Speech Recognition with Deep Learning Methods

Ming-Han Yang
楊明翰

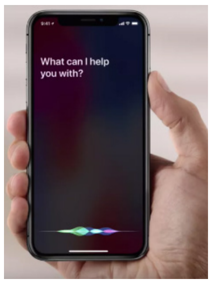
台科大
TAIWAN TECH

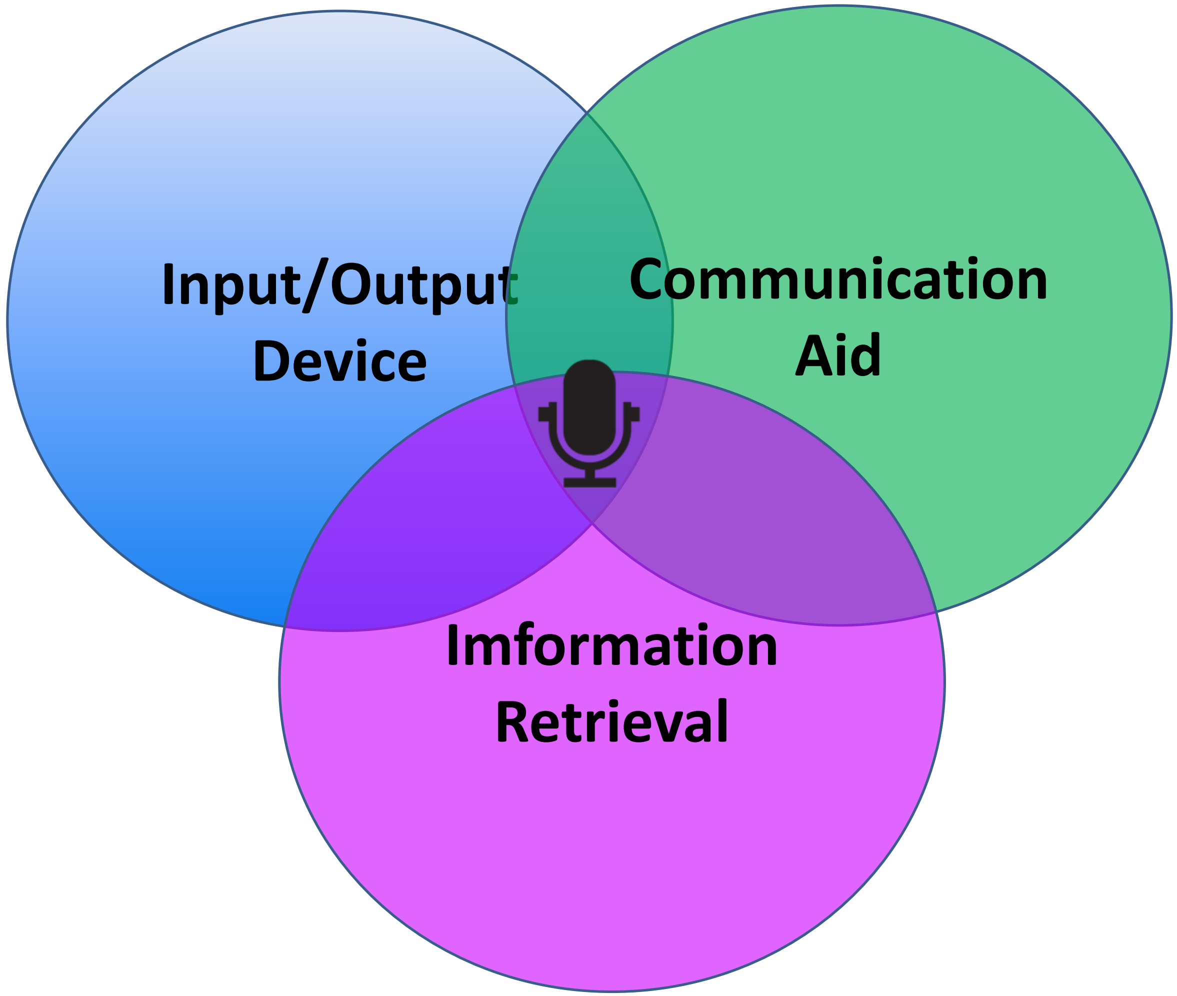
DELTA

Challenges in Automatic Speech Recognition

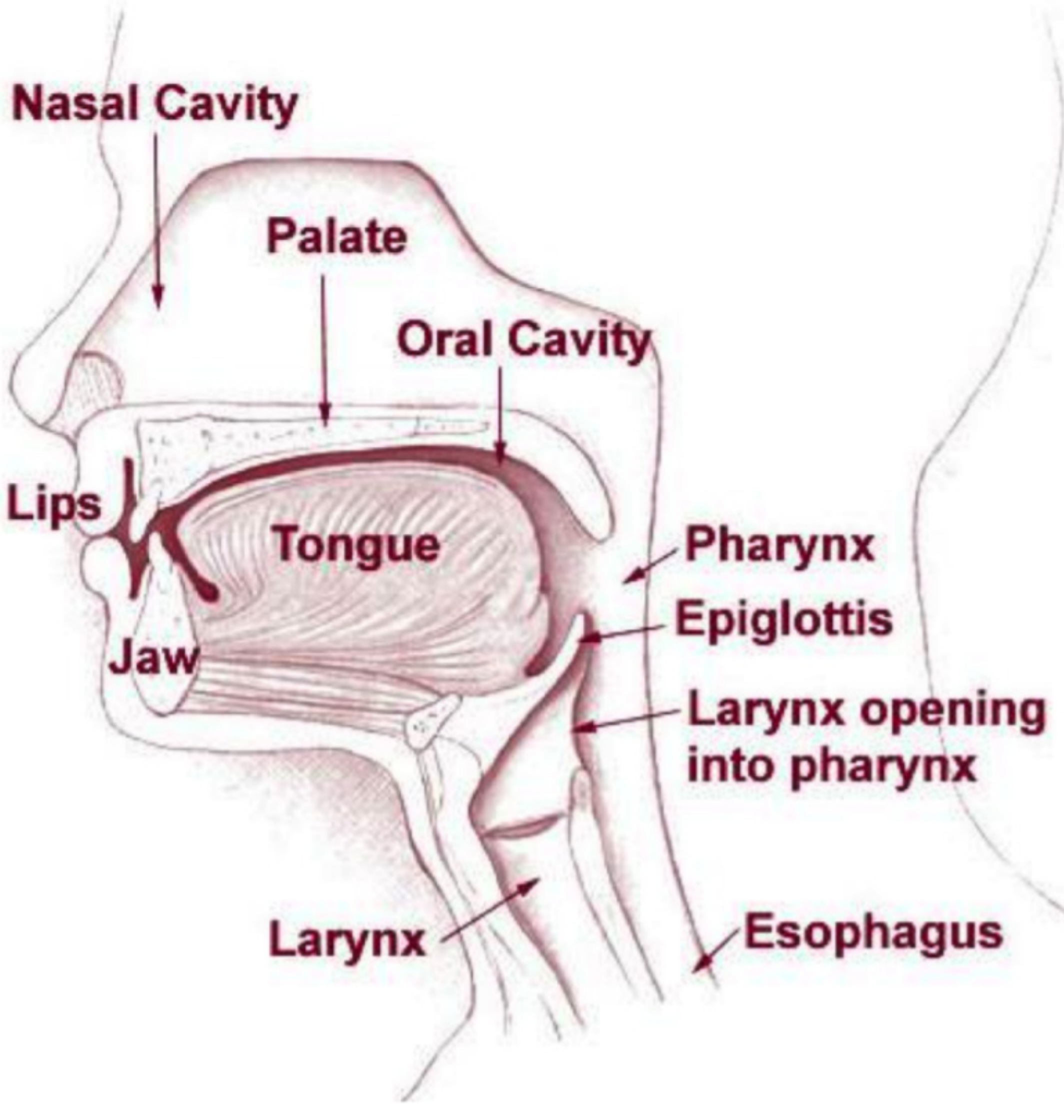
https://www.youtube.com/watch?v=wV9At3aB7mU

Speech Application Areas

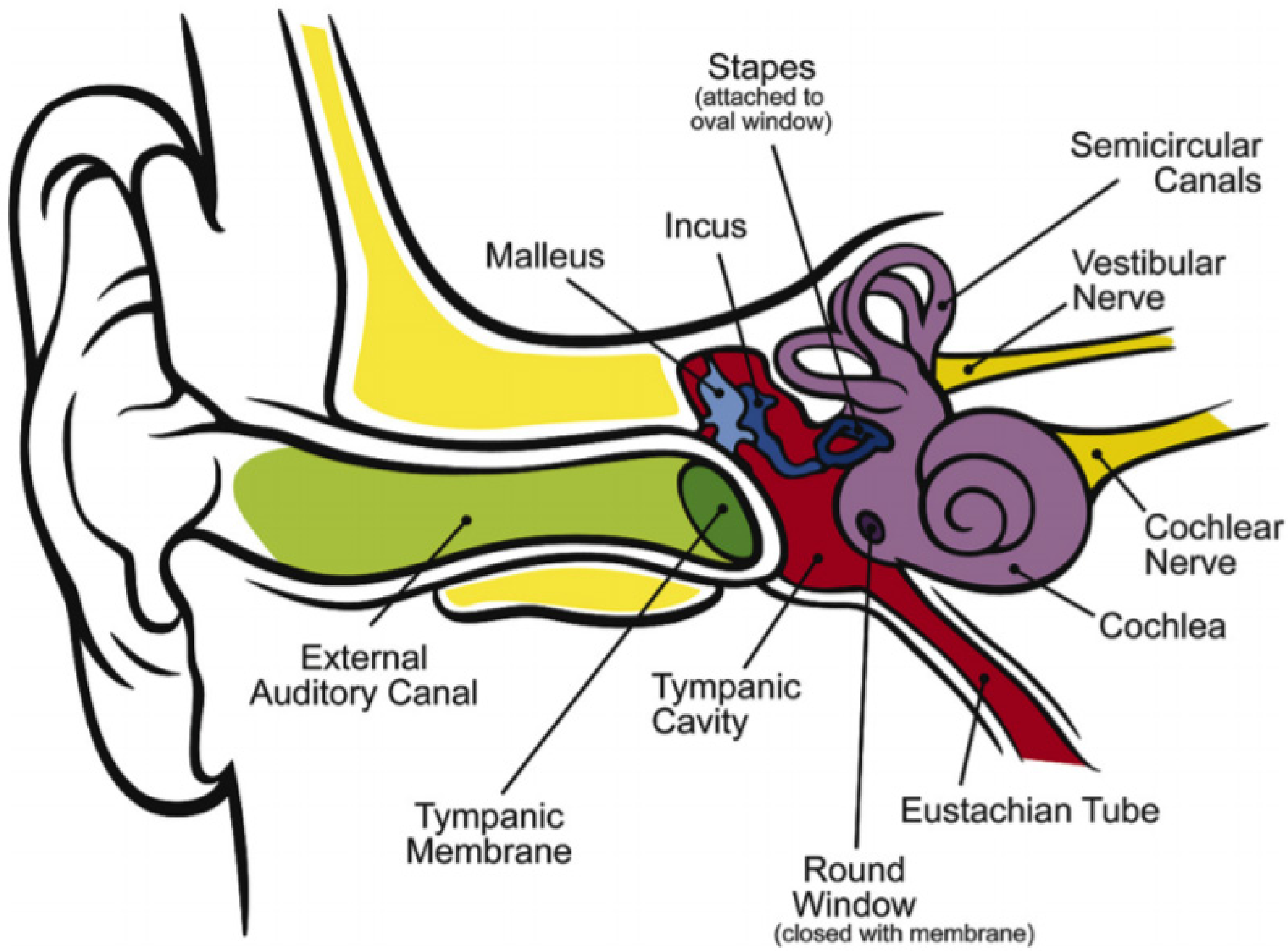




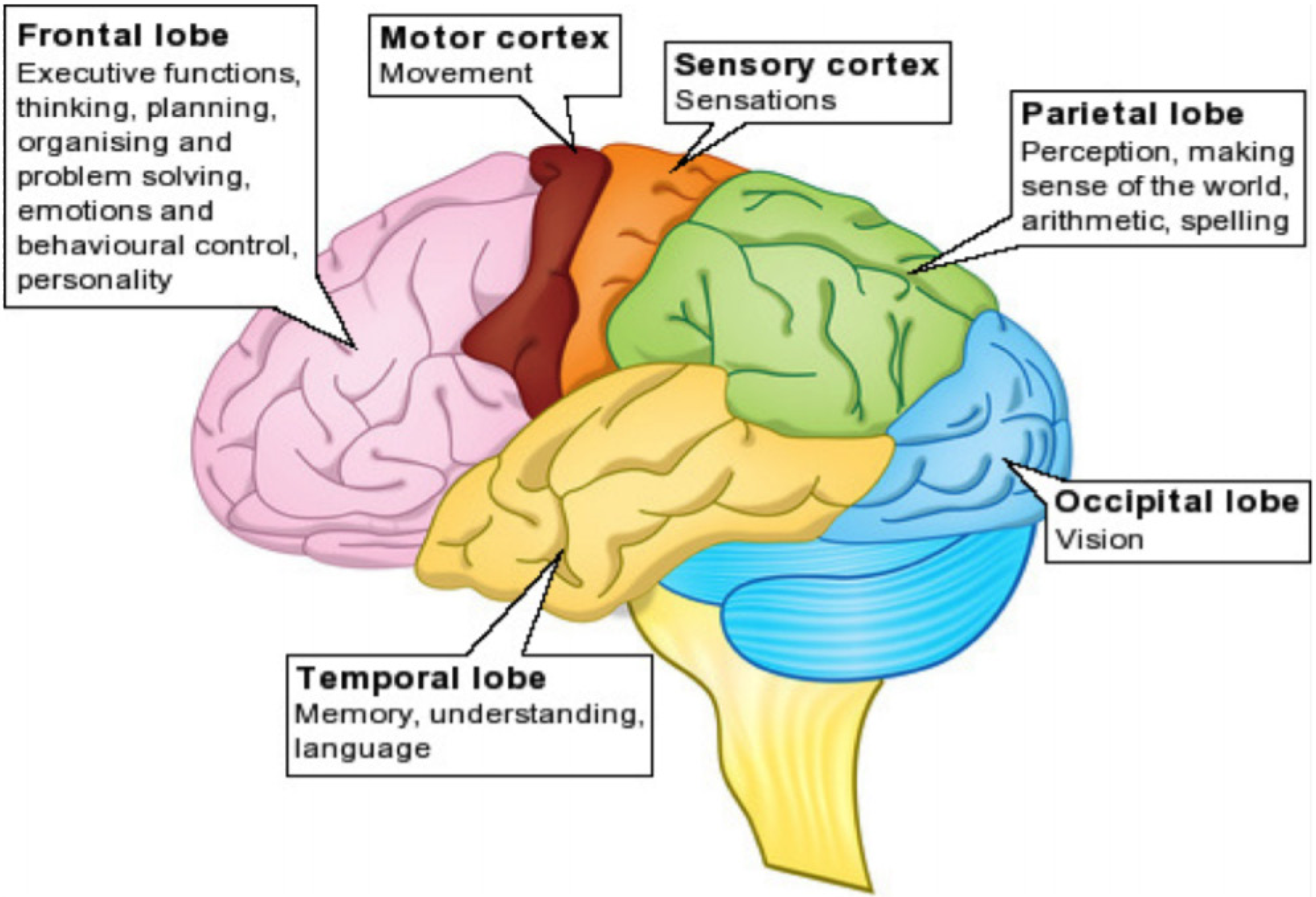
Speech Production (Synthesis)



Speech Perception (Recognition)



