Combining Speech and Speaker Recognition - A Joint Modeling Approach

Hang Su

Supervised by: Prof. N. Morgan, Dr. S. Wegmann

EECS, University of California, Berkeley, CA USA
International Computer Science Institute, Berkeley, CA USA

August 16, 2018

Table of contents

Table of contents

Joint modeling of speech and speaker

The brief idea

- Automatic speech recognition (ASR)
  - translate speech to text automatically
- Speaker recognition or speaker identification
  - identify speakers from characteristics of voice
- Combining speech and speaker recognition
  - capture speech and speaker characteristics together

Why speech / speaker recognition

Application of speech & speaker recognition

- Human-Computer Interface
- Automatic speech recognition
    - In-car system, smart home, speech search...
- Speaker recognition
    - Authentication, safety, personalization...

A problem

They are handled separately

- Different datasets / evaluations
- Different models / methods

But they are closely related to each other

- Take speech as input
- Similar features / models

A problem

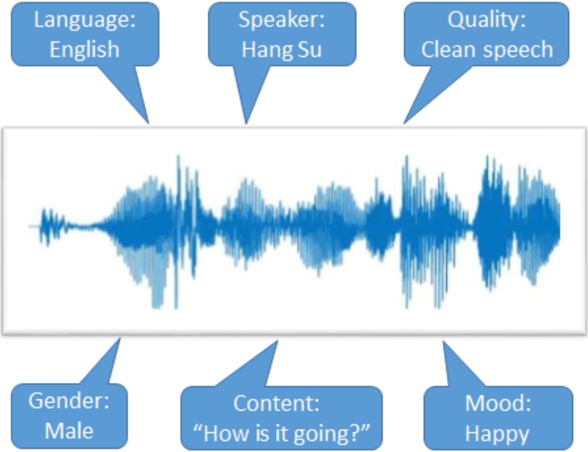
They are handled separately

- Different datasets / evaluations
- Different models / methods

But they are closely related to each other

- Take speech as input
- Similar features / models
- (Same group of researchers :)

An ideal AI agent for speech



An ideal AI agent for speech

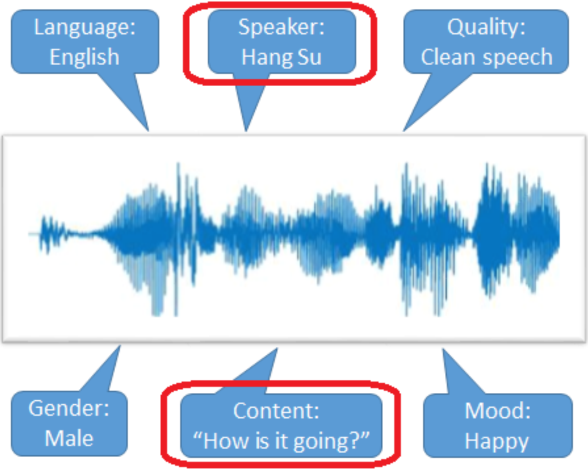


Table of contents

Table of contents

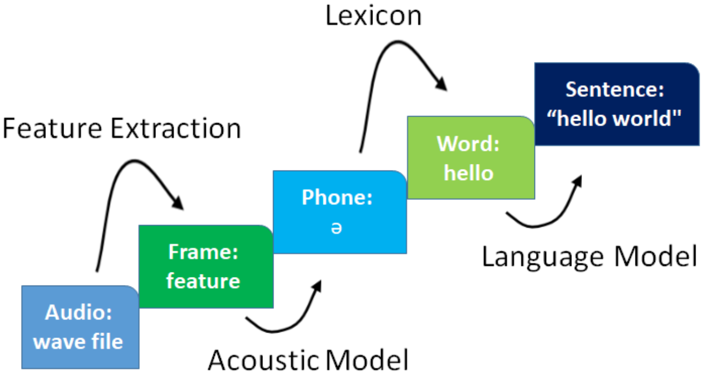
Automatic Speech Recognition (ASR)

