Deep Learning Strategies for Voice Conversion

CSLU Seminar 03/10/2014

Seyed. Hamidreza. Mohammadi

Center for Spoken Language Understanding (CSLU)
Oregon Health & Science University (OHSU)
Portland, Oregon, USA

February 3, 2015

Voice Conversion

Voice Conversion (VC)

process the speech of a *source* speaker to sound like a *target* speaker

- Applications
  - personalized TTS
    - for individuals with disabilities
    - message readers with custom/sender identities
  - movie dubbing
  - interpretive services by human or machine
- Important criteria
  - speaker recognizability
  - speech quality

Voice Conversion

Generative approaches for VC:

- Generative
    - source-filter speech model (source: vocal cords, filter: vocal tract)
    - compact parametrization of speech as parameters
    - we assume we have a parallel sentence corpus of source and target speakers
    - direct mapping from source x to target y parameters
    - quality is limited by parametric vocoder

LSFs

1. Linear Predictive Coding (LPC) coefficients
   - they model spectral peaks
   - interpolating LPCs may cause unstable filters
2. LSFs are another representation of LPCs
   - they represent spectral peaks directly
   - The main problem of LSFs is that: one specific LSF coefficient does not necessarily represent the same formant
   - For 16kHz speech, 18 coefficients
   - Two similar spectral may not have similar LSFs

MCEPs

1. Mel Cepstrum (MCEP) coefficients
   - they model spectrum directly
   - they weight peaks and valleys equally
   - The main problem of MCEPs is oversmoothing, since we average a lot of frames and it leads to wide formants
   - For 16kHz speech, 24 coefficients
   - Two similar spectral do have similar MCEPs

LPCs vs MCEPs

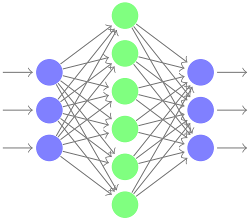
images/cep_vs_lpc.png

Autoencoders

- Deep AutoEncoders (DAEs) have been used for pre-training and feature extraction, specially in image and text processing literature
- compute speech feature using a DAE
- Autoencoder (AE):
  - The encoder: $y = f(Wx + b)$ where x and y, W and b are the input, output, weights and bias, respectively.
  - The decoder: $\tilde{x} = g(W'y + b')$.
  - f and g are usually non-linear functions (sigmoid or tanh)
  - weighs are tied $W' = W^\top$
- DAE:
  - Multiple AEs are trained layer-by-layer and stacked together.
  - The output of the last layer can be treated as a new feature type.

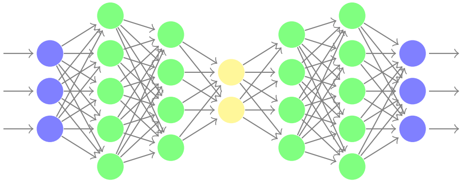
AEs

- Auto Encoders (AEs)



AEs

- Deep AutoEncoders (DAEs)



Features

draw spectrogram + features

Mapping approaches

- The voice conversion problem using the generative approach
- separate source signal and vocal tract features (LSF, MCEP or AE features)
- map source speaker vocal tract features x to target features y
  - $\hat{y} = \mathcal{F}(x)$
- where $\mathcal{F}$ is a transformation function:
  - Frame Selection (FS) [Dutoit08, Sundermann06]
  - Joint Density Gaussian Mixture Model (JDGMM) [Kain98]
  - Artificial Neural Networks (ANN) [Desai08]
  - Deep Neural Networks (DNN)

Frame Selection

- Overall idea similar to Unit-Selection Text to Speech Synthesis (TTS)
- A memory-based approach
- Keep all training data $[x, y]$
- At conversion time, find k-nearest entries to x_t, $C_{m_t^k}$
- Find the best output sequence $\hat{y} = [1, ..., \hat{y}_t]$ using Viterbi where it minimizes "target" and "concatenation" costs

$$Cost_{concatenation}(C^y_{m_t^k}, C^y_{m_{t+1}^k}) = d(C^y_{m_t^k}, C^y_{m_{t+1}^k}),$$
$$Cost_{target}(x_t, C^x_{m_t^k}) = d(x_t, C^x_{m_t^k})$$

