	14.2	< 0.001
47.0	-	20.2	-	23.8	< 0.001
56.0	-	-	24.2	28.8	< 0.001

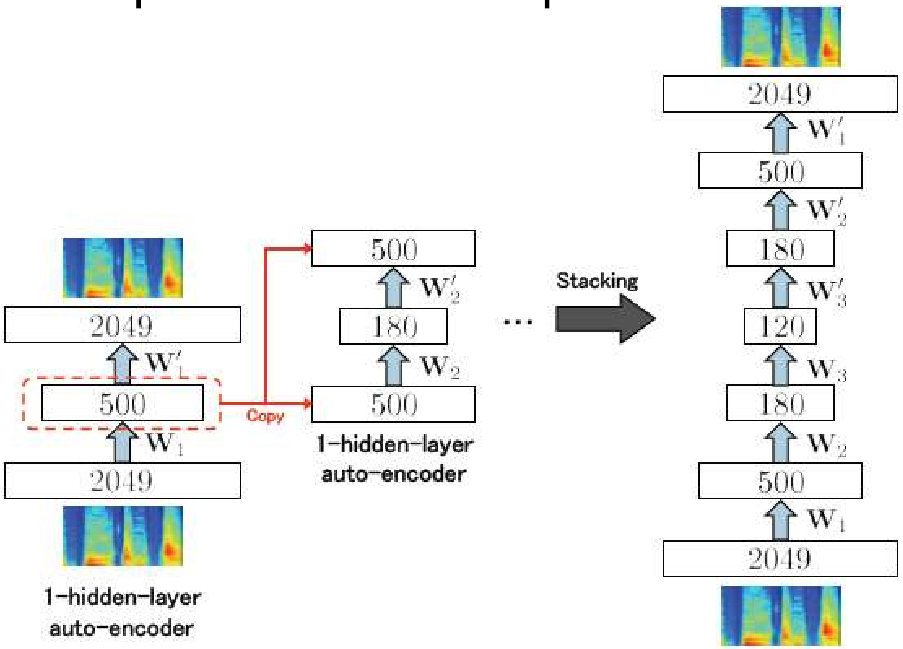
Preference Scores (%)



STRAIGHT spectrogram of synthetic speech

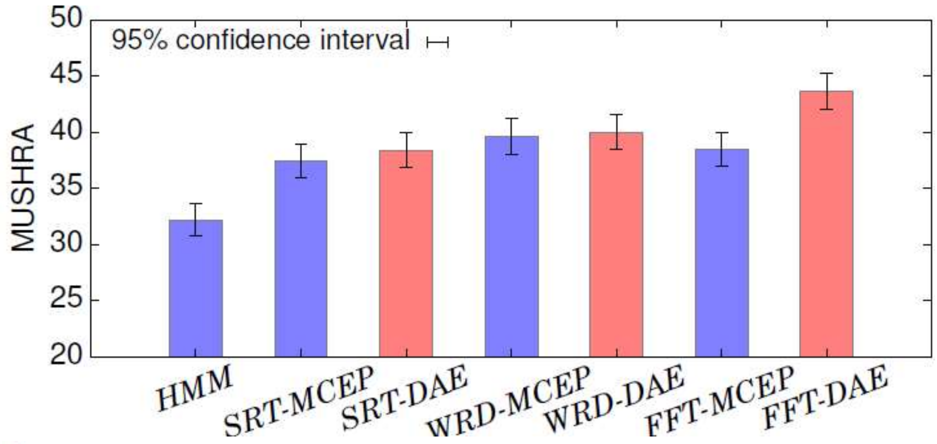
DAE-base feature extraction [Takaki 2016]

- Build a DAE to extract low-dimensional features from STRAIGHT/WORLD/FFT spectrum
- Deep auto-encoder (DAE)
  - an deep neural network with multiple layers of encoders and decoders to learn a compressed representation of input vector
  - layer-wise pre-training
  - minimum MSE fine-tuning
- DNN acoustic modeling