Transcribe speech into texts

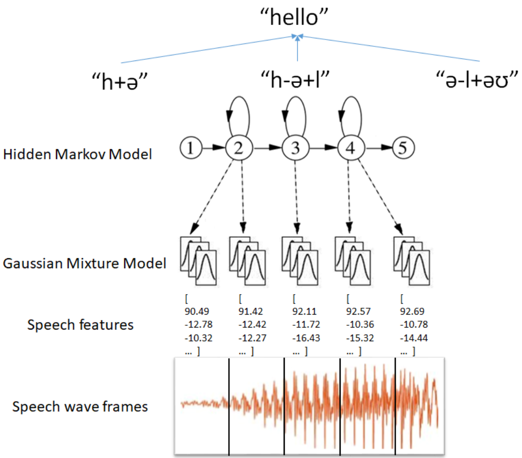
- Frame-by-frame approach (10 ~30 ms)
- Components*:
  - Feature extraction
  - Acoustic modeling (GMM-HMM)
  - Lexicon
  - Language modeling (LM)
- Or use end-to-end approach: discard HMM, optionally discard lexicon or language model

*For a traditional ASR system.

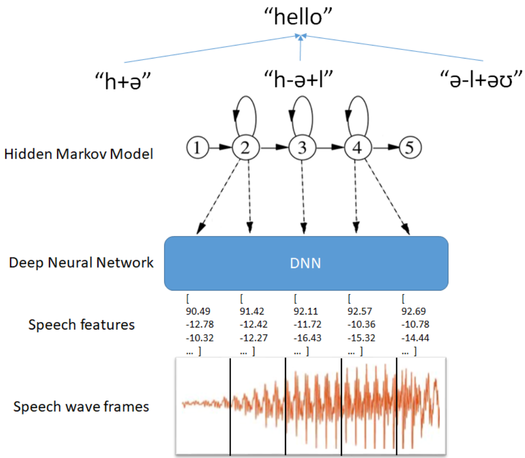
Traditional ASR pipeline



Gaussian Mixture Model - HMM[9, 3]



Deep Neural Network - HMM[1, 11]



Long-Short Term Memory - HMM [8]

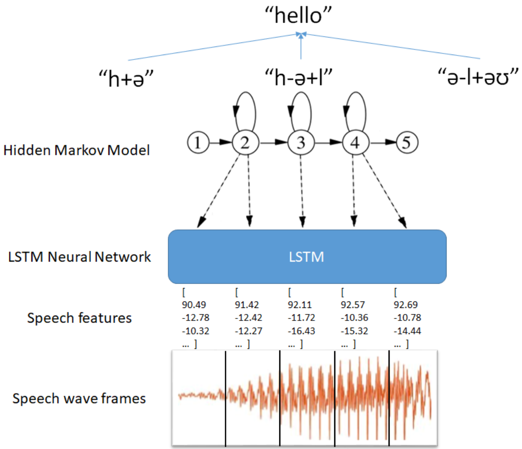


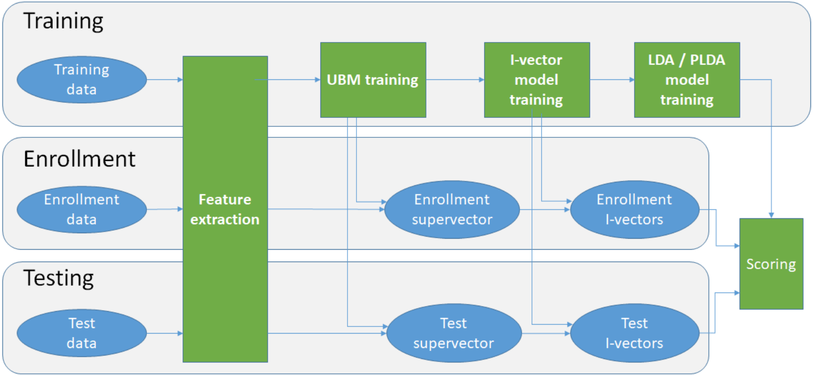
Table of contents

Speaker Recognition

Speaker Recognition: Identify speakers from speech

- Components:
  - Feature extraction
  - Acoustic modeling
  - Speaker modeling
  - Scoring
- Make utterance-level predictions

Text-independent speaker recognition



Factor analysis approach [2]

$$
\begin{aligned}
x_t &\sim \sum_{k}^{K} \pi_k \, \mathcal{N}(\mu_k + A_k z_i, \Sigma_k) \\
z_i &\sim \mathcal{N}(0, \mathrm{I}) \qquad \sum_{k=1}^{K} \pi_k = 1
\end{aligned}
\tag{1}
$$

- x_t is p-dim speech feature for frame t
- π_k is prior for mixture k
- z_i : a q-dim speaker specific latent factor (i.e. i-vector)
- A_k : a p-by-q projection matrix for mixture c
- μ_k and Σ_k are Gaussian parameters