Speech Understanding



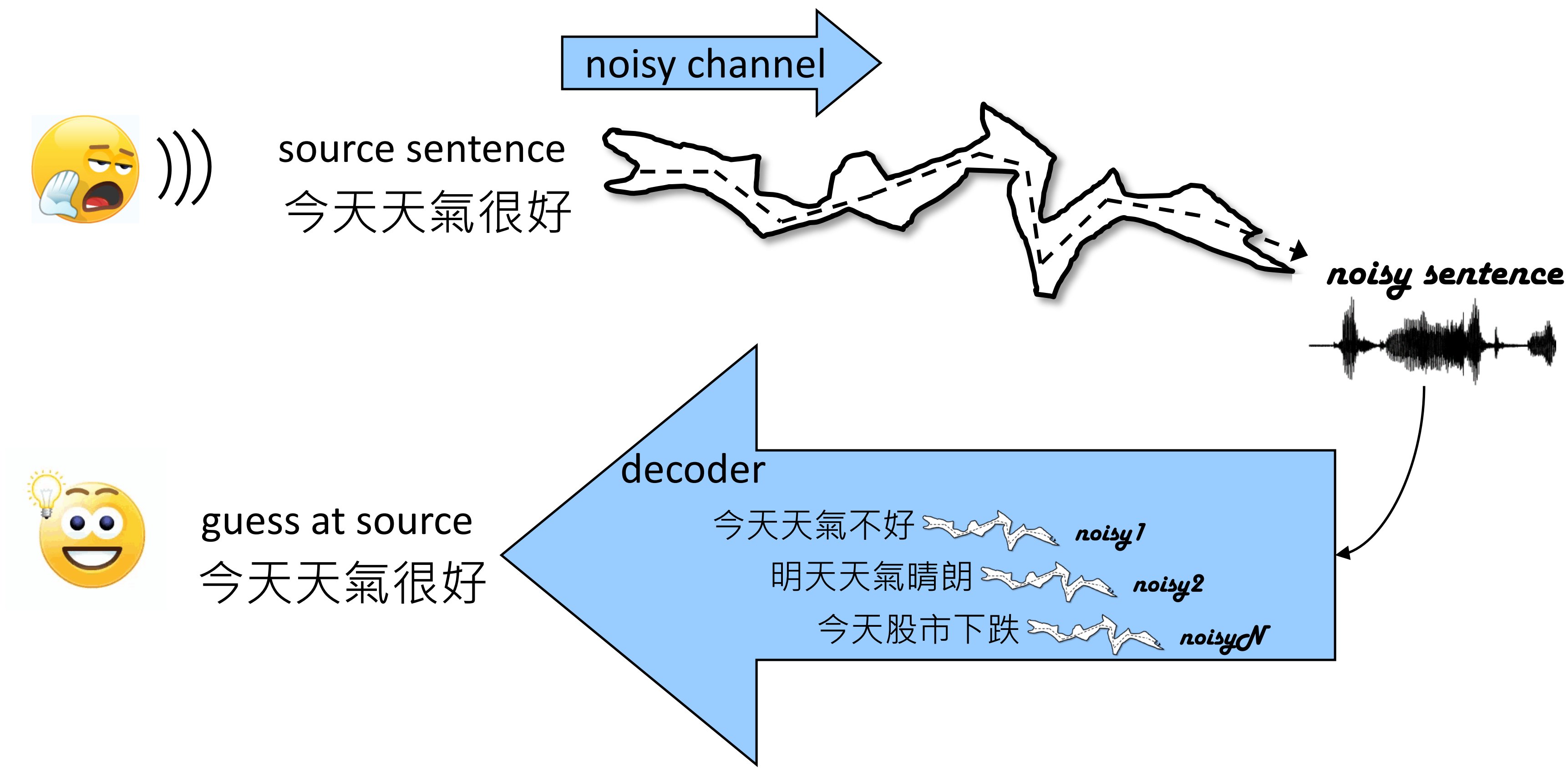
Outline

- Traditional Speech Recognition
- How to use Deep Learning in acoustic modeling?
- Why Deep Learning?
- Speaker Adaptation
- Multi-task Deep Learning
- New acoustic features
- Convolutional Neural Network (CNN)
- Applications in Acoustic Signal Processing

Traditional Speech Recognition

Overview

- Search through space of all possible sentences.
- Pick the one that is most probable given the waveform.



Speech Recognition (Traditional)

Wavform

今 天 天 氣 很 好

Words

Speech Recognition (Traditional)

10 ms

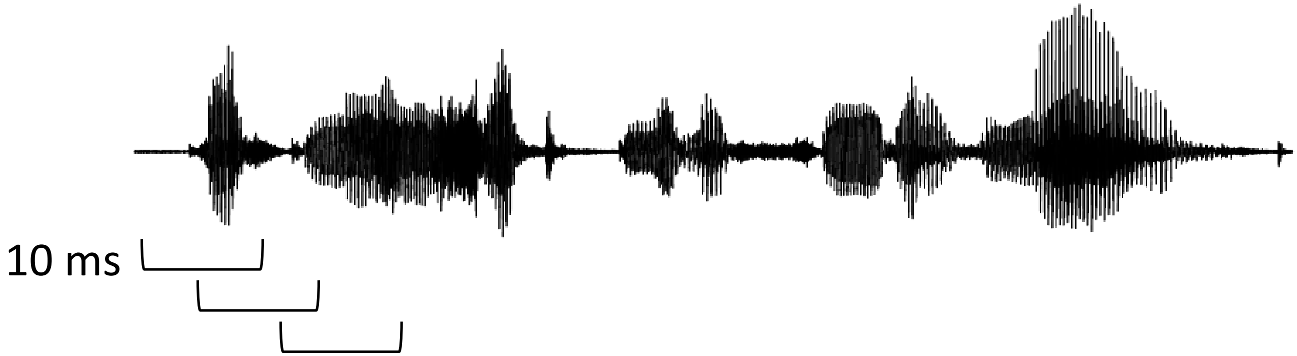
Wavform

Features

今 天 天 氣 很 好

Words

Speech Recognition (Traditional)



Wavform

Features

/t/ /i/ /a/ /n/

Phones

今 天 天 氣 很 好

Words

Speech Recognition (Traditional)

10 ms

Wavform

Features

/t/-/i/+/a/

Context–Dependent Phones

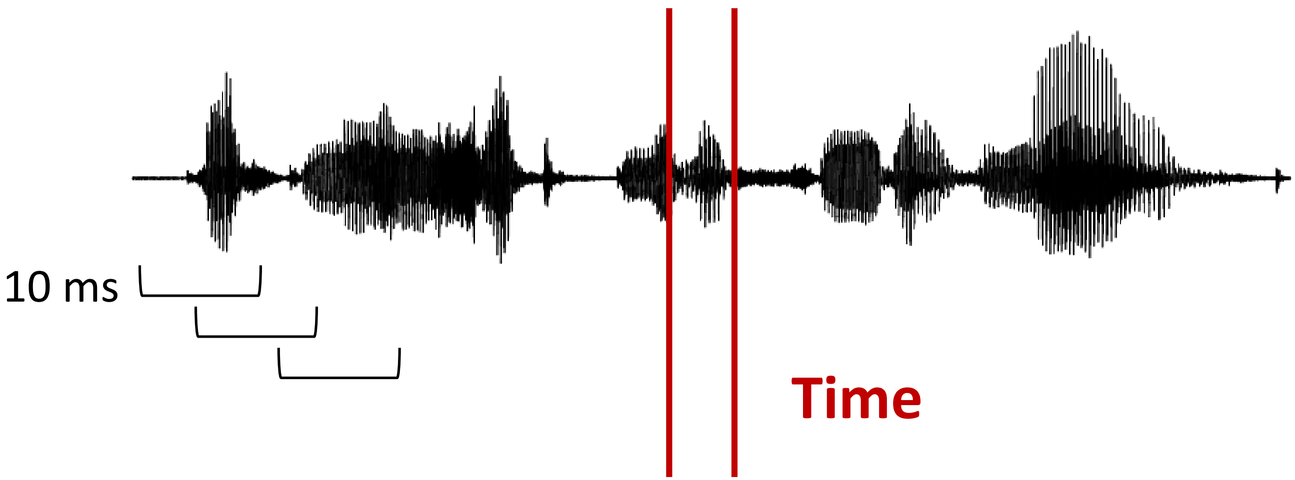
/t/ /i/ /a/ /n/

Phones

今 天 天 氣 很 好

Words

Speech Recognition (Traditional)



Wavform

Features

/t/-/i/+/a/ — **Context–Dependent Phones**

/t/ /i/ /a/ /n/ — **Phones**

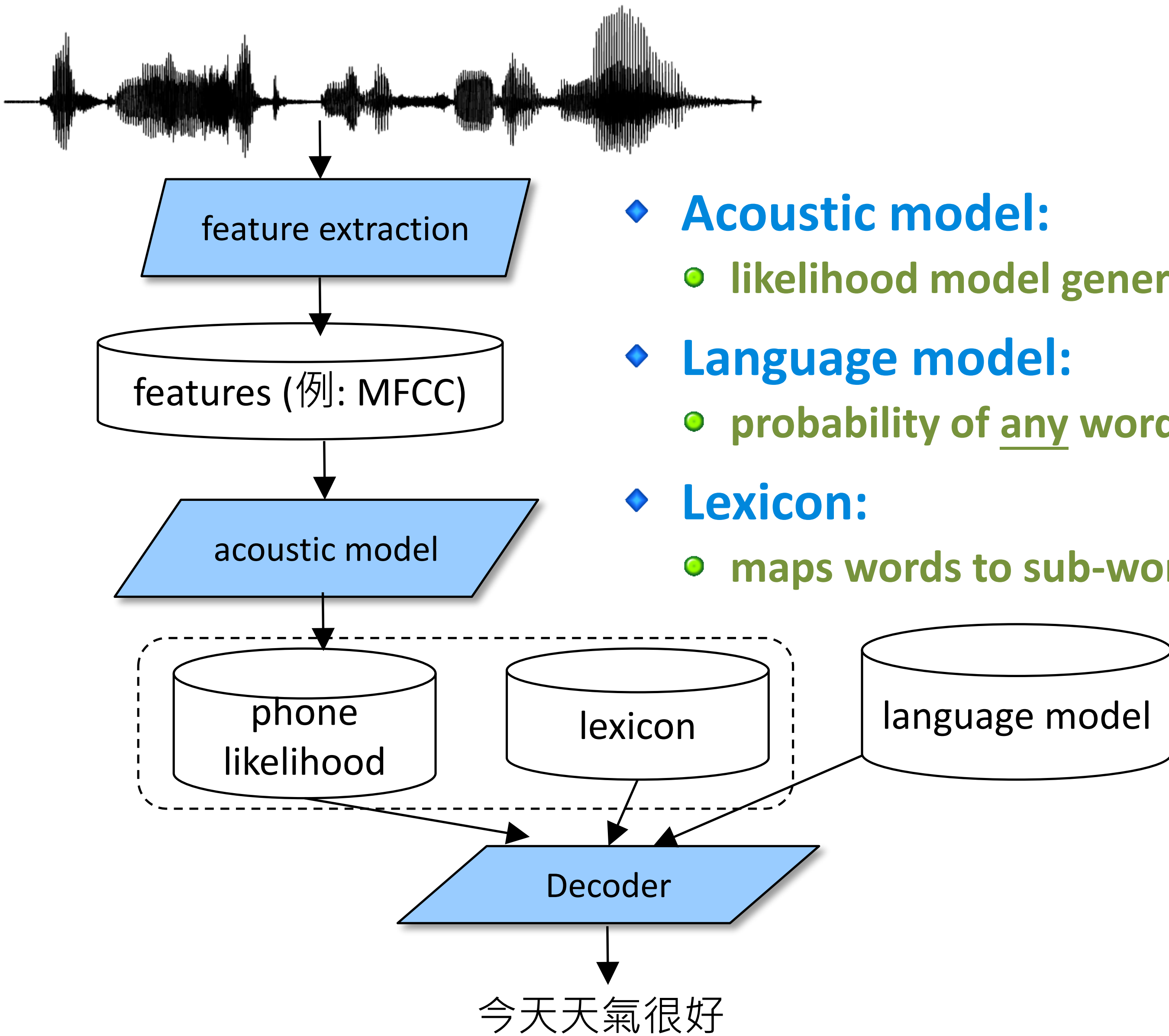
今 天 天 氣 很 好 — **Words**

Sequence-to-Sequence Modelling

- **Sequence-to-sequence modelling central to speech/language:**
  - machine translation:
    - word sequence (discrete) → word sequence (discrete)
      - 你好 → Hello
  - speech synthesis:
    - word sequence (discrete) → waveform (continuous)
      - 你好 → 
  - speech recognition:
    - waveform (continuous) → word sequence (discrete)
      -  → 你好

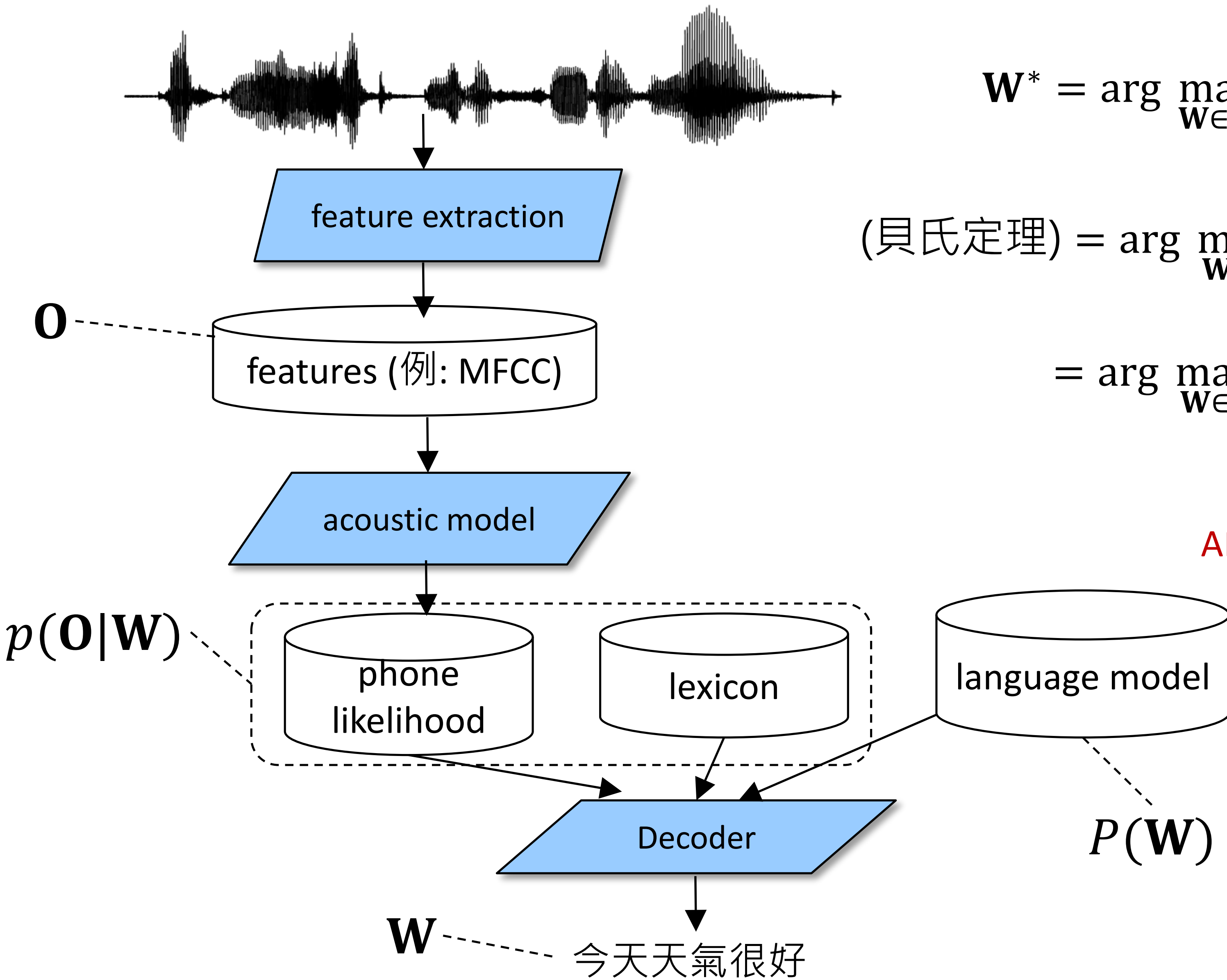
- **The sequence lengths on either side can differ**
  - waveform sampled at 10ms/5ms frame-rate: T-length
  - word/token sequences: L-length
  - T遠大於L

Speech Recognition Architecture



- **Acoustic model:**
  - likelihood model generating observed features
- **Language model:**
  - probability of <u>any</u> word sequence
- **Lexicon:**
  - maps words to sub-word units (phones)