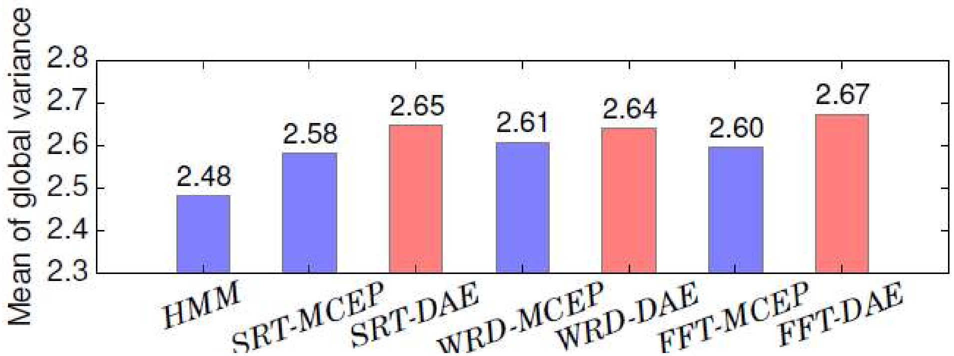


DAE-base feature extraction [Takaki 2016]

- Experiments
  - Blizzard Challenge 2011 database, 17hrs, 48kHz



Subjective results



Objective results
(mean GV of log spectra)

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussion & Summary

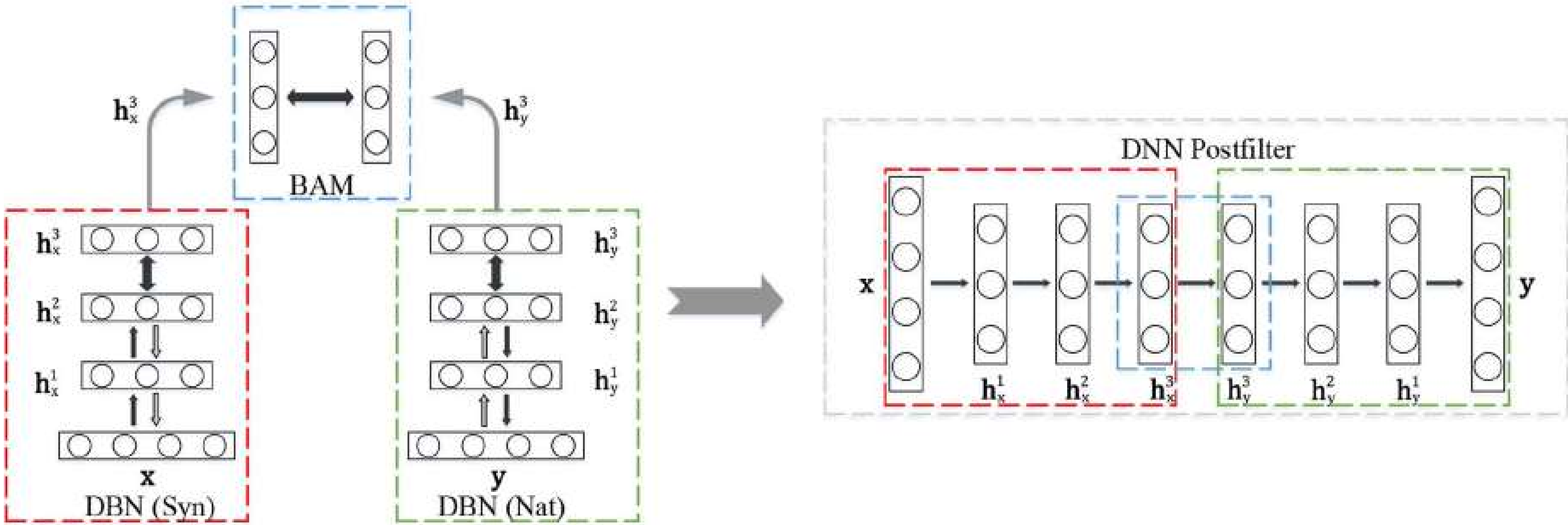
Overview

- Motivation
  - To deal with the over-smoothing effect of parameter generation

- Method
  - To map generated spectral features towards natural ones using DNNs or DBNs