Post-processing of i-vectors

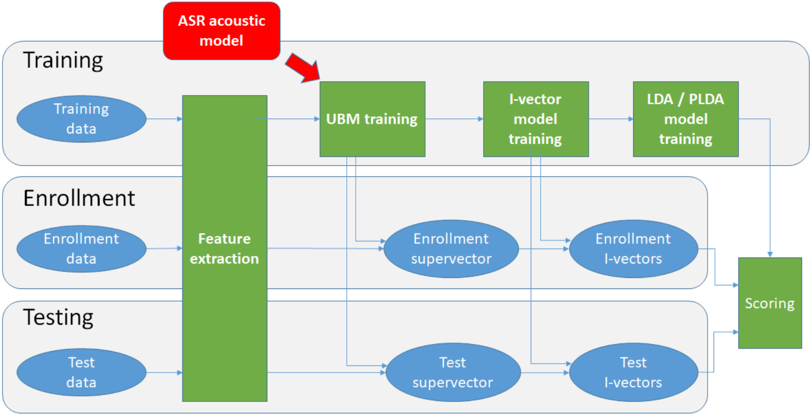
The factor-analysis model is an unsupervised model.
Supervised methods could be used to improve i-vectors.

- Linear Discriminant Analysis [6]
- Probabilistic Linear Discriminant Analysis [6, 5]

Table of contents

Table of contents

Speaker recognition using ASR



Speaker recognition using ASR cont.

- Substitute UBM with DNN model [7]
- Substitute UBM with Time-delay DNN [13]
- Use DNN initialized GMM acoustic model [13]
- Proposal: Use better DNN models for ASR †
    - Trained with raw MFCC feature
    - Trained with LDA transformed feature
    - Trained with LDA + fMLLR transformed feature
    - Trained with Minimum Phone Error (MPE) method

†Factor Analysis Based Speaker Verification Using ASR. Hang Su and Steven Wegmann. Interspeech 2016

Data description

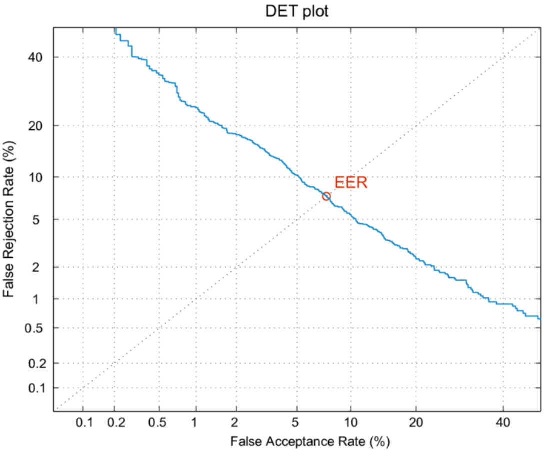
Speaker recognition evaluation (SRE) data set

- Training data (SRE 2004-2008)
  - 18,715 recordings from 3,009 speakers
  - 1,000+ hours of data, 360,000,000 frame samples
- Test data (SRE 2010)
  - 387,112 trials (98% non-target)
  - 11,983 enrollment speakers, 767 test speakers
  - 2 ~3 mins per speaker

ASR data set

- Training data (Switchboard)
- Testing data (Eval2000)

Metric – DET curve and EER



Metric – Word Error Rate (WER)

$$WER = \frac{S + D + I}{R} \tag{2}$$

- S : number of substitutions
- D : number of deletions
- I : number of insertions
- R : number of words in references

Experimental results

	Eval2000 WER	SRE2010 EER
UBM	–	6.31
DNN-MFCC	19.4	6.39
+ LDA + MLLT	16.3	4.84
+ fMLLR*	14.9	4.55
+ MPE*	13.5	4.38

Table 1: EER for speaker recognition systems in different settings

*ASR decoding needed

Experimental results

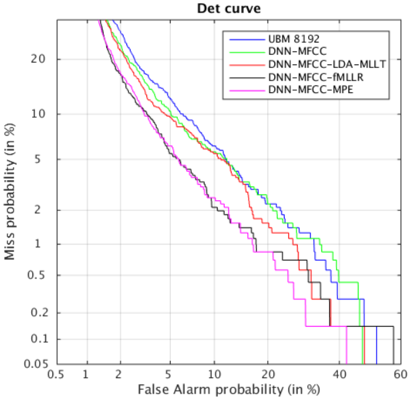


Figure 1: DET curve for systems in different settings

Table of contents

Speaker Adaptation

How to handle speaker-specific characteristics during recognition?