Speech Recognition Architecture



$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in \boldsymbol{L}} P(\mathbf{W}|\mathbf{O})$$

$$(\text{貝氏定理}) = \arg\max_{\mathbf{W}\in \boldsymbol{L}} \frac{p(\mathbf{O}|\mathbf{W})P(\mathbf{W})}{p(\mathbf{O})}$$

$$= \arg\max_{\mathbf{W}\in \boldsymbol{L}} p(\mathbf{O}|\mathbf{W})P(\mathbf{W})$$

AM (likelihood)

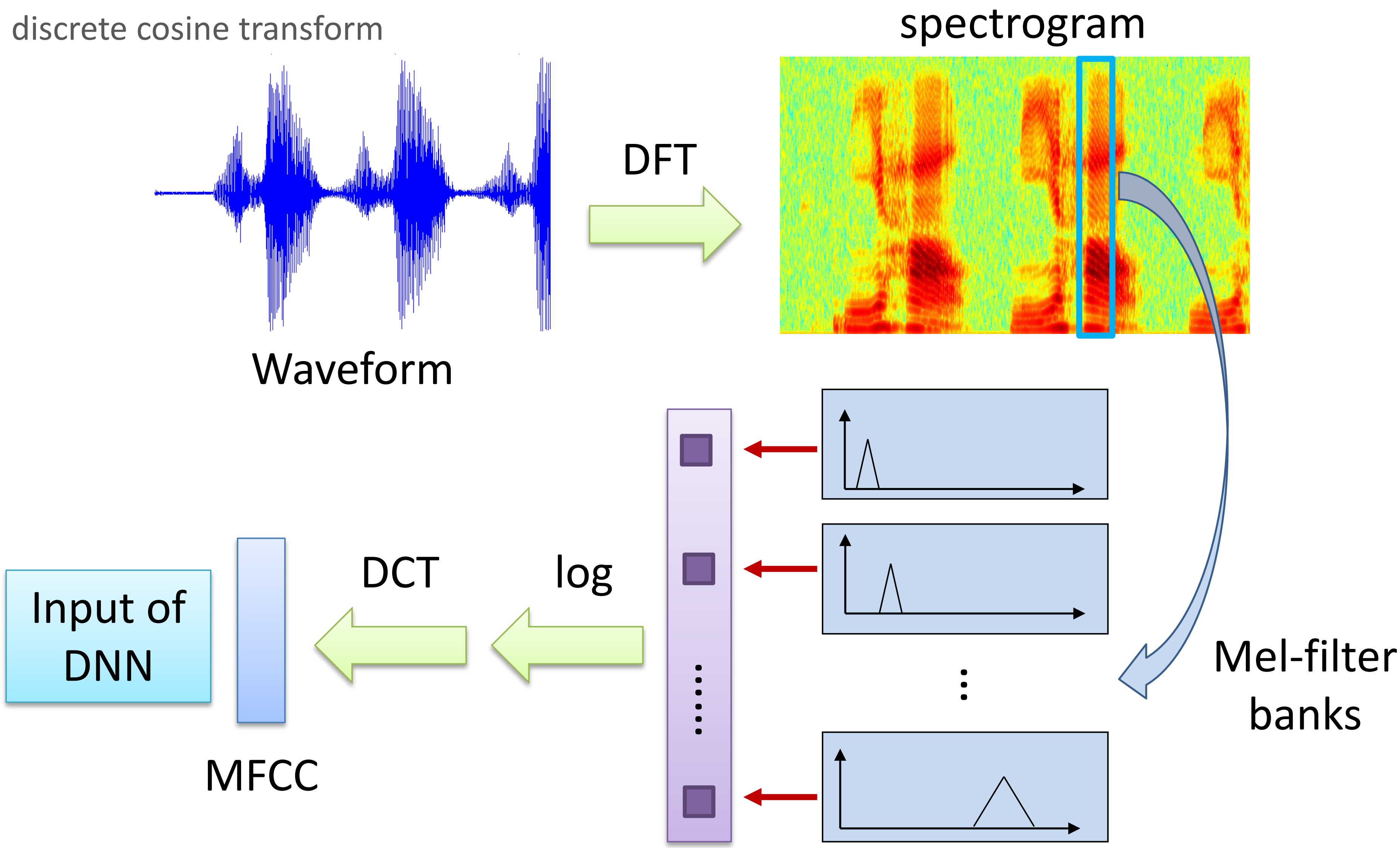
LM (prior)

Input Feature - MFCC

※Mel-scale Frequency Cepstral Coefficients

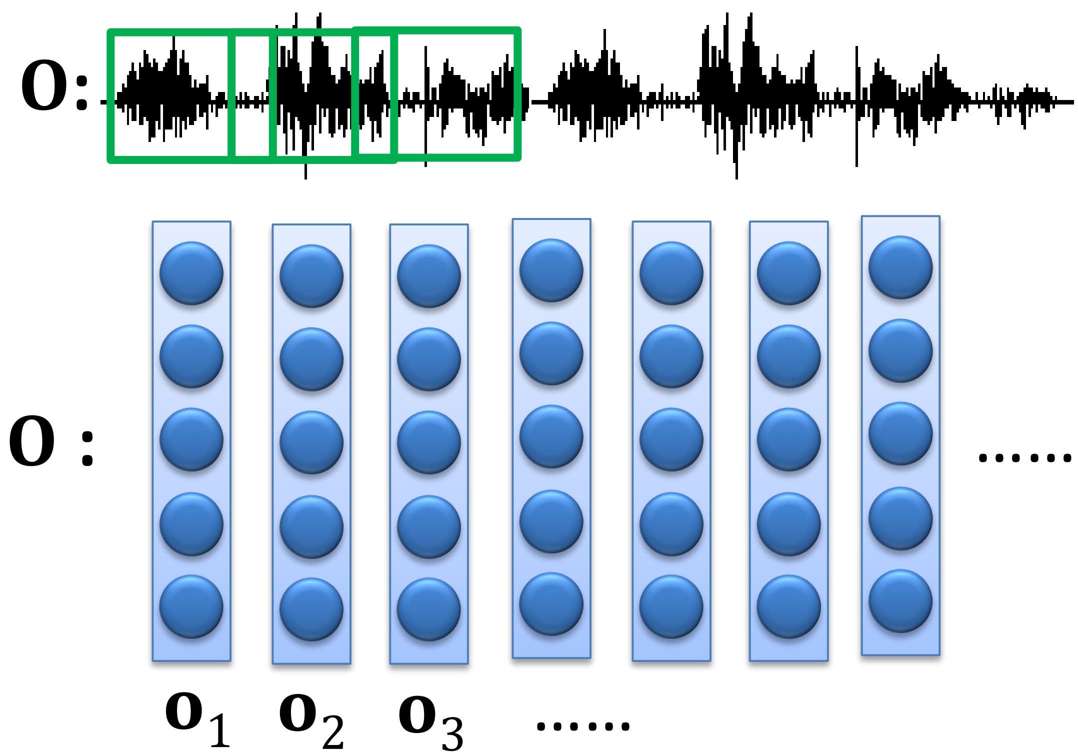
DFT: discrete fourier transform

DCT: discrete cosine transform



Input Feature

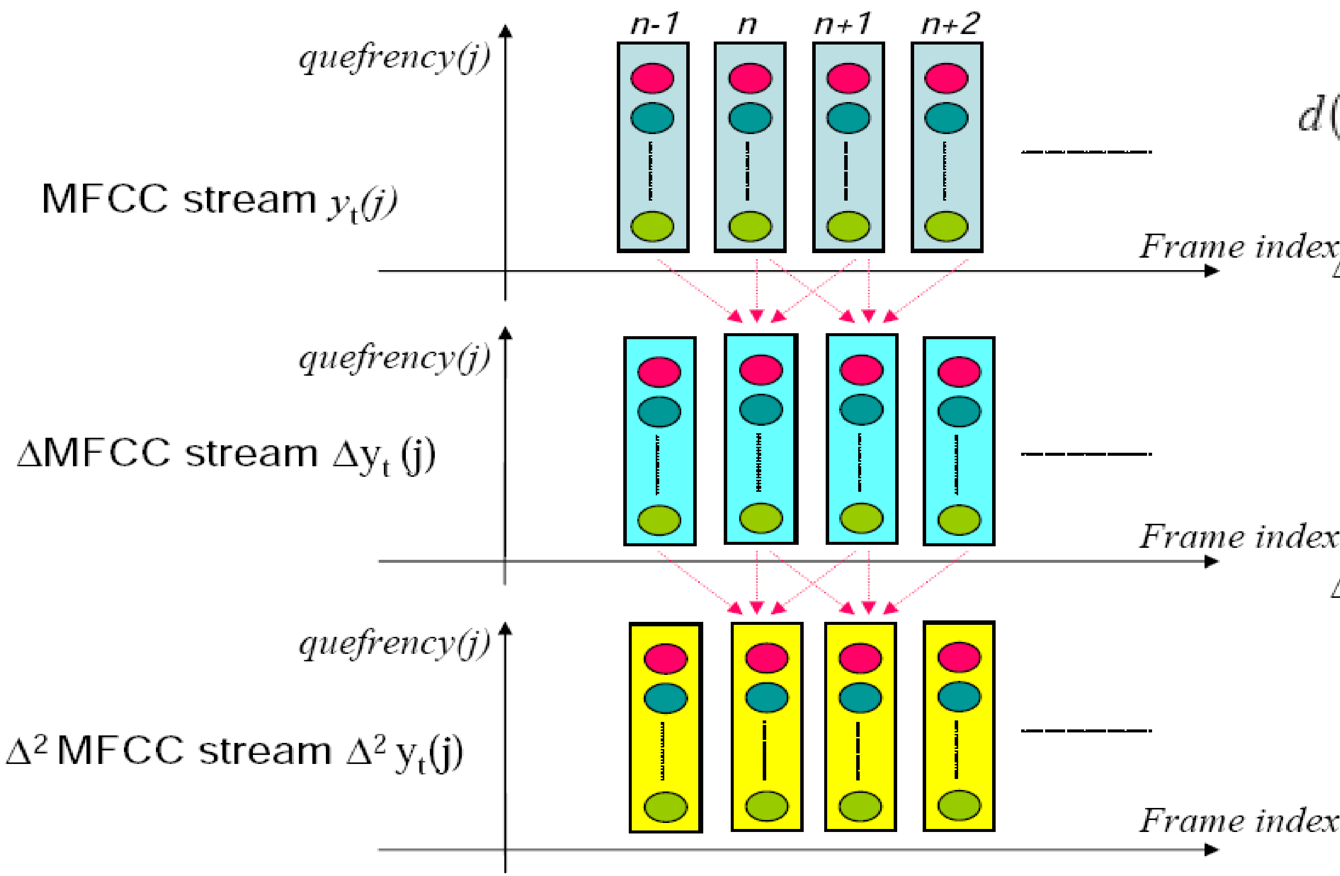
- **Audio is represented by a vector sequence**
  - 聲音 → 音框(frame): $\mathbf{o}_1, \mathbf{o}_2, \mathbf{o}_3$, ...



每一個frame是
13 dim MFCC

Input Feature - Δ and ΔΔ

- **Derivative: in order to obtain temporal information**
  - 可以看成是MFCC的速度與加速度的資訊
  - 所以每一個frame會變成 39 dim MFCC



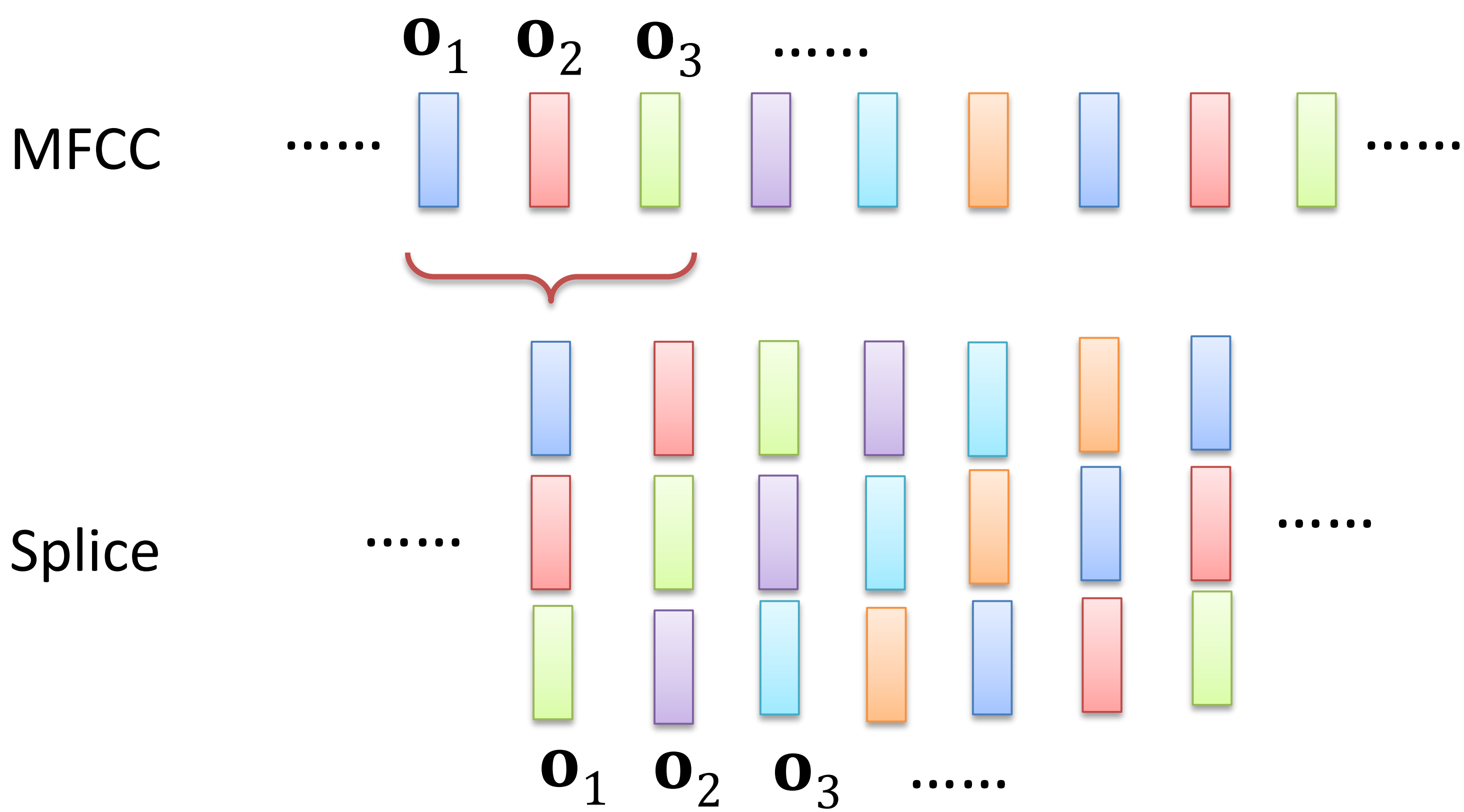
$$d(t) = \frac{c(t+1) - c(t-1)}{2}$$

$$\Delta y_t(j) = \frac{\sum_{m=-p}^{p} m \bullet y_{t-m}(j)}{\sum_{m=-p}^{p} m^2}$$

$$\Delta^2 y_t(j) = \frac{\sum_{m=-p}^{p} m \bullet \Delta y_{t-m}(j)}{\sum_{m=-p}^{p} m^2}$$

Input Feature - Splice

- To consider some temporal information
  - 把前後的frame跟目前的frame串起來，變成一個新的vector



Phoneme & Lexicon

- **Phoneme: basic unit**
  - Lexicon: maps words to sub-word units
- Each word corresponds to a sequence of phonemes

Lexicon

黑面琵鷺	h ei1 m ian4 p i2 l u4
黑馬	h ei1 m a3
黑體	h ei1 t i3
黑髮	h ei1 f a3
黑鯛	h ei1 d iao1
黑鯨	h ei1 j ing1
context	k aa n t eh k s t
everyday	eh v r iy d ey
include	ih n k l uw d