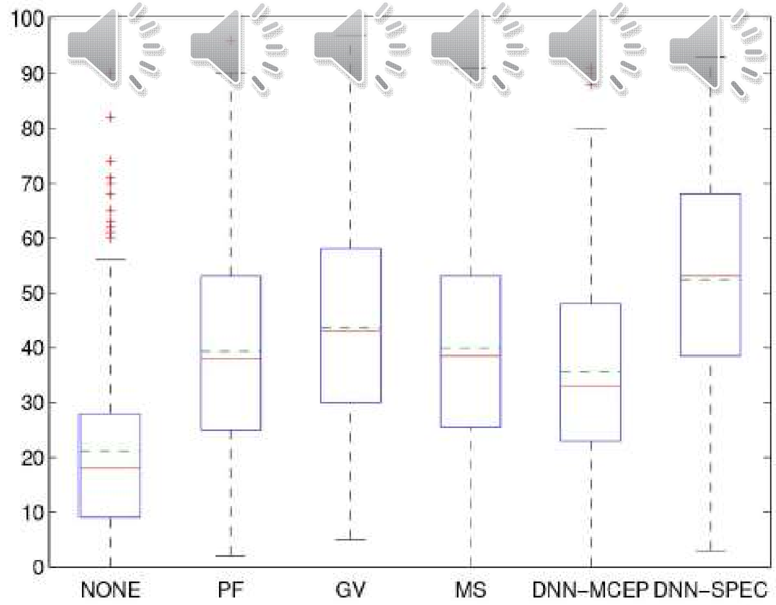
GTDNN for post-filtering [Chen 2015]

- Generatively trained DNN (GTDNN)
  - train a DNN in a generative way without fine-tuning to map generated spectral features towards natural ones
  - initially proposed for voice conversion

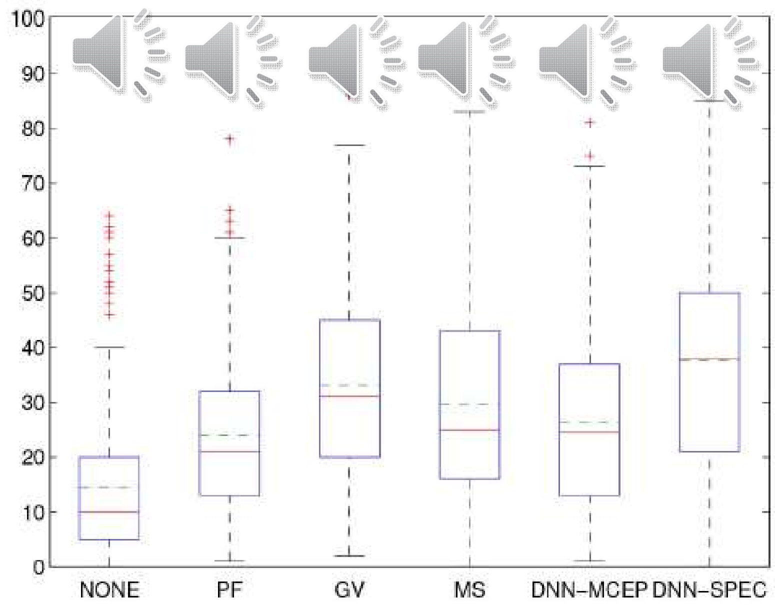


GTDNN for post-filtering [Chen 2015]

- Experiments
  - British male/Scottish female speakers (2840/4546 sentences); 48kHz
  - MUSHRA tests with other post-filtering techniques



male speaker



female speaker

  - Disadvantage: poste-filter depends on parameter generation

DBN for post-filtering [Hu 2016b]

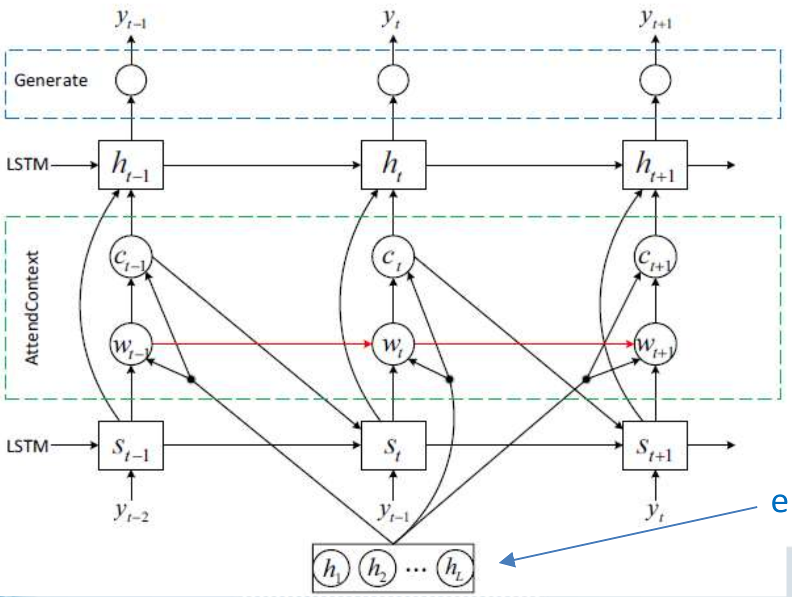
- A simplified version of GTDNN-based post-filtering
  - Discard the BAM for feature mapping
  - Use two identical DBNs trained from natural speech
- Training
  - Train a DBN similar to the DBN-based feature extraction
- Synthesis
  - Convert generated spectral features into spectral envelopes
  - Map spectral envelopes into DBCs in a bottom-up manner
  - Reconstruct spectral envelopes from DBCs in a top-down manner
- Performance
  - mel-cepstra: achieve equivalent performance to GV
  - LSPs: outperform the formant enhancement method