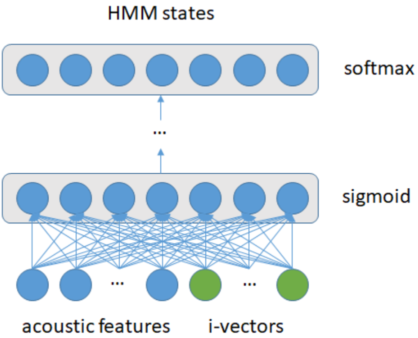
- Adapt speaker-independent systems to different speakers (model-space)
- Normalize speech features to compensate speaker characteristics (feature-space)

Speaker adaptation for DNN systems

Existing methods:

- Feature-space transformations (fMLLR) [4]
- Model-space transformations [15]
- Adapting model parameters via regularization [16]
- Learning hidden unit contributions (LHUC) [14]

Speaker adaptation using i-vectors[10]



$$h = W_a x + W_s z$$

Speaker adaptation using i-vectors

Benefits of using i-vectors

- Does not require model re-training or ASR decoding
- Single DNN model for all speakers

Potential drawback:

- Tend to overfit

Problem of speaker adaptation using i-vector

I-vectors are extracted for every recordings

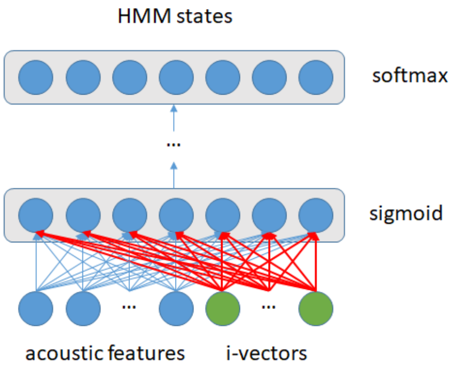
- Frames 100 million, 4,800 recordings
- Acoustic feature dim ~440, i-vector dim 100~400
- Better objective on training data does not translate into WER improvement
- Overfitting occurs

Treatment for overfitting

Mitigate overfitting by

- Reducing i-vector dimension[10]
- Using utterance-based i-vectors[12]
- Extract i-vectors using sliding window (in Kaldi)
- L2 regularization back to baseline DNN[12]

Regularization on i-vector sub-nnetwork



$$L_{re} = L_{ce} + \beta \|w_{ivec}\|^2$$

Data description

Switchboard data set

- Clean telephone speech, English
- ~300 hours transcribed data (~108,000,000 samples)
- ~4,800 recordings

Eval2000 hub5 test set

- Switchboard portion + CallHome (family members)
- 40 + 40 speakers
- 2 hours + 1.6 hours

Metric – Word Error Rate (WER)

$$WER = \frac{S + D + I}{R} \tag{3}$$

- S : number of substitutions
- D : number of deletions
- I : number of insertions
- R : number of words in references

Experimental results

feature	MFCC		+fMLLR	
data	Swbd	Callhome	Swbd	Callhome
acoustic feature	16.0	28.5	14.9	25.6
+ i-vector	15.2	27.1	14.4	25.7
+ regularization	**14.6**	**26.3**	**14.3**	**24.9**

Table 2: WER on i-vector adaptation using regularization

Conclusion

A brief summarization:

- Speech and speaker recognition are two tasks that are closely related
- Speaker information can be used to improve speech recognition performance
- Acoustic models trained for ASR can be used to assist speaker recognition

Table of contents

Table of contents

Existing Tools for Speech

Kaldi:

- Popular speech recognition tool
- Supports GMM, HMM, DNN, LSTM ....
- State-of-the-art recipes

Tensorflow (TF)

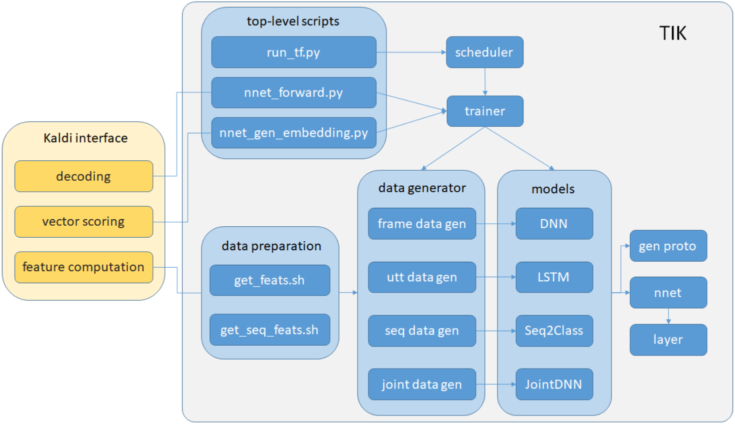
- Flexible deep learning research framework
- Tensorflow Lite: esay to deploy on embedded devices
- Tensor Processing Unit (TPU)