- overall quality can suffer from highly incomplete coverage

JDGMM

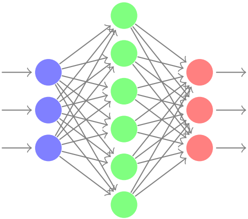
- have the potential to generalize to unseen data (unlike FS)
- Let $x = [x_1, ..., x_t]$ and $y = [y_1, ..., y_t]$ be D-dimensional source and target feature vectors
- Let $z_t = [z_1, ..., z_t]$ be the joint feature vector
- Each GMM performs a linear transformation of type $Ax_t + b$

$$\hat{y}_t = \sum_{m=1}^{M} P(m|x_t, \lambda_m^z) \cdot (W_m x_t + b_m)$$

- where $W_m = \Sigma_m^{yx} {\Sigma_m^{xx}}^{-1}$, $b_m = \mu_m^y + \Sigma_m^{yx} {\Sigma_m^{xx}}^{-1} \mu_m^x$
- and $P(m|x_t, \lambda_m^z)$ is the posterior probability of a mixture component m given the input vector x_t

Artificial Neural Networks

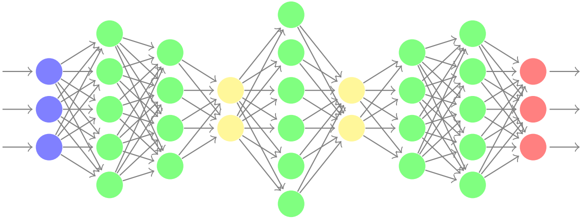
- we use a two-layered ANNs as a transformation function
- Each layer is $y = f(Wx + b)$, where x and y are the input and output of that layer, respectively
- Each layer applies a linear transformation using weights and biases (W and b) and then applies a non-linear activation function $f(.)$
- The parameters of each layer are trained using the back-propagation algorithm



Deep Neural Network

- The DNN consists of the trained ANN (on DAE-features) connecting the original hidden layers
- The hidden layers are duplicated to the top and the bottom of the of the DAE.
- Thus, the DNN is effectively pre-trained, taking its top and bottom weights from the DAE
- and the middle weights from the ANN.
- The network can now be further fine-tuned by back-propagation
- The AE is trained to be speaker independent

Deep Neural Network



Trajectory Generation

- The TG algorithm is used to smooth the feature sequence after the conversions [Toda07].

images/tg.png

Setup

- Training corpus:
  - 11 speakers
  - 7 chosen to train AE (1-2 recording from each, no need to be parallel)
  - 4 chosen for testing purposes
  - big training set: 70 harvard sentence from each of the 4 speakers
  - small training set: two randomly selected sentences from above
  - testing sentences: 20 sentences from each of the 4 speakers
- 4 speakers: two male (M1, M2) and two female (F1, F2)
- 4 conversion pairs
  - 2 cross-gender (M1→F1, F2→M2)
  - 2 intra-gender (M2→M1, F1→F2)

Model Parameters

- Feature order: MCEP: 24, LSF: 18 , DAE: 15

	MCEP	LSF	DMCEP	DLSF
JDGMM big (H)	32	32	64	32
ANN big (Q)	64	64	64	64
ANN small (H)	16	8	16	8
JDGMM small (Q)	8	2	8	4

Objective Scores

- mel-warped log spectral distance between (target and converted source)
- Average of all conversion over all 20 test sentences
- Large training set

feat/map	FS	GMM	ANN	DNN
LSF	8.14 (0.27)	8.00 (0.29)	7.95 (0.30)	NA
MCEP	**6.83 (0.31)**	6.90 (0.31)	6.85 (0.34)	**6.83 (0.31)**
DAE-LSF	8.68 (0.32)	8.61 (0.30)	8.63 (0.30)	-
DAE-MCEP	7.05 (0.28)	6.93 (0.29)	6.89 (0.29)	-

Objective Score

- Small training set (2 sentences)

feat/map	FS	GMM	ANN	DNN
LSF	8.81 (0.36)	9.14 (0.34)	8.23 (0.31)	NA
MCEP	7.60 (0.35)	8.31 (0.29)	7.58 (0.28)	**7.40 (0.30)**
DAE-LSF	9.31 (0.33)	9.56 (0.32)	9.03 (0.30)	-
DAE-MCEP	7.57 (0.31)	7.90 (0.29)	7.46 (0.26)	-

Future Work!

- Soon: Do a subjective experiment on Amazon Mechanical Turk (AMT)
  - Speaker Similarity
  - Speech Quality
- Include neighboring frames (11 frames?) + directly on spectrum (not mceps)
  - It requires a huge corpus
  - we can use speech recognition databases to train speaker independent AE

Thank you!

Questions?