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussion & Summary

End-to-End SPSS

- Attention-based Recurrent Sequence Transducer (ARST) for end-to-end SPSS [Wang 2016]
  - Motivation
    - directly mapping from text sequence to acoustic trajectory
    - bypass text analysis / learn alignment
    - success of attention-based recurrent networks in machine translation, ASR, etc.
  - ARST generate $\boldsymbol{y} = (\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_T)$ from $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_L)$, where $T \gg L$



encoded representation of **x**

main architecture

$$s_t = RNN(s_{t-1}, c_{t-1}, y_{t-1})$$

$$c_t = AttendContext(s_t, h)$$

$$y_t = Generate(s_t, c_t)$$

attention selection

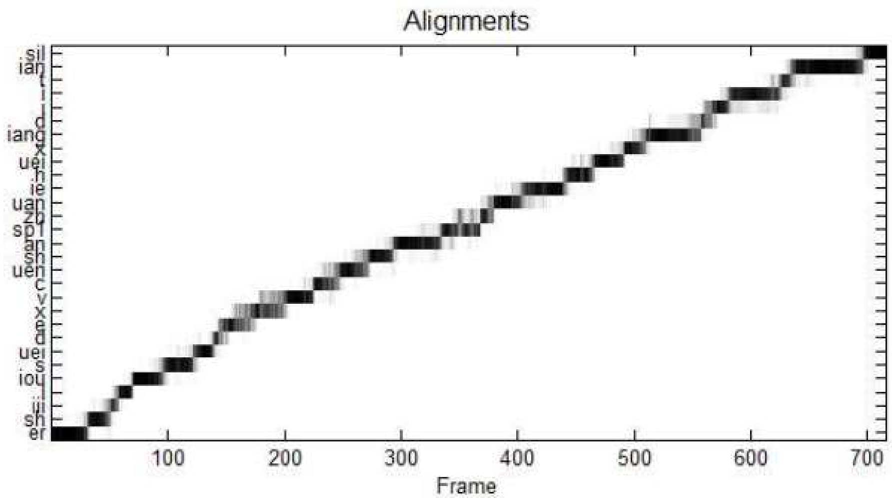
$$e_{t,i} = v^{\mathrm{T}} \tanh(W s_t + V h_i + b)$$

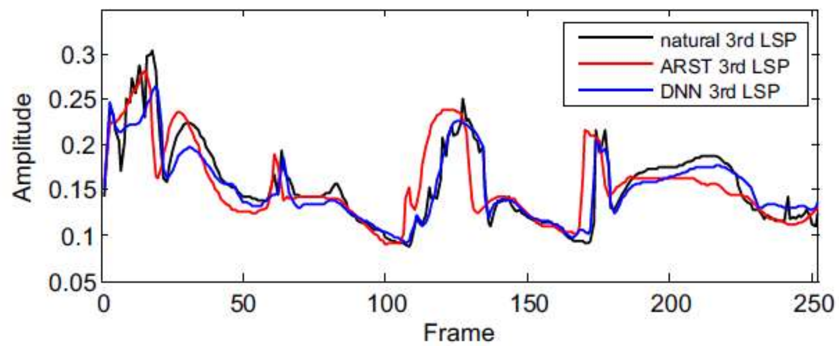
$$w_{t,i} = \exp(e_{t,i}) \Big/ \sum_{j=1}^{L} \exp(e_{t,j})$$