TIK

Bridge the gap between Tensorflow and Kaldi

- It supports acoustic modeling using Tensorflow
- It integrates with Kaldi decoder through a pipe
- It covers both speech and speaker recognition tasks

System Design of TIK



ASR performance using TIK

	Swbd	CallHome	All
Kaldi GMM	21.4	34.8	28.2
Kaldi DNN	14.9	25.6	20.3
TIK DNN	14.5	25.5	20.0
TIK BLSTM	13.6	24.3	19.0

Table 3: WER of ASR systems trained with Kaldi and TIK (Eval2000 test set)

Speaker recognition performance using TIK

	Cosine	LDA	PLDA
Kaldi UBM	6.91	3.36	2.51
Kaldi DNN	4.00	1.83	**1.27**
TIK DNN	4.53	2.00	**1.27**

Table 4: EER of speaker recognition systems trained Kaldi and TIK (SRE2010 test set)

Table of contents

X-vector approach

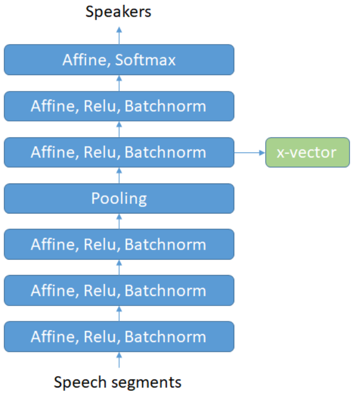


Figure 2: Structure of x-vector approach for speaker recognition

JointDNN model

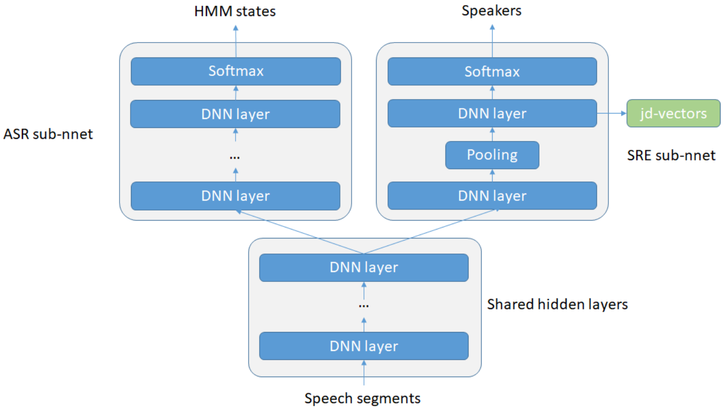


Figure 3: Structure of JointDNN model

Loss function

$$L(\theta) = -\sum_{s=1}^{S}\sum_{t=1}^{s_T} h_{s,t} \log P(h_{s,t}|o_{s,t}) - \beta \sum_{s=1}^{S} x_s \log P(x_s|o_s) \quad (4)$$

- Interpolation of two cross-entropy losses
- β is the interpolation weight
- $h_{s,t}$ denotes the HMM state for frame t of segment s
- $o_{s,t}$ is the observed feature vector
- x_s is the correct speaker
- o_s is speech features for segment s

Data description

Training data

- Switchboard data set
- ~300 hours transcribed data (~108,000,000 samples)
- ~520 speakers

Testing data

- Eval2000 hub5 test set for speech recognition
- SRE2010 test set for speaker recognition

Performance of speaker recognition

	EER
Baseline i-vector	4.85
Kaldi x-vector	8.94
TIK x-vector	8.81
TIK jd-vector (beta0.01)	4.75

Table 5: EER of JointDNN model for speaker recognition (SRE2010 test set)

Performance of speaker recognition

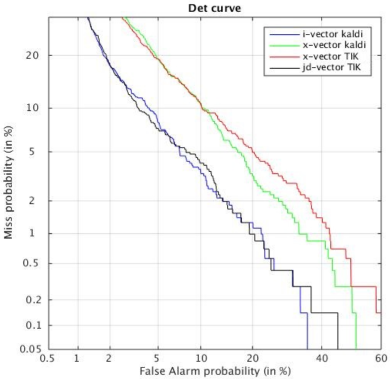


Figure 4: DET curve of JointDNN model for speaker recognition (SRE2010 test set)