Word what do you think

Different words can correspond to the same phonemes

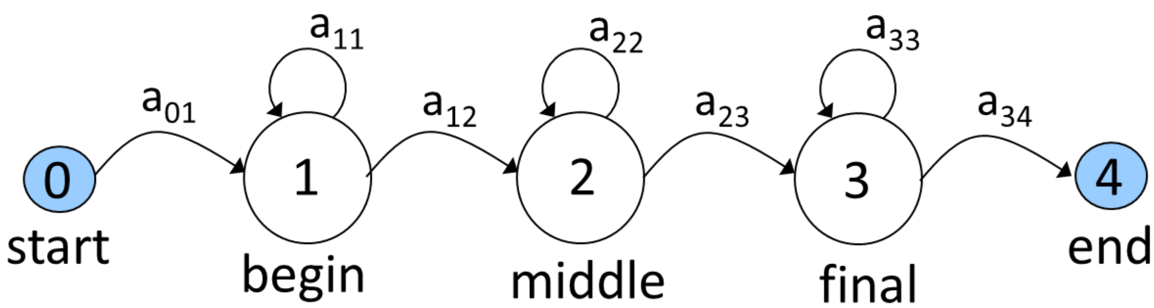
Lexicon hh w aa t d uw y uw th ih ng k

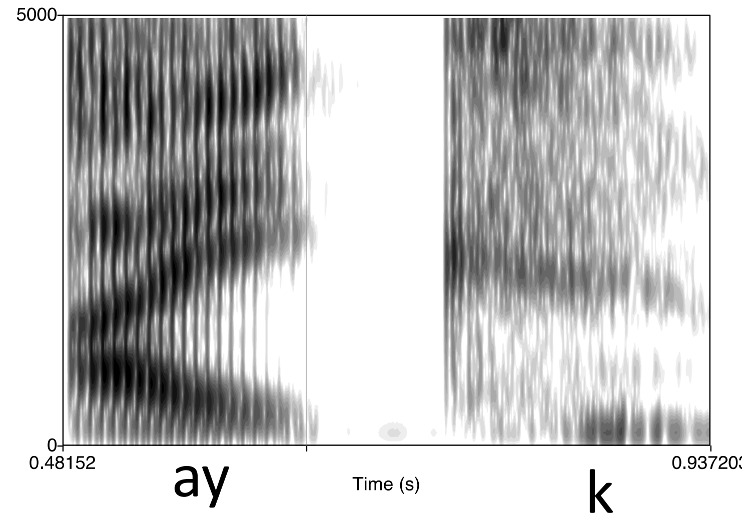
Language Model

- 利用語言模型, 可以限制Acoustic Model找到的Phone Sequence, 組合起來要長的像人話
- 常用的LM: N-grams

<s> the cat sat on the mat </s>
<s> **the** cat sat on the mat </s>
<s> **the** **cat** sat on the mat </s>
<s> **the** **cat** **sat** on the mat </s>
<s> the **cat** **sat** **on** the mat </s>
<s> the cat **sat** **on** **the** mat </s>
<s> the cat sat **on** **the** **mat** </s>
<s> the cat sat on **the** **mat** **</s>**

Acoustic Model - State





- Important sequence model: **Hidden Markov Model (HMM)**
- HMMs standard model for many year (1970s-2010s)
  - each (context-dependent) phone modelled by an HMM
  - typically 3-emitting state topology, **left-to-right HMM**
  - non-emitting (end) states used for “gluing” models together

Acoustic Model - State

- **Each phoneme correspond to a sequence of states**

what do you think

Phone:

hh w aa t d uw y uw th ih ng k

Tri-phone:

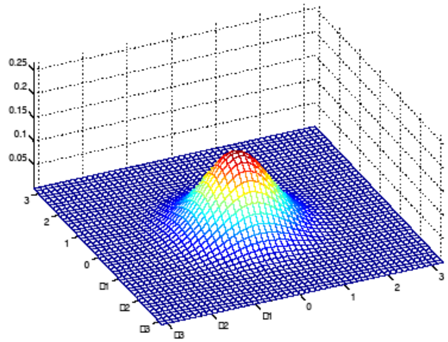
…… t-d+uw d-uw+y uw-y+uw y-uw+th ……

State:

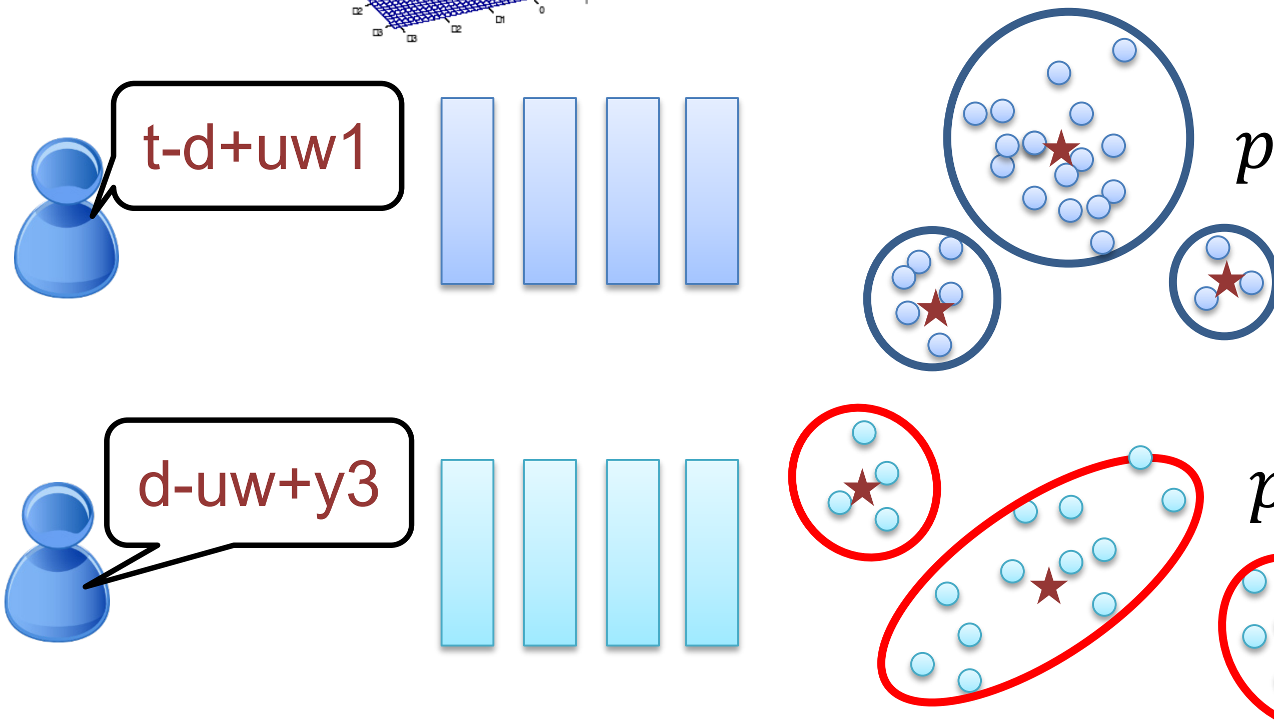
t-d+uw1 t-d+uw2 t-d+uw3 d-uw+y1 d-uw+y2 d-uw+y3

Acoustic Model - State

- **Each state has a stationary distribution for acoustic features**



Gaussian Mixture Model (GMM)



$$p(\mathbf{o}|\text{"}t-d+uw1\text{"})$$

$$p(\mathbf{o}|\text{"}d-uw+y3\text{"})$$

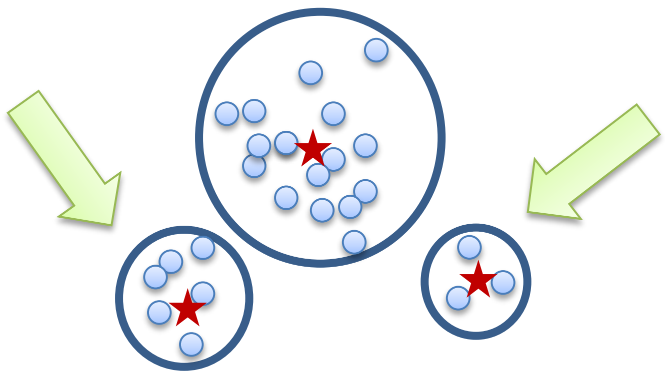
Acoustic Model - State

- 我們可以暴力組合出所有triphone：
  - 假設有50個phone, 總共就會有$50x50x50 = 125,000$個tri-phone
  - 實際用不到那麼多
- States which are clustered together will share their Gaussians
  - Decision-Tree based clustering of triphone states

Tied-state

pointer

$p(\mathbf{o}|"d-uw+y3")$



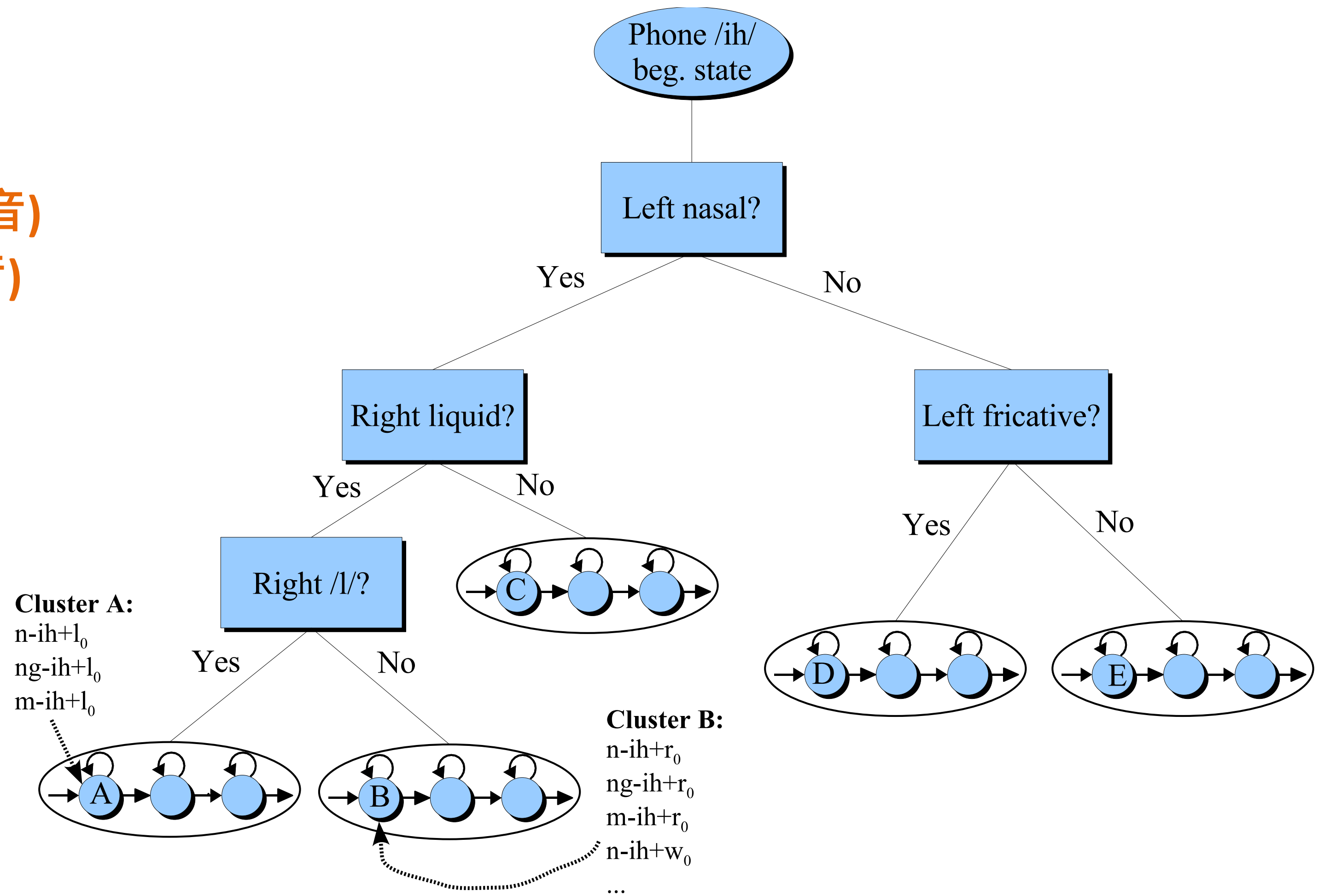
pointer

$p(\mathbf{o}|"k-uw+i3")$

Same Address

Acoustic Model - State tying/clustering

- How do we decide which triphones to cluster together?
  - 可以用語言學家定義的broad phonetic classes
    - Stop
    - Nasal (鼻音)
    - Liquid (流音)
    - Fricative (摩擦音)
    - Sibilant (齒擦音)
    - Vowel (母音)
    - Lateral (邊音)
    - …

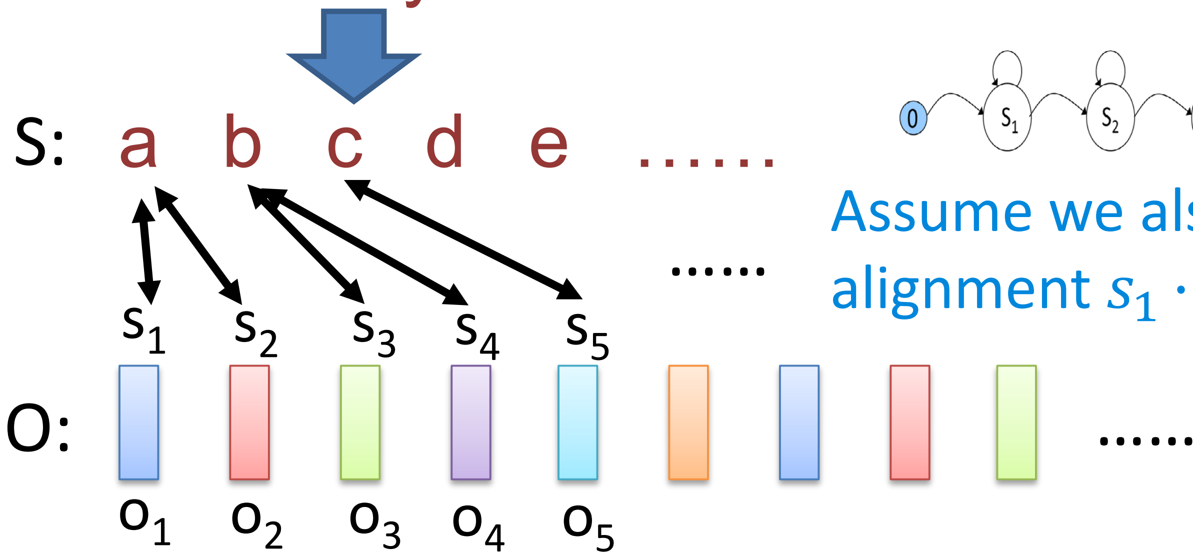


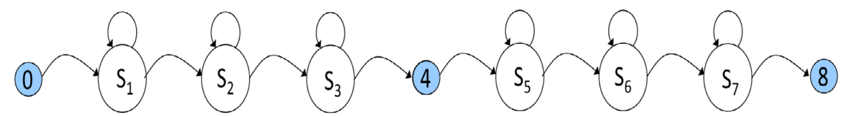
Decoding

$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in L} \boxed{p(\mathbf{O}|\mathbf{W})} P(\mathbf{W})$$

$$p(\mathbf{O}|\mathbf{W}) = P(\mathbf{O}|S)$$

W: what do you think?





Assume we also know the alignment $s_1 \cdots s_T$

$$p(\mathbf{O}|S) = \prod_{t=1}^{T} \underbrace{p(s_t|s_{t-1})}_{\text{transition}} \underbrace{p(\mathbf{o}_t|s_t)}_{\text{emission}}$$

Decoding

$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in \boldsymbol{L}} \boxed{p(\mathbf{O}|\mathbf{W})} P(\mathbf{W})$$

$$p(\mathbf{O}|\mathbf{W}) = P(\mathbf{O}|S)$$

W: what do you think?

S: a b c d e ……

Actually, we don't know the alignment → Viterbi algorithm

s_1 s_2 s_3

O: ……

o_1 o_2 o_3 o_4 o_5

$$p(\mathbf{O}|S) \approx \max_{s_1 \cdots s_T} \prod_{t=1}^{T} p(s_t|s_{t-1})\, p(\mathbf{o}_t|s_t)$$

- **Viterbi algorithm is used to find the most probable path through a probabilistically scored time/state lattice**

Evaluation

- How to evaluate the word string output by a speech recognizer?