$$c_t = \sum_{i=1}^{L} w_{t,i} h_i$$

End-to-End SPSS

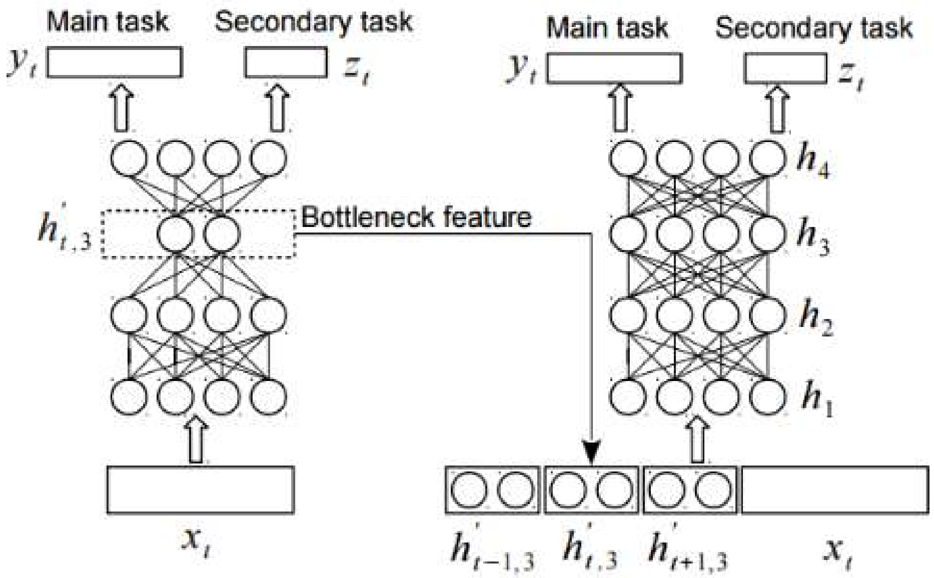
- Attention-based Recurrent Sequence Transducer (ARST) for end-to-end SPSS [Wang 2016]
  - Some specific techniques for applying ARST to TTS
  - Experiments
    - 7-hr Mandarin database; 16kHz
    - untoned phoneme → LSPs
    - ARST can generate smooth trajectories; fairly intelligible; inferior to DNN





Deep Learning for Unit Selection

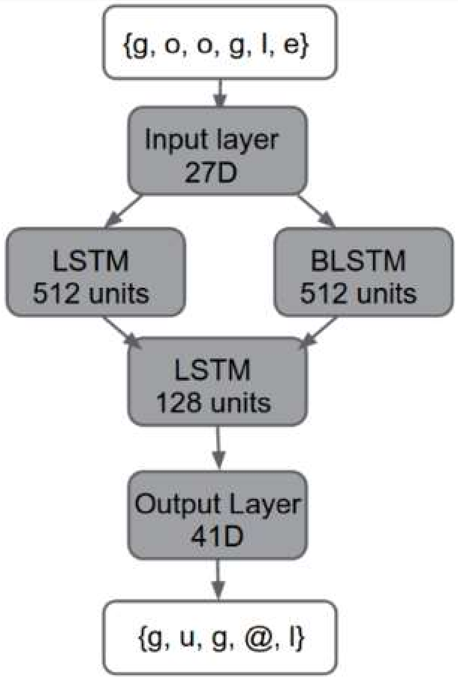
- DNN-guided unit selection [Merritt 2016]
  - Hybrid synthesis
    - Use statistical models to guide the selection of natural units
    - HMMs + acoustic feature domain
  - Proposed hybrid target cost
    - DNNs + context embedding (bottleneck feature) domain



[Wu 2015a]

Deep Learning for Text Analysis

- Grapheme-to-Phoneme Conversion using LSTM-RNN [Rao 2015]



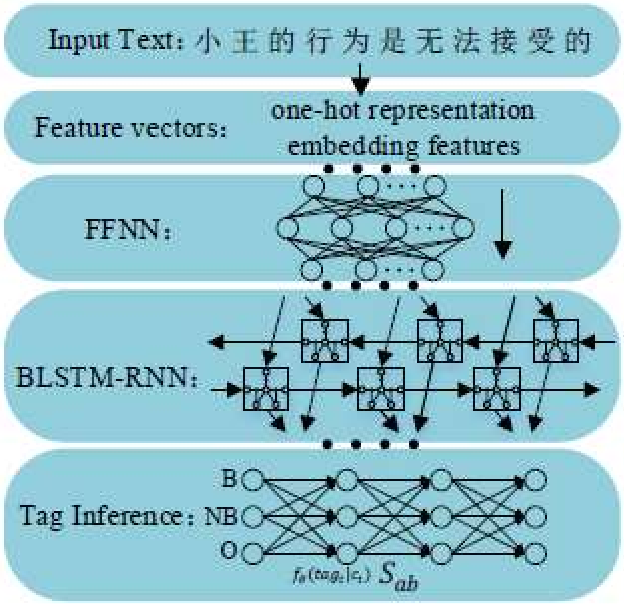
model structure

Model	Word Error Rate (%)
Galescu and Allen [4]	28.5
Chen [7]	24.7
Bisani and Ney [2]	24.5
Novak et al. [6]	24.4
Wu et al. [12]	23.4
5-gram FST	27.2
8-gram FST	26.5
Unidirectional LSTM with Full-delay	30.1
DBLSTM-CTC 128 Units	27.9
DBLSTM-CTC 512 Units	25.8
DBLSTM-CTC 512 + 5-gram FST	21.3