Performance of speech recognition

	Swbd	Callhome	All
Baseline DNN	16.1	28.4	22.3
JointDNN (beta 0.01)	16.8	29.0	22.9

Table 6: WER of JointDNN model for speech recognition

Adjusting Interpolation Weight β

	Development (%)		Evaluation (%)	
Beta	ASR acc	Speaker acc	SRE EER	Swbd WER
0.1	39.07	97.22	5.10	16.7
0.01	39.20	94.10	4.75	16.8
0.001	38.60	85.36	9.19	17.2
0.0001	38.59	41.95	13.25	17.0

Table 7: EER of JointDNN model with different β

Conclusion

Summary of JointDNN model

- JointDNN can be used for ASR and SRE simultaneously
- ASR part helps guide speaker recognition sub-network
- Effective in using a limited amount of training data
- Uses less memory compared to i-vector approach (better for embeded device)

Table of contents

Conclusion and Future Work

Conclusion of the talk

- Speech and speaker recognition are beneficial to each other
- A joint model helps exploit both speech and speaker information
- Effective in using limited amount of training data

Future work

Future work on joint modeling

- Use a larger data set or data augmentation techniques
- Introduce recurrent structures into joint model
- End-to-end approaches for joint modeling
- Towards an all-around speech AI agent

The End

Reference I

Herve A. Bourlard and Nelson Morgan.
Connectionist Speech Recognition: A Hybrid Approach.
Kluwer Academic Publishers, Norwell, MA, USA, 1993.

Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet.
Front-end factor analysis for speaker verification.
IEEE Transactions on Audio, Speech, and Language Processing, 19(4):788–798, 2011.

Reference II

Mark Gales and Steve Young.
The application of hidden markov models in speech recognition.
Foundations and Trends® in Signal Processing, 1(3):195–304, 2008.

Mark JF Gales.
Maximum likelihood linear transformations for hmm-based speech recognition.
Computer speech & language, 12(2):75–98, 1998.

Reference III

Daniel Garcia-Romero and Carol Y Espy-Wilson.
Analysis of i-vector length normalization in speaker recognition systems.
In *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

Patrick Kenny, Themos Stafylakis, Pierre Ouellet, Md Jahangir Alam, and Pierre Dumouchel.
Plda for speaker verification with utterances of arbitrary duration.

Reference IV

In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 7649–7653. IEEE, 2013.

Yun Lei, Scheffer Nicolas, Luciana Ferrer, and Mitchell McLaren.
A novel scheme for speaker recognition using a phonetically-aware deep neural network.
In *ICASSP*. IEEE, 2014.

Reference V

Abdel-rahman Mohamed, Frank Seide, Dong Yu, Jasha Droppo, Andreas Stoicke, Geoffrey Zweig, and Gerald Penn.
Deep bi-directional recurrent networks over spectral windows.
In *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pages 78–83. IEEE, 2015.

Lawrence R Rabiner.
A tutorial on hidden markov models and selected applications in speech recognition.
Proceedings of the IEEE, 77(2):257–286, 1989.

Reference VI

- George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny.
  Speaker adaptation of neural network acoustic models using i-vectors.
  In *ASRU*, pages 55–59, 2013.

- Frank Seide, Gang Li, and Dong Yu.
  Conversational speech transcription using context-dependent deep neural networks.
  In *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

Reference VII

Andrew Senior and Ignacio Lopez-Moreno.
Improving dnn speaker independence with i-vector inputs.
In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 225–229. IEEE, 2014.

David Snyder, Daniel Garcia-Romero, and Daniel Povey.
Time delay deep neural network-based universal background models for speaker recognition.
In *ASRU*. IEEE, 2015.

Reference VIII

Pawel Swietojanski and Steve Renals.
Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models.
In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 171–176. IEEE, 2014.

Kaisheng Yao, Dong Yu, Frank Seide, Hang Su, Li Deng, and Yifan Gong.
Adaptation of context-dependent deep neural networks for automatic speech recognition.
In *Spoken Language Technology Workshop (SLT), 2012 IEEE*, pages 366–369. IEEE, 2012.

Reference IX

Dong Yu, Kaisheng Yao, Hang Su, Gang Li, and Frank Seide.
Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition.
In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 7893–7897. IEEE, 2013.