- Word Error Rate!

$$WER = \frac{100\% \times (\boldsymbol{Insertions} + \boldsymbol{Substitutions} + \boldsymbol{Deletions})}{\boldsymbol{Total\ Word\ in\ Correct\ Transcript}}$$

  - Insertion: 多一個字, deletion: 少一個字, substitution: 字錯
  - Example:

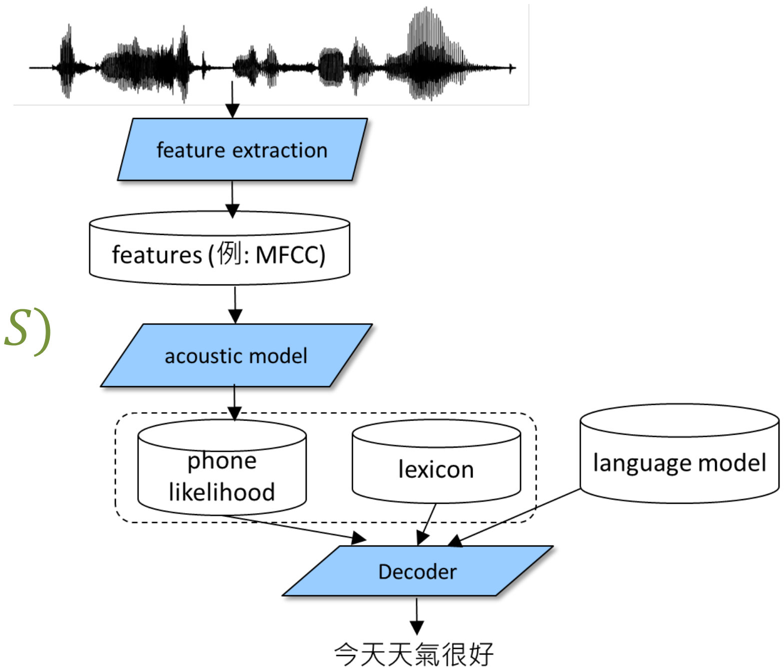
```
REF:  portable ****  PHONE UPSTAIRS last night so
HYP:  portable FORM  OF    STORES   last night so
Eval           I     S     S
```

$WER = 100\% \times (1 + 2 + 0)/6 = 50\%$

- WER越低，表示辨識器的效果越好！

Summary: ASR Architecture

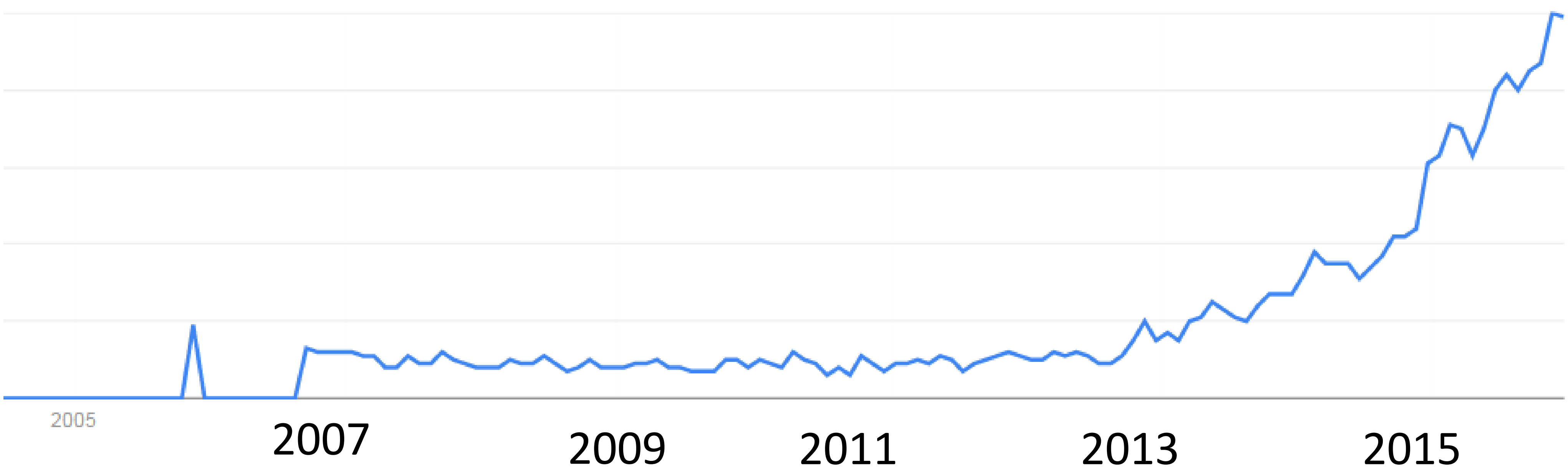
- **Feature Extraction:**
  - 39 “MFCC” features
- **Acoustic Model:**
  - Gaussians for computing phone likelihood $p(O|S)$
- **Lexicon/Pronunciation Model**
  - HMM: what phones can follow each other
- **Language Model**
  - N-grams for computing $p(w_i|w_{i-1})$
- **Decoder**
  - Viterbi algorithm: dynamic programming for combining all these to get word sequence from speech



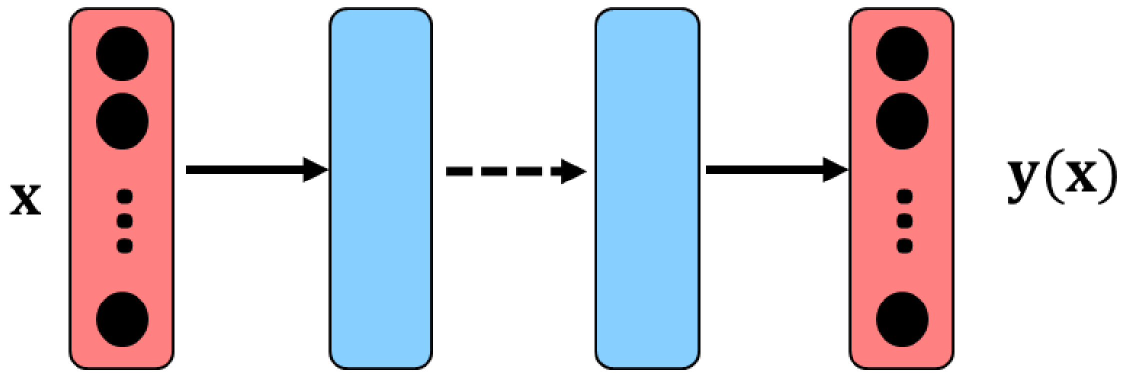
How to use Deep Learning?

Deep Neural Network

- **Deep learning attracts lots of attention**
  - Deep learning obtains many exciting results
- **From Wikipedia:**
  - 深度學習（deep learning）是機器學習的分支，是一種試圖使用包含複雜結構或由多重非線性變換構成的多個處理層對資料進行高層抽象的演算法。



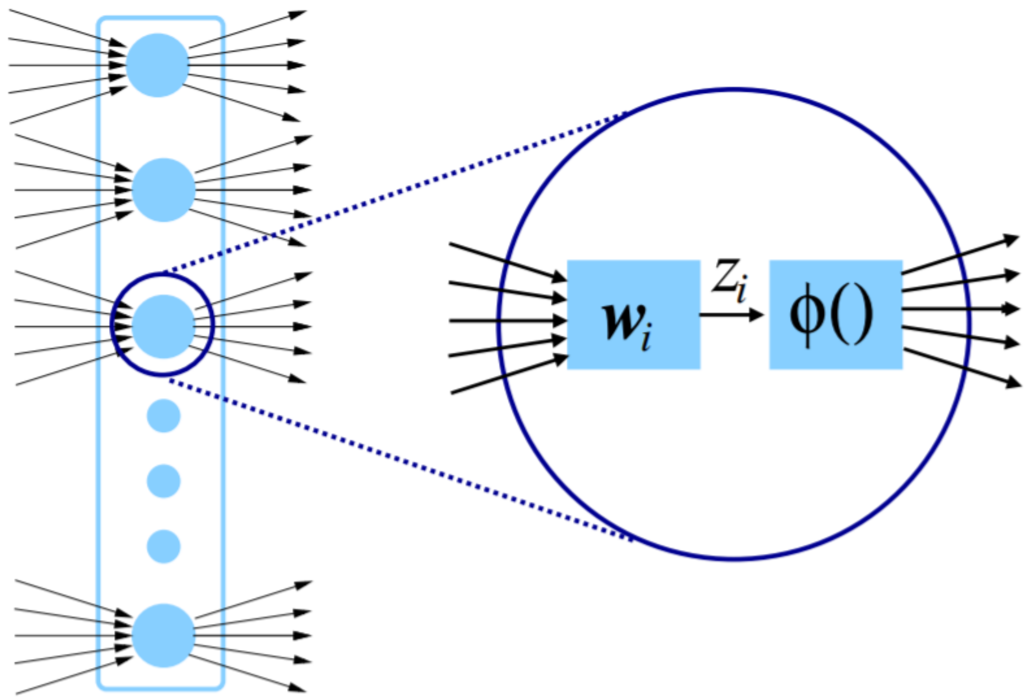
Deep Neural Network



- General mapping process from input $\mathbf{x}$ to output $\mathbf{y}(\mathbf{x})$
  - deep refers to number of hidden layers

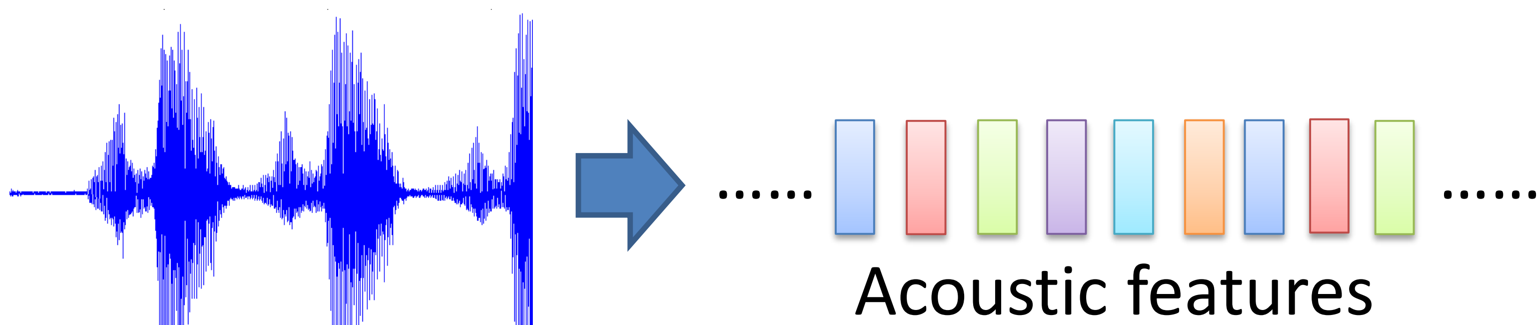
- Output from the previous layer connected to following layer:
  - $x^{(k)}$ is the input to layer k
  - $x^{(k+1)} = y^{(k)}$ the output from layer k

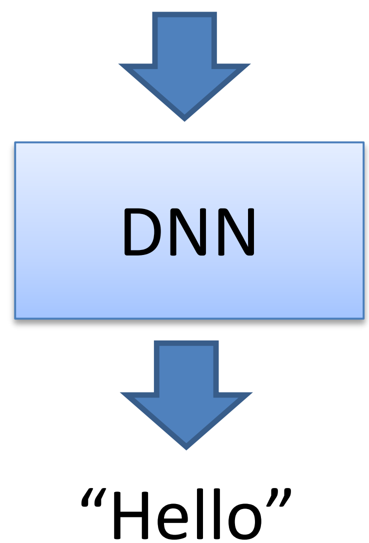
Deep Neural Network



- **General form for layer k:**
  - $y_i^{(k)} = \phi\left(\mathbf{w}_i\mathbf{x}^{(k)} + \mathbf{b}_i\right) = \phi(z_i^{(k)})$
    - $\mathbf{w}$: weight matrix
    - $\mathbf{b}$: bias vector
    - ϕ: activation function
      - Sigmoid, RELU, …etc

People imagine ……



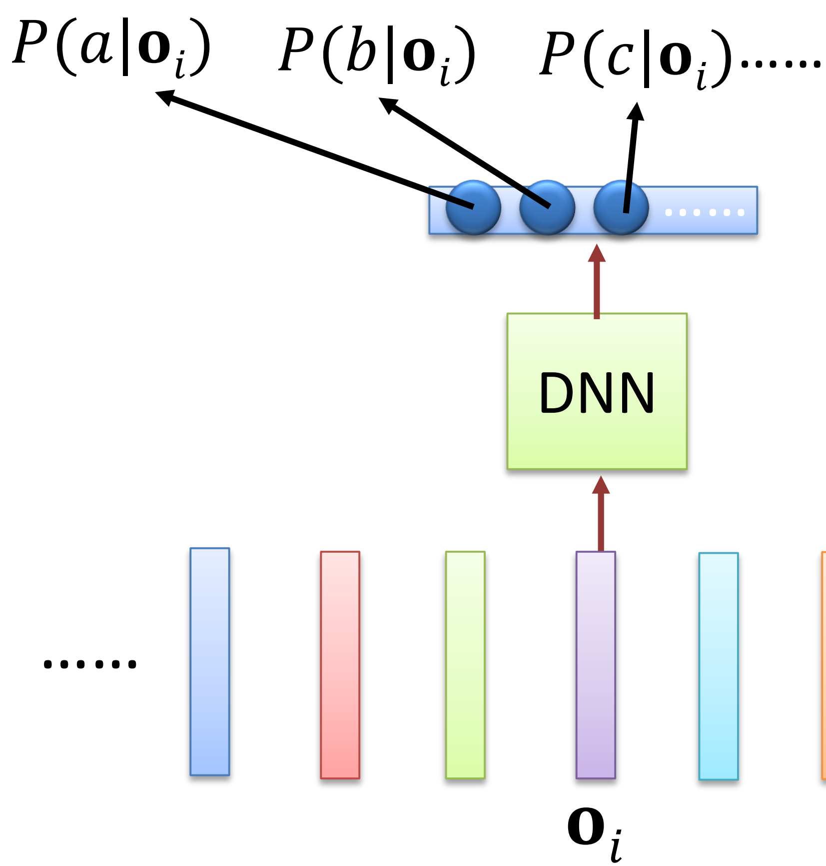


This can not be true!

DNN can only take fixed length vectors as input.