results on the CMU dataset

Deep Learning for Text Analysis

- Prosodic boundary prediction using BLSTM-RNN [Ding 2015]



model structure

Boundary	P (%)	R (%)	F (%)
PW	95.34	96.73	96.03
PPH	83.41	83.68	83.06
IPH	84.85	73.39	78.71

results of using CRF

Boundary	P (%)	R (%)	F (%)	Embedding feature size
PW	96.27	96.91	96.59	300
PPH	82.89	87.13	84.96	400
IPH	84.81	79.88	82.27	100

results of using BLSTM-RNN & word embeddings

Outline

- Statistical Parametric Speech Synthesis (SPSS)
- HMM-Based SPSS
- Some Key Techniques of Deep Learning
- Deep Learning Based Acoustic Modeling for SPSS
- Deep Learning Based Feature Representation for SPSS
- Deep Learning Based Post-Filtering for SPSS
- Other Applications of Deep Learning for Speech Synthesis
- Discussions & Summary

Discussion

- Deep learning in ASR
  - acoustic model: to map acoustic features towards posterior probabilities of senones using various NN architectures
  - language model: to predict current word using context words

- Issues of applying deep learning to TTS
  - rich context features
  - detailed spectral representations
  - long-term dependency, especially for prosodic features
  - perceptual-related objective function
  - comparison / integration with existing techniques
  - common datasets for evaluation

Discussion

- Future directions
  - to grow with the development of deep learning techniques
    - DNN → LSTM-RNN → PixelCNN → …
  - towards unified modeling
    - acoustic modeling + vocoder
    - text analysis + acoustic modeling
  - to be more flexible
    - multi-speaker / multi-lingual / multi-style / expressive
  - to make use of big data
    - 1 hr → 10 hrs → 100 hrs → …

Software

- HMM-based Speech Synthesis System (HTS)
  - http://hts.sp.nitech.ac.jp/?Home

- The Merlin toolkit
  - For building neural networks for SPSS
  - http://www.cstr.ed.ac.uk/projects/merlin/

- Toolkits for NN implementation
  - Theano http://deeplearning.net/software/theano/
  - TensorFlow https://www.tensorflow.org/
  - CNTK https://www.cntk.ai/

Summary

- the limitations of conventional HMM-based SPSS
- some basic techniques of deep learning, e.g., RBM, DBN, DNN, RNN
- various ways of applying deep learning techniques to SPSS, including acoustic modeling, feature representation, and post-filtering, which improved the quality of SPSS effectively
- three approaches to deep-learning-based acoustic modeling for SPSS
- detailed review on the acoustic modeling of SPSS using deep NNs
- the topics to be explored in the future

Thanks for your attention !

References

- Klatt D. The Klattalk text-to-speech conversion system[C]//Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'82. IEEE, 1982, 7: 1589-1592.
- Moulines E, Charpentier F. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones[J]. Speech communication, 1990, 9(5-6): 453-467.
- Sagisaka Y. ATR v-talk speech synthesis system[J]. Proc. ICSLP, 1992, 1992.
- Hunt A J, Black A W. Unit selection in a concatenative speech synthesis system using a large speech database[C]//Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on. IEEE, 1996, 1: 373-376.
- Beutnagel M, Conkie A, Schroeter J, et al. The AT&T next-gen TTS system[C]//Joint meeting of ASA, EAA, and DAGA. 1999: 18-24.
- Yoshimura T, Tokuda K, Masuko T, et al. Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis[C]//Proc. Eurospeech. 1999: 2347-2350.
- Shinoda K, Watanabe T. MDL-based context-dependent subword modeling for speech recognition[J]. The Journal of the Acoustical Society of Japan (E), 2000, 21(2): 79-86.
- Tokuda K, Yoshimura T, Masuko T, et al. Speech parameter generation algorithms for HMM-based speech synthesis[C]//Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on. IEEE, 2000, 3: 1315-1318.
- Kawahara H, Masuda-Katsuse I, De Cheveigne A. Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds[J]. Speech communication, 1999, 27(3): 187-207.
- Zen H, Tokuda K, Kitamura T. Reformulating the HMM as a trajectory model by imposing explicit relationships between static and dynamic feature vector sequences[J]. Computer Speech & Language, 2007, 21(1): 153-173.