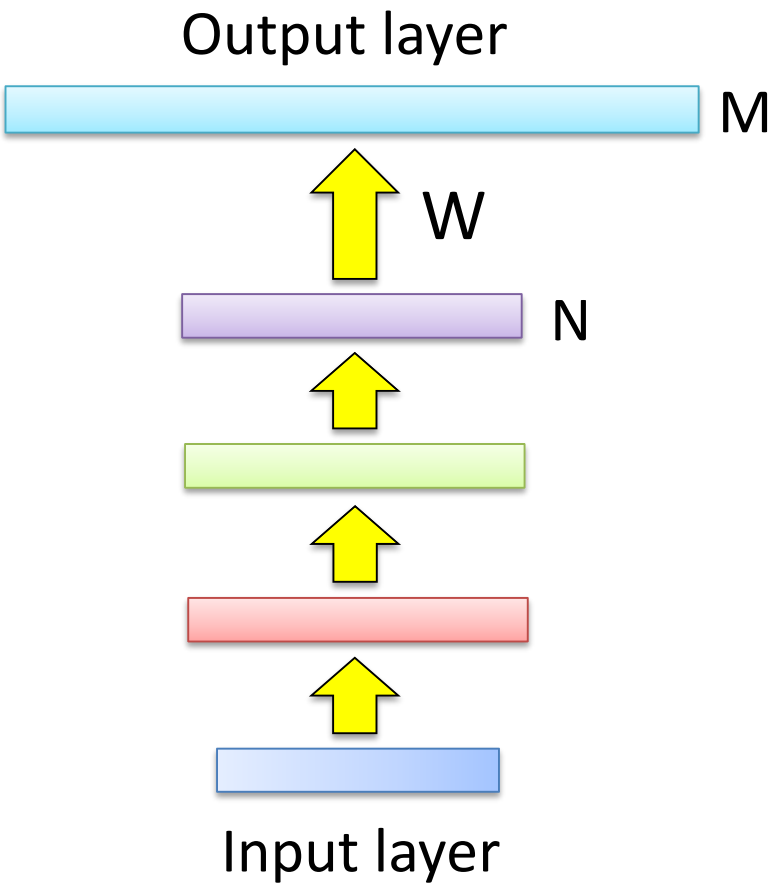
What DNN can do is ……

- **DNN output:**
  - Probability of each state
- **DNN input:**
  - One acoustic feature



Size of output layer = Num of states

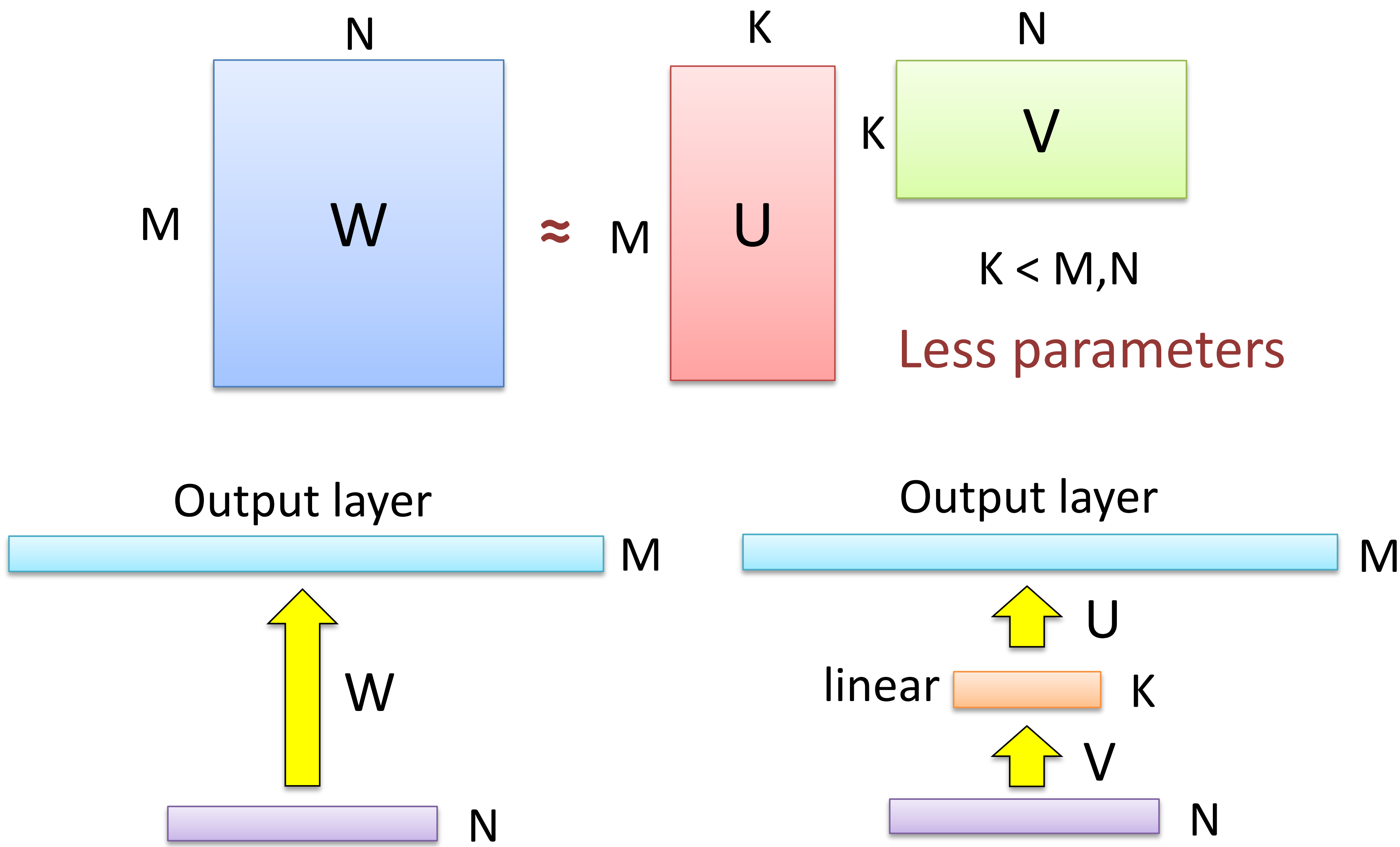
Low rank approximation



W: M X N

- **N is the size of the last hidden layer**
- **M is the size of output layer**
  - **Number of states**
- **M can be large if the outputs are the *states of tri-phone***
  - **e.g. 3000~5000**

Low rank approximation



How we use deep learning

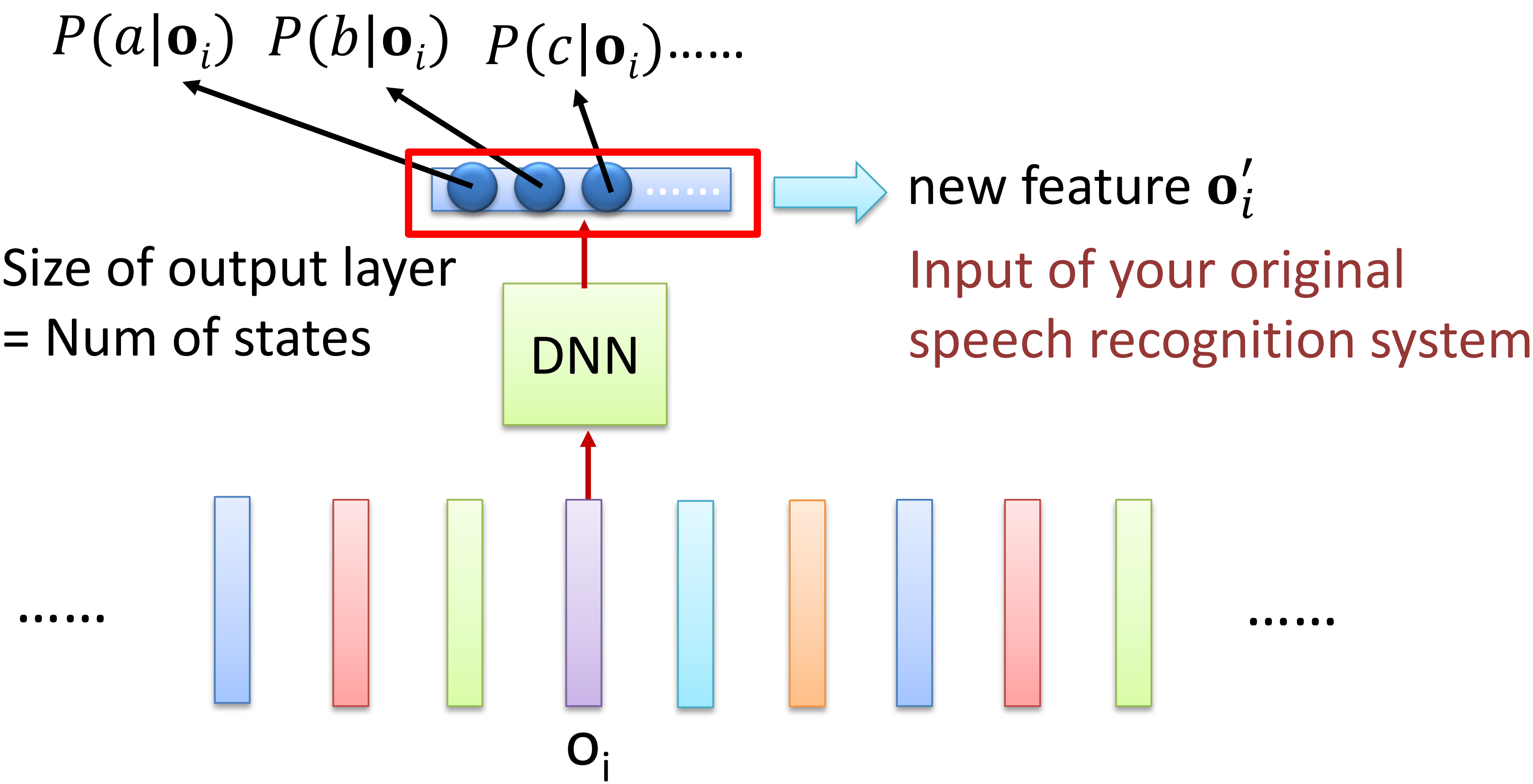
- There are three ways to use DNN for acoustic modeling
  - Way 1. Tandem
  - Way 2. DNN-HMM hybrid
  - Way 3. End-to-end

Efforts for exploiting deep learning

How to use Deep Learning?

Way 1: Tandem

Way 1: Tandem system



Last hidden layer or bottleneck layer are also possible.

How to use Deep Learning?

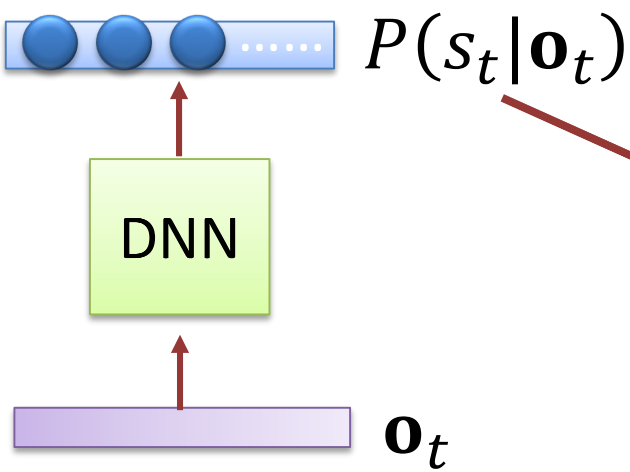
Way 2: DNN-HMM hybrid

Way 2: DNN-HMM Hybrid

$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in \boldsymbol{L}} p(\mathbf{W}|\mathbf{O}) = \arg\max_{\mathbf{W}\in \boldsymbol{L}} p(\mathbf{O}|\mathbf{W})P(\mathbf{W})$$

$$P(\mathbf{O}|W) \approx \max_{s_1\cdots s_T} \prod_{t=1}^{T} P(s_t|s_{t-1})\,\underline{P(\mathbf{o}_t|s_t)}$$

From DNN



$$P(\mathbf{o}_t|s_t) = \frac{P(\mathbf{o}_t, s_t)}{P(s_t)}$$

$$= \frac{P(s_t|\mathbf{o}_t)\cancel{P(\mathbf{o}_t)}}{P(s_t)}$$

Count from training data

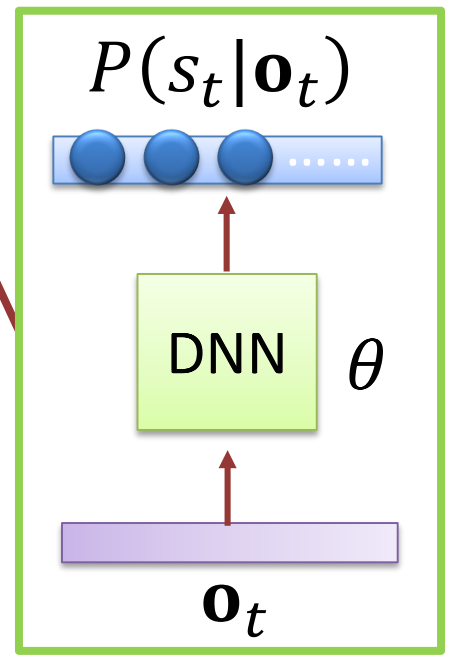
Way 2: DNN-HMM Hybrid

$$P(\mathbf{O}|\mathbf{W}) \approx \max_{s_1 \cdots s_T} \prod_{t=1}^{T} P(s_t|s_{t-1})\, P(\mathbf{o}_t|s_t)$$

$$\cancel{P(\mathbf{O}|\mathbf{W})} \approx \max_{s_1 \cdots s_T} \prod_{t=1}^{T} P(s_t|s_{t-1}) \frac{P(s_t|\mathbf{o}_t)}{P(s_t)}$$

$$P(\mathbf{O}|\mathbf{W};\theta)$$

DNN



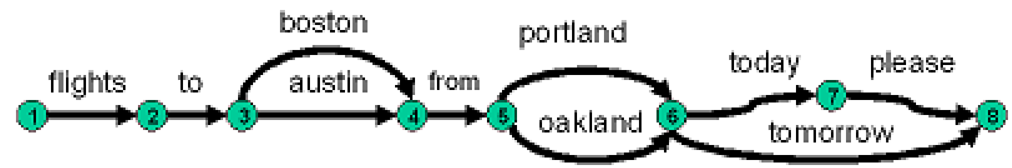
From original HMM

Count from training data

This assembled vehicle works …….

Way 2: DNN-HMM Hybrid

- ***Sequential Training***



$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in\boldsymbol{L}} p(\mathbf{O}|\mathbf{W};\theta)P(\mathbf{W})$$

Given training data $(\mathbf{O}_1, \mathbf{W}_1^*), (\mathbf{O}_2, \mathbf{W}_2^*), \cdots (\mathbf{O}_r, \mathbf{W}_3^*), \cdots$

Find-tune the DNN parameters θ such that

$P(\mathbf{O}_r|\mathbf{W}_r^*;\theta)P(\mathbf{W}_r^*)$ → increase

$P(\mathbf{O}_r|\mathbf{W};\theta)P(\mathbf{W})$ → decrease

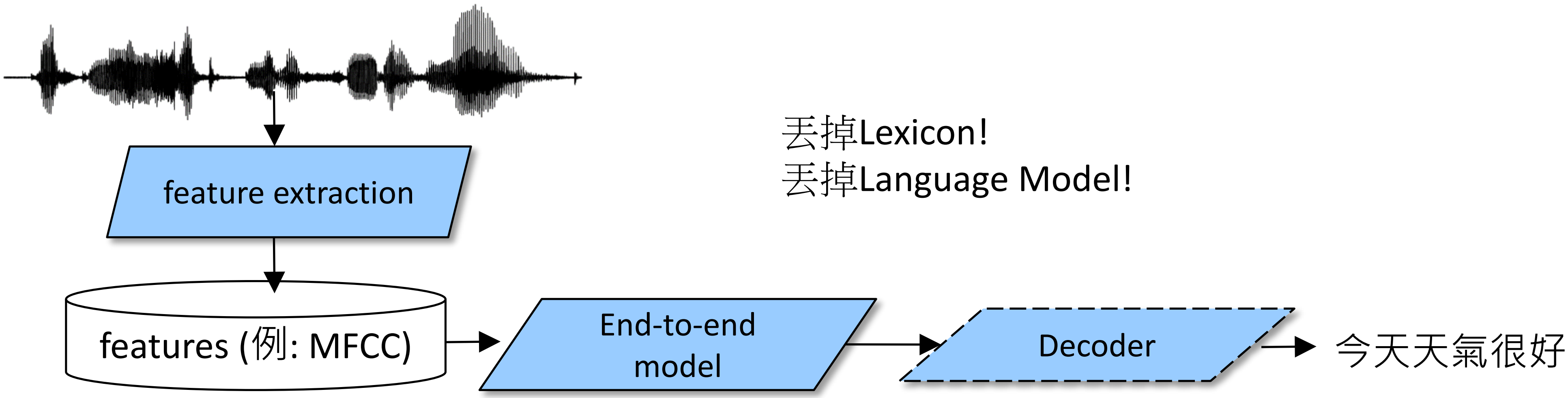
($\mathbf{W}$ is any word sequence different from $\mathbf{W}_r^*$)

How to use Deep Learning?

Way 3: End-to-end

DELTA

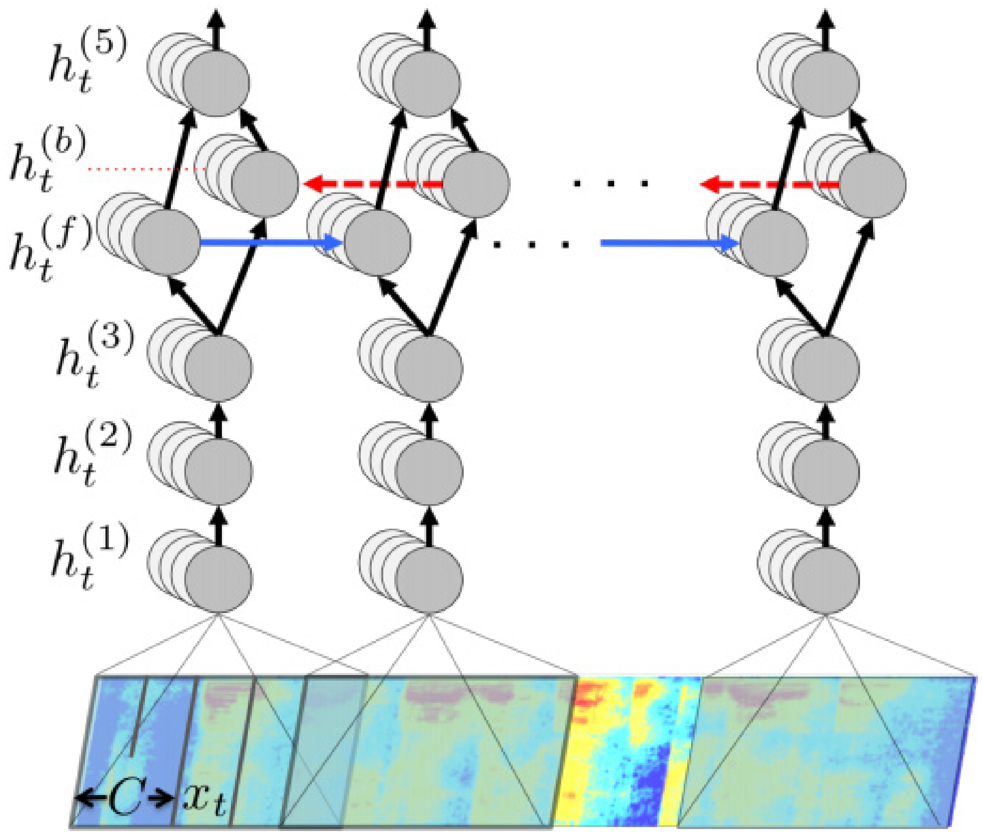
End-to-end Framework



丟掉Lexicon!
丟掉Language Model!

- **Directly train model to solve task ("speech-to-text")**
  - **single model trained**
  - **no separate acoustic and language models**
- **More complicated to incorporate additional LM data**
- **Output layer從state換成character, phone or words**

Way 3: End-to-end - Character



Input: acoustic features (spectrograms)

Output: characters (and space) + null (~)

No phoneme and lexicon

(No OOV problem)

A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, A. Ng "Deep Speech: Scaling up end-to-end speech recognition", arXiv:1412.5567v2, 2014.

Way 3: End-to-end - Character

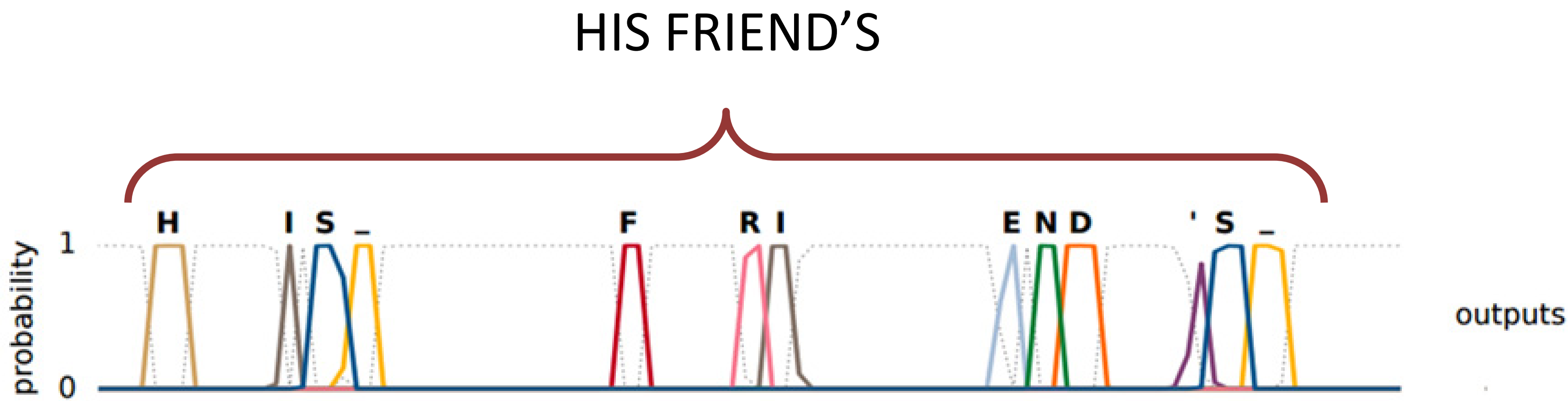


Figure 4. **Network outputs.** The figure shows the frame-level character probabilities emitted by the CTC layer (different colour for each character, dotted grey line for 'blanks'), along with the corresponding training errors, while processing an utterance. The target transcription was 'HIS_FRIENDS_', where the underscores are end-of-word markers. The network was trained with WER loss, which tends to give very sharp output decisions, and hence sparse error signals (if an output probability is 1, nothing else can be sampled, so the gradient is 0 even if the output is wrong). In this case the only gradient comes from the extraneous apostrophe before the 'S'. Note that the characters in common sequences such as 'IS', 'RI' and 'END' are emitted very close together, suggesting that the network learns them as single sounds.

Graves, Alex, and Navdeep Jaitly. "Towards end-to-end speech recognition with recurrent neural networks." *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*. 2014.

Way 3: End-to-end – Word?

- **training corpus: 1.2 billions words, vocabulary: 1.7 million words.**
- **125,000 hours of semi-supervised acoustic training data**
  - **deep bi-directional LSTM RNNs with CTC loss**

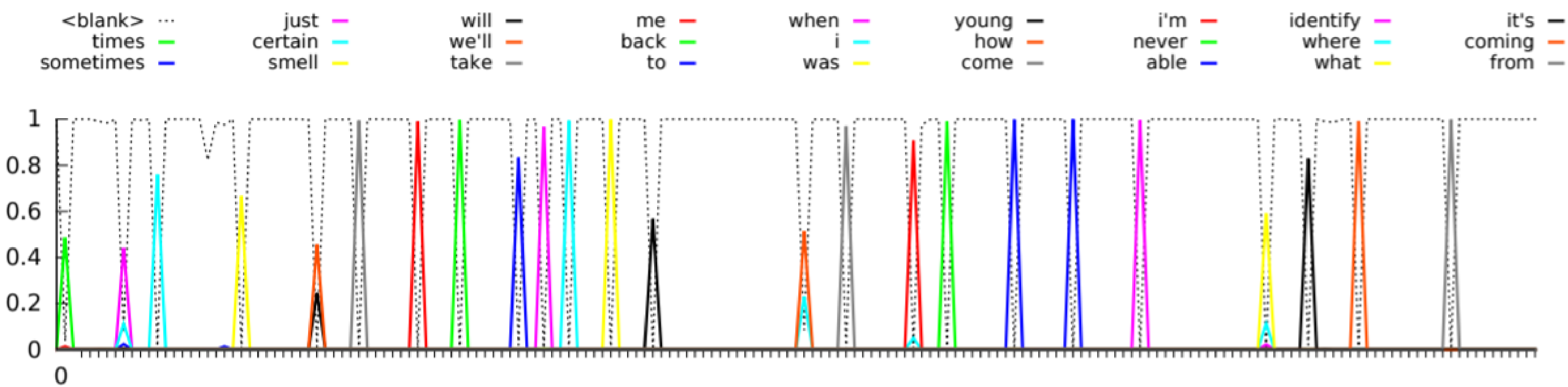


Figure 1: The word posterior probabilities as predicted by the NSR model at each time-frame (30 msec) for a segment of music video 'Stressed Out' by Twenty One Pilots. We only plot the word with highest posterior and the missing words from the correct transcription: '*Sometimes a certain smell will take me back to when I was young, how come I'm never able to identify where it's coming from*'.

Hagen Soltau et al.,"Neural Speech Recognizer: Acoustic-to-Word LSTM Model for Large Vocabulary Speech Recognition". arXiv. 2016.

Why Deep Learning?

Deeper is better?

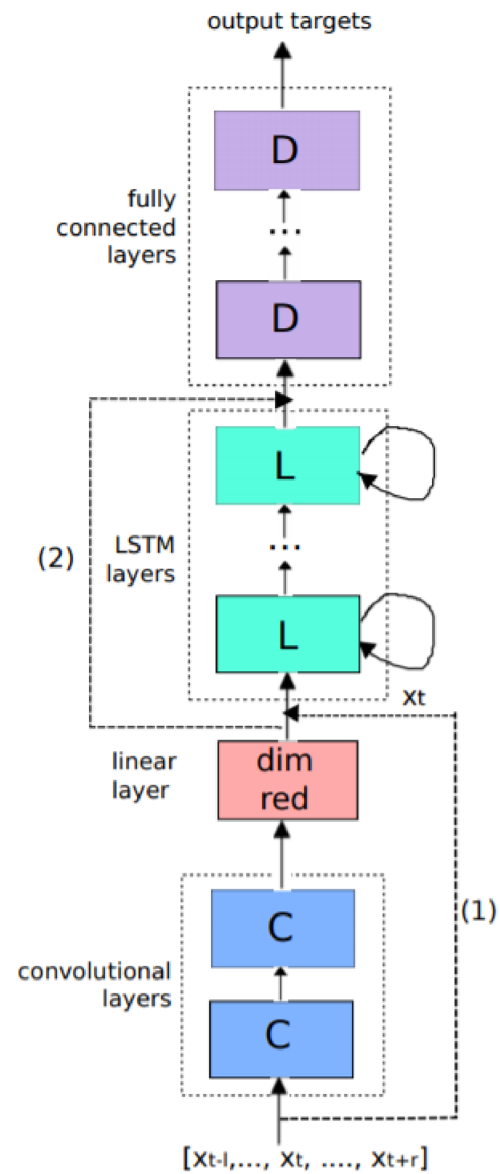
Layer X Size	Word Error Rate (%)
1 X 2k	24.2
2 X 2k	20.4
3 X 2k	18.4
4 X 2k	17.8
5 X 2k	17.2
7 X 2k	17.1

Not surprised, more parameters, better performance

Seide, Frank, Gang Li, and Dong Yu. "Conversational Speech Transcription Using Context-Dependent Deep Neural Networks." *Interspeech*. 2011.

Example: "General" Acoustic Model

- Example Architecture from Google (2015)
  - C: CNN layer (with pooling)
  - L: LSTM layer
  - D: fully connected layer
- Two multiple layer "skips"
  - (1) connects input to LSTM input
  - (2) connects CNN output to DNN input
- Additional linear projection layer
  - reduces dimensionality
  - and number of network parameters!



Speaker Adaptation

Speaker Adaptation

- Speaker adaptation: use different models to recognition the speech of different speakers
  - Collect the audio data of each speaker

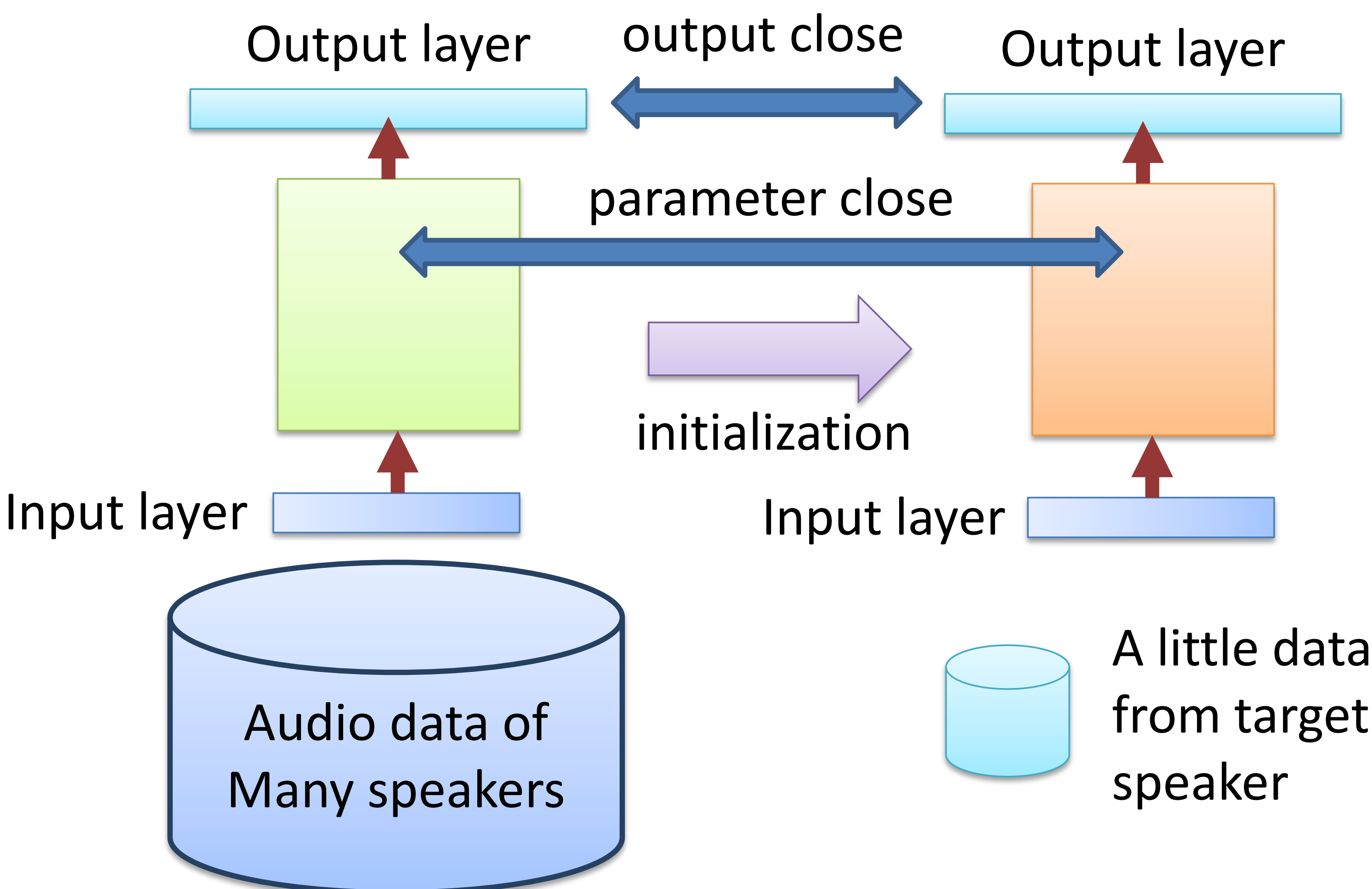
- A DNN model for each speaker
  - Challenge: limited data for training
    - Not enough data for directly training a DNN model
    - Not enough data for just fine-tune a speaker independent DNN model