References

- Wu Y J, Wang R H. Minimum generation error training for HMM-based speech synthesis[C]//2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings. IEEE, 2006, 1: I-I.
- Toda T, Tokuda K. A speech parameter generation algorithm considering global variance for HMM-based speech synthesis[J]. IEICE TRANSACTIONS on Information and Systems, 2007, 90(5): 816-824.
- Ling Z H, Dai L R. Minimum kullback–leibler divergence parameter generation for hmm-based speech synthesis[J]. IEEE Transactions on Audio, Speech, and Language Processing, 2012, 20(5): 1492-1502.
- Takamichi S, Toda T, Black A W, et al. Parameter generation algorithm considering modulation spectrum for HMM-based speech synthesis[C]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015: 4210-4214.
- Salakhutdinov R. Learning deep generative models[D]. University of Toronto, 2009.
- Hinton G E. Training products of experts by minimizing contrastive divergence[J]. Neural computation, 2002, 14(8): 1771-1800.
- Neal R M. Connectionist learning of belief networks[J]. Artificial intelligence, 1992, 56(1): 71-113.
- Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. Science, 2006, 313(5786): 504-507.
- Rumelhart D E, Hinton G E, Williams R J. Learning internal representations by error propagation[R]. CALIFORNIA UNIV SAN DIEGO LA JOLLA INST FOR COGNITIVE SCIENCE, 1985.
- Hopfield J J. Neural networks and physical systems with emergent collective computational abilities[J]. Proceedings of the national academy of sciences, 1982, 79(8): 2554-2558.
- Werbos P J. Backpropagation through time: what it does and how to do it[J]. Proceedings of the IEEE, 1990, 78(10): 1550-1560.
- Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780.

References

- Graves A, Mohamed A, Hinton G. Speech recognition with deep recurrent neural networks[C]//2013 IEEE international conference on acoustics, speech and signal processing. IEEE, 2013a: 6645-6649.
- Graves A. Generating sequences with recurrent neural networks[J]. arXiv preprint arXiv:1308.0850, 2013b
- Ling Z H, Deng L, Yu D. Modeling spectral envelopes using restricted Boltzmann machines and deep belief networks for statistical parametric speech synthesis[J]. IEEE Transactions on Audio, Speech, and Language Processing, 2013, 21(10): 2129-2139.
- Yin X, Ling Z H, Dai L R. Spectral modeling using neural autoregressive distribution estimators for statistical parametric speech synthesis[C]//2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014: 3824-3828.
- Chen L H, Ling Z H, Song Y, et al. Joint spectral distribution modeling using restricted boltzmann machines for voice conversion[C]//Interspeech. 2013: 3052-3056.
- Kang S, Qian X, Meng H. Multi-distribution deep belief network for speech synthesis[C]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2013: 8012-8016.
- Liu Z C, Ling Z H, Dai L R. LIP movement generation using restricted Boltzmann machines for visual speech synthesis[C]//Signal and Information Processing (ChinaSIP), 2015 IEEE China Summit and International Conference on. IEEE, 2015: 606-610.
- Zen H, Senior A, Schuster M. Statistical parametric speech synthesis using deep neural networks[C]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2013: 7962-7966..
- Zen H, Senior A. Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis[C]//2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014: 3844-3848.

References

- Yin X, Ling Z H, Hu Y J, et al. Modeling spectral envelopes using deep conditional restricted Boltzmann machines for statistical parametric speech synthesis[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016a: 5125-5129.
- Fan Y, Qian Y, Xie F L, et al. TTS synthesis with bidirectional LSTM based recurrent neural networks[C]//Interspeech. 2014: 1964-1968.
- Wu Z, King S. Investigating gated recurrent networks for speech synthesis[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5140-5144.
- Lu H, King S, Watts O. Combining a vector space representation of linguistic context with a deep neural network for text-to-speech synthesis[J]. Proc. ISCA SSW8, 2013: 281-285.
- Wang P, Qian Y, Soong F K, et al. Word embedding for recurrent neural network based tts synthesis[C]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015: 4879-4883.
- Watts O, Wu Z, King S. Sentence-level control vectors for deep neural network speech synthesis[C]//Proc. Interspeech. 2015.
- Hojo N, Ijima Y, Mizuno H. An Investigation of DNN-Based Speech Synthesis Using Speaker Codes}[J]. Interspeech 2016, 2016: 2278-2282.
- Hu Q, Yamagishi J, Richmond K, et al. Initial investigation of speech synthesis based on complex-valued neural networks[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5630-5634.

References

- Tokuda K, Zen H. Directly modeling voiced and unvoiced components in speech waveforms by neural networks[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5640-5644.
- Oord A, Dieleman S, Zen H, et al. WaveNet: A Generative Model for Raw Audio[J]. arXiv preprint arXiv:1609.03499, 2016.
- Valentini-Botinhao C, Wu Z, King S. Towards minimum perceptual error training for DNN-based speech synthesis[C]//Proc. Interspeech. 2015.
- Wu Z, Valentini-Botinhao C, Watts O, et al. Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis[C]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015a: 4460-4464.
- Hashimoto K, Oura K, Nankaku Y, et al. Trajectory training considering global variance for speech synthesis based on neural networks[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5600-5604.
- Fan Y, Qian Y, Soong F K, et al. Multi-speaker modeling and speaker adaptation for DNN-based TTS synthesis[C]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015: 4475-4479.
- Wu Z, Swietojanski P, Veaux C, et al. A study of speaker adaptation for DNN-based speech synthesis[C]//Proceedings interspeech. 2015b.
- Li B, Zen H. Multi-Language Multi-Speaker Acoustic Modeling for LSTM-RNN based Statistical Parametric Speech Synthesis[J]. Interspeech 2016}, 2016: 2468-2472.
- Yu Q, Liu P, Wu Z, et al. Learning cross-lingual information with multilingual BLSTM for speech synthesis of low-resource languages[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5545-5549.

References

- Raitio T, Lu H, Kane J, et al. Voice source modelling using deep neural networks for statistical parametric speech synthesis[C]//2014 22nd European Signal Processing Conference (EUSIPCO). IEEE, 2014: 2290-2294.
- Song E, Kang H G. Deep Neural Network-Based Statistical Parametric Speech Synthesis System Using Improved Time-Frequency Trajectory Excitation Model[C]//Sixteenth Annual Conference of the International Speech Communication Association. 2015.
- Yin X, Lei M, Qian Y, et al. Modeling F0 trajectories in hierarchically structured deep neural networks[J]. Speech Communication, 2016b, 76: 82-92.
- Zen H, Sak H. Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis[C]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015: 4470-4474.
- Zen H, Agiomyrgiannakis Y, Egberts N, et al. Fast, compact, and high quality LSTM-RNN based statistical parametric speech synthesizers for mobile devices[J]. arXiv preprint arXiv:1606.06061, 2016.
- Morise M. Cheaptrick, a spectral envelope estimator for high-quality speech synthesis[J]. Speech Communication, 2015, 67: 1-7.
- Hu Y J, Ling Z H. DBN-based Spectral Feature Representation for Statistical Parametric Speech Synthesis[J]. IEEE Signal Processing Letters, 2016, 23(3): 321-325.
- Takaki S, Yamagishi J. A DEEP AUTO-ENCODER BASED LOW-DIMENSIONAL FEATURE EXTRACTION FROM FFT SPECTRAL ENVELOPES FOR STATISTICAL PARAMETRIC SPEECH SYNTHESIS[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016.
- Chen L H, Raitio T, Valentini-Botinhao C, et al. A deep generative architecture for postfiltering in statistical parametric speech synthesis[J]. IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP), 2015, 23(11): 2003-2014.

References

- Hu Y J, Ling Z H, Dai L R. Deep belief network-based post-filtering for statistical parametric speech synthesis[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016b: 5510-5514.
- Rao K, Peng F, Sak H, et al. Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks[C]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015: 4225-4229.
- Ding C, Xie L, Yan J, et al. Automatic prosody prediction for Chinese speech synthesis using BLSTM-RNN and embedding features[C]//2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, 2015: 98-102.
- Wang W, Xu S, Xu B. First Step Towards End-to-End Parametric TTS Synthesis: Generating Spectral Parameters with Neural Attention[C]//Proceedings Interspeech. 2016: 2243-2247.
- Merritt T, Clark R A J, Wu Z, et al. Deep neural network-guided unit selection synthesis[C]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5145-5149.