Categories of Methods

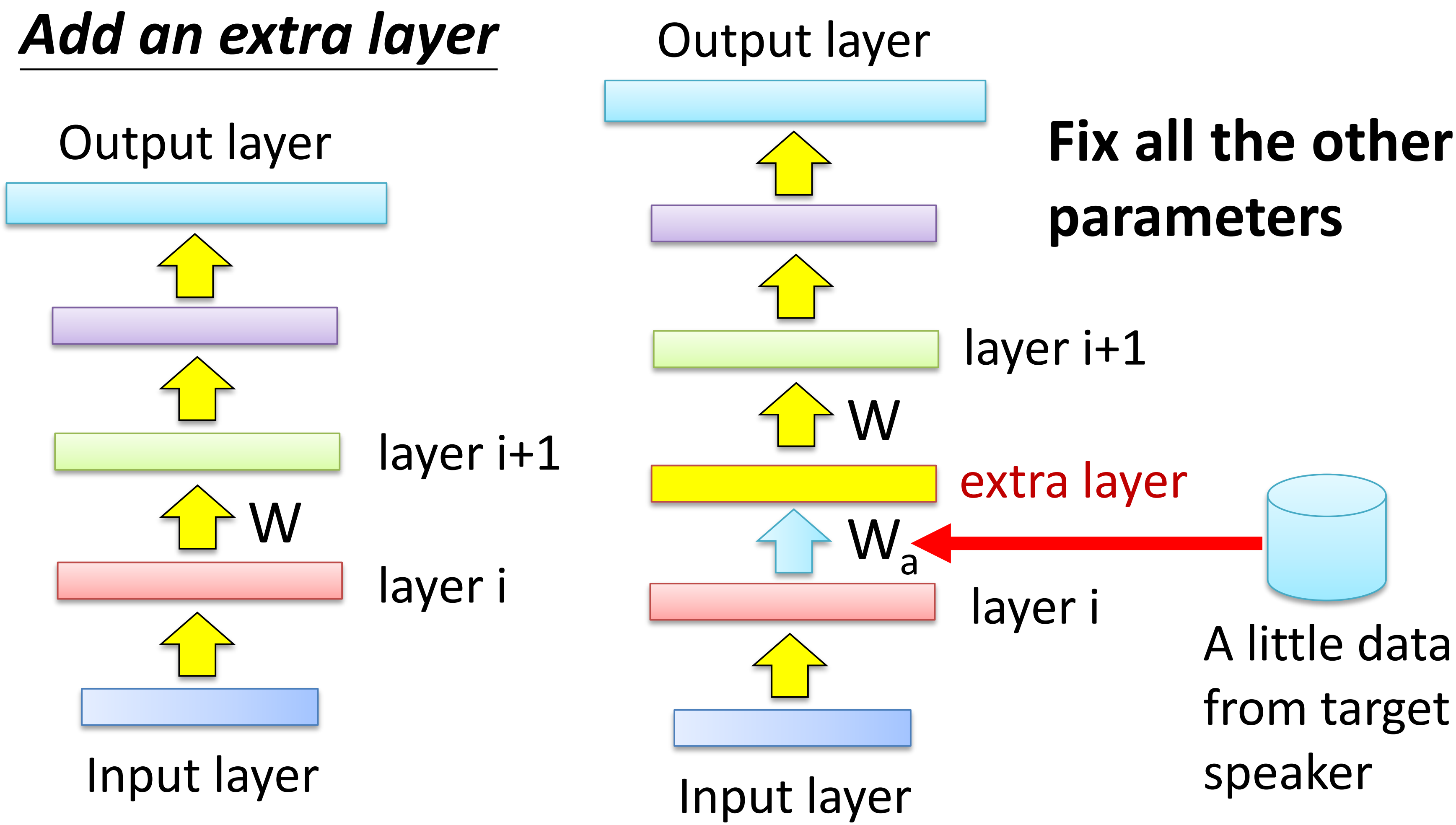
Conservative training	• Re-train the whole DNN with some constraints
Transformation methods	• Only train the parameter of one layer
Speaker-aware Training	• Do not really change the DNN parameters

Need less training data

Conservative Training

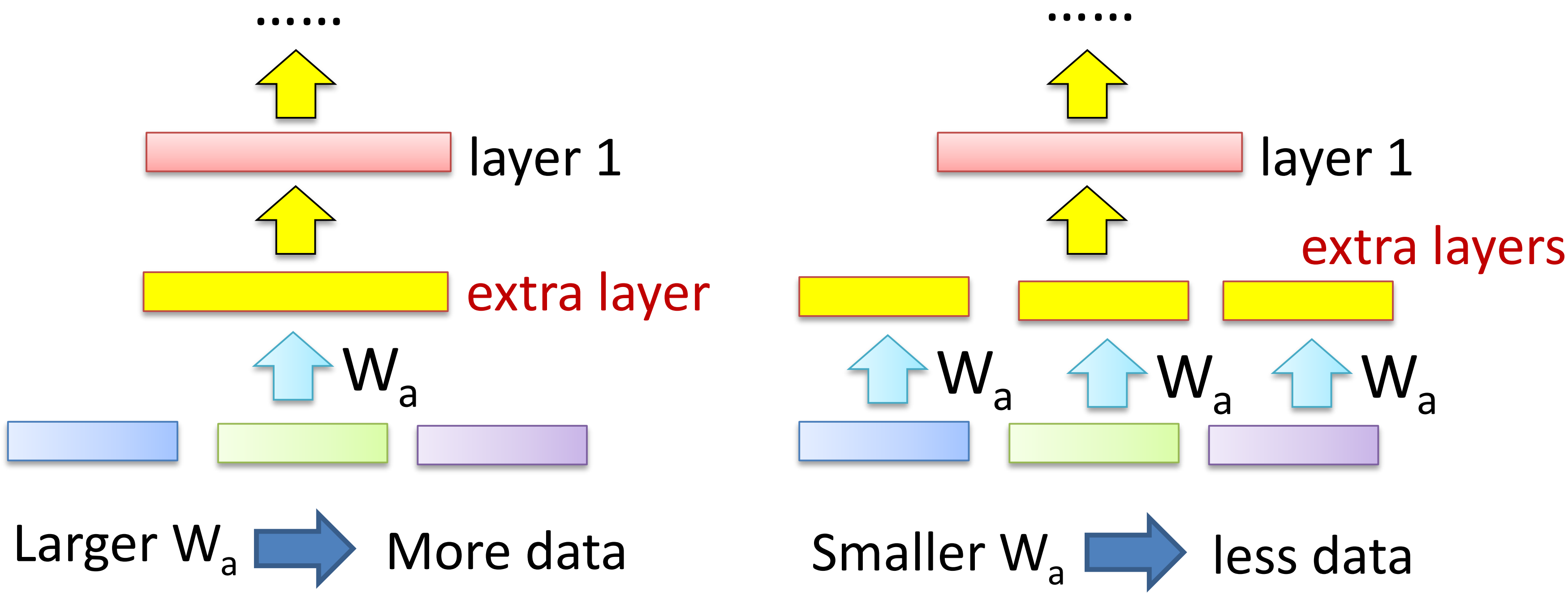


Transformation methods



Transformation methods

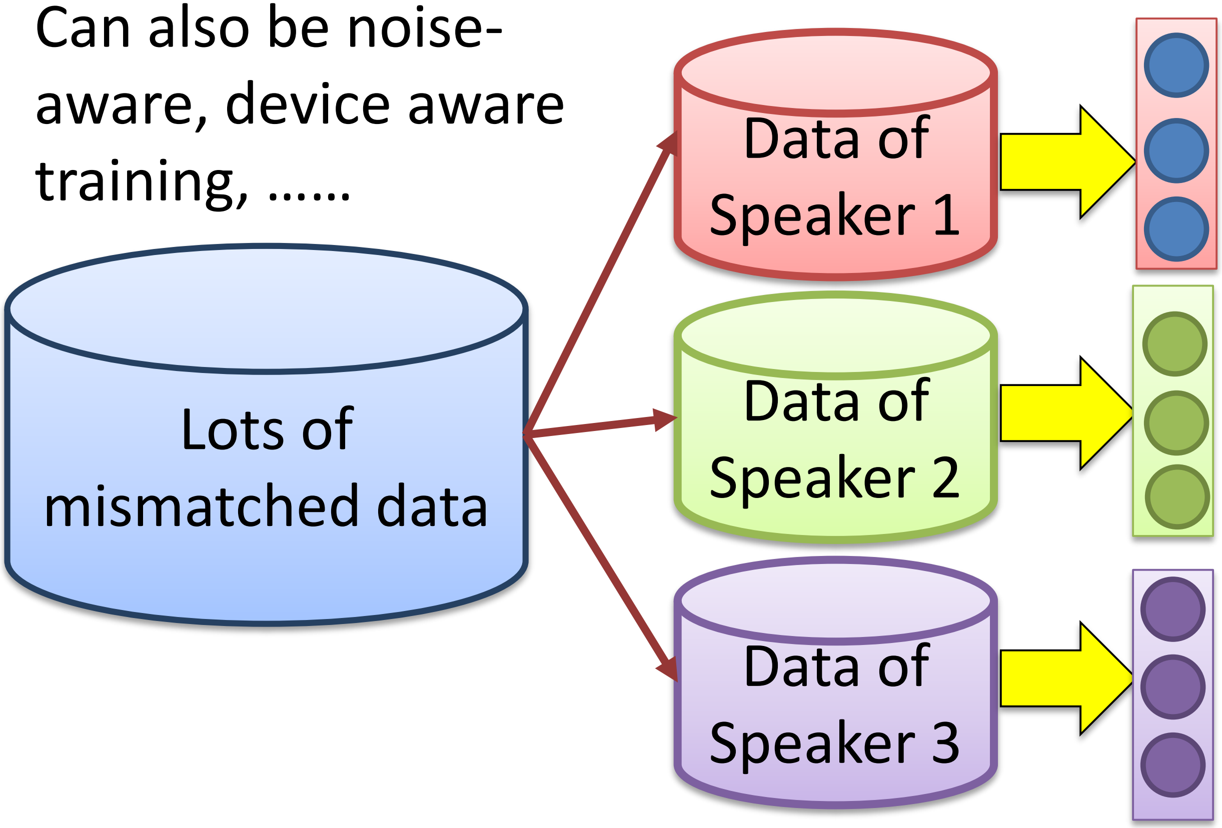
- Add the extra layer between the input and first layer
- With splicing



Speaker-aware Training

Speaker information (i-vector)

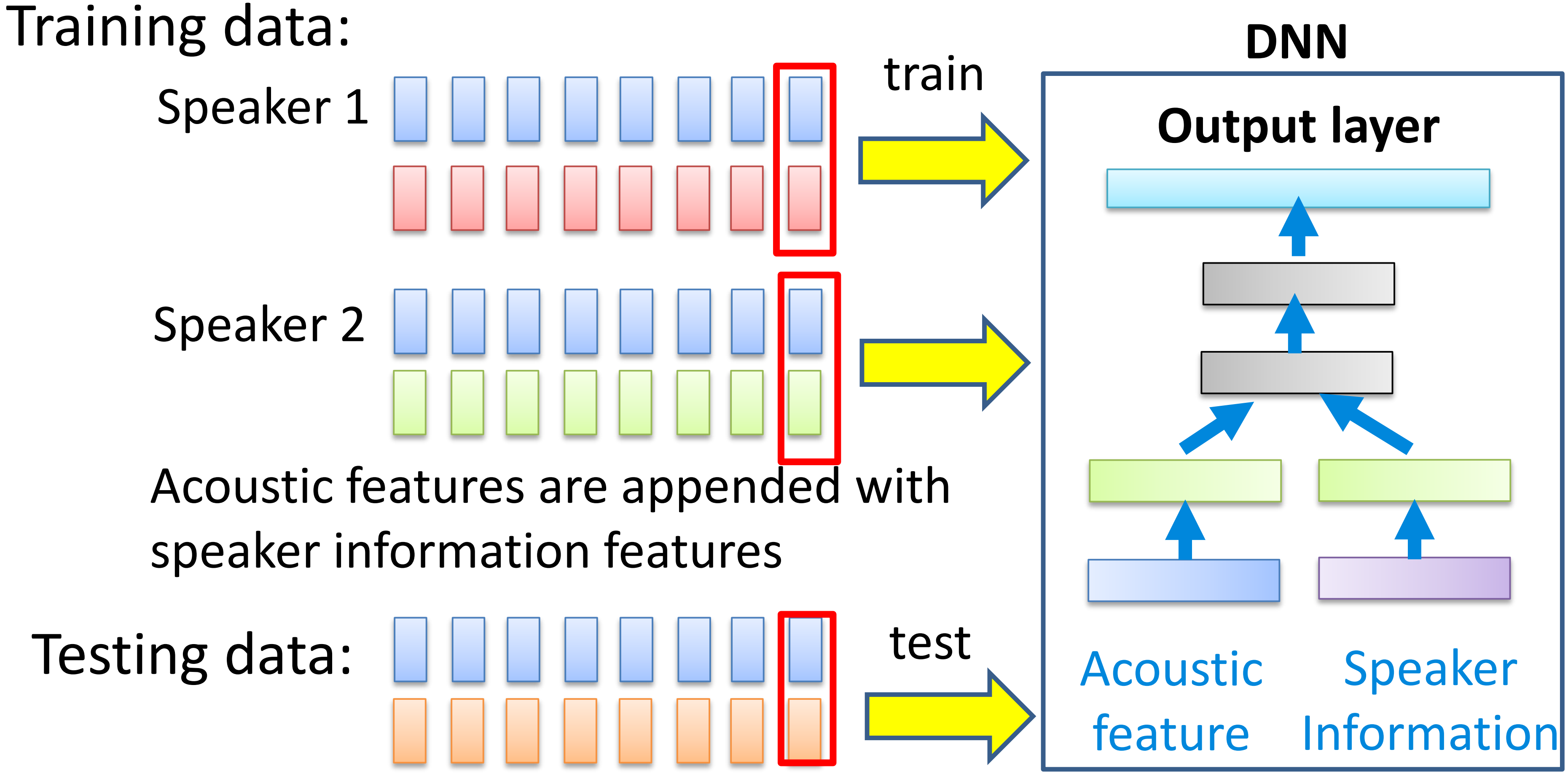
Can also be noise-aware, device aware training, ……



Fixed length low dimension vectors

Text transcription is not needed for extracting the vectors.

Speaker-aware Training



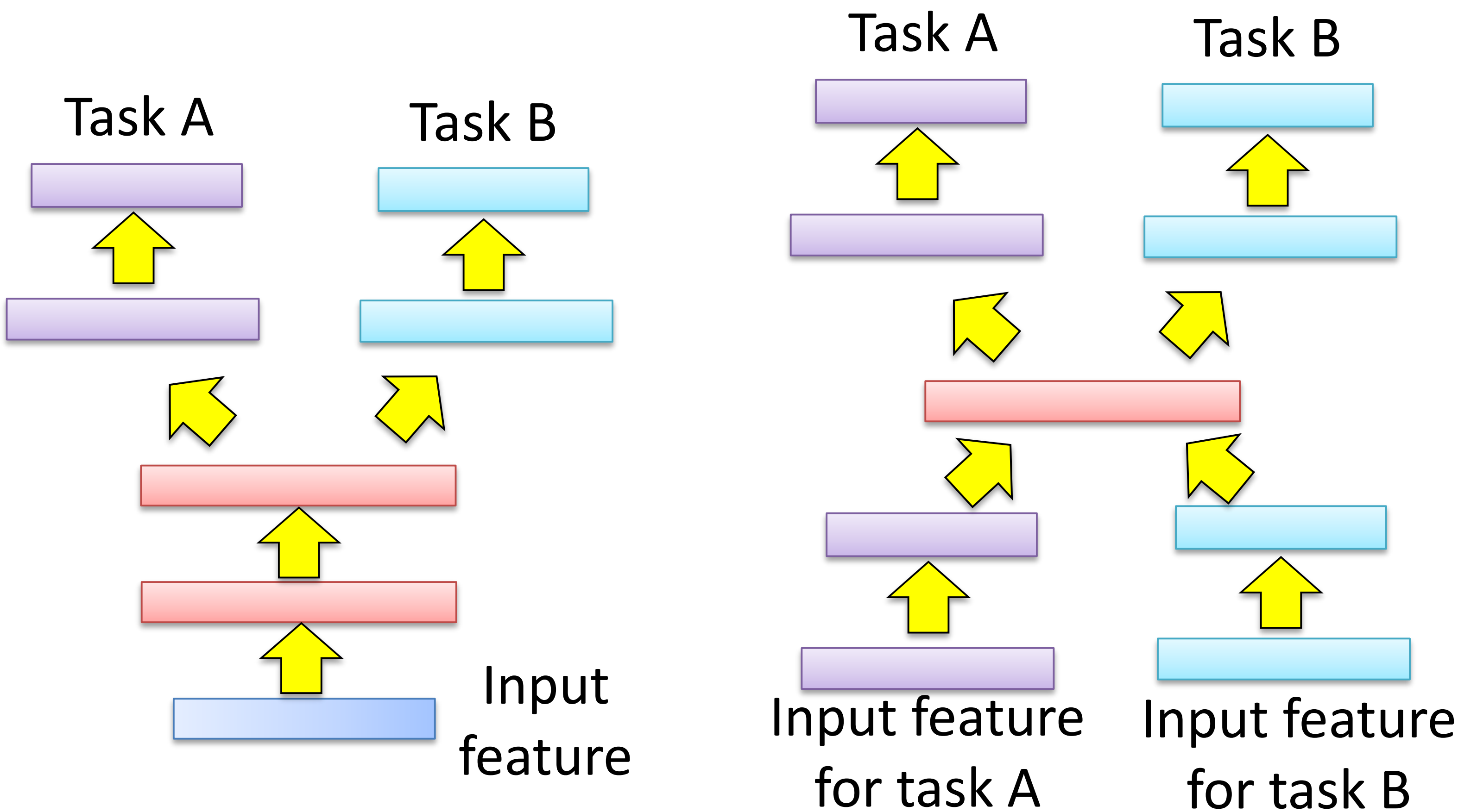
All the speaker use the same DNN model

Different speaker augmented by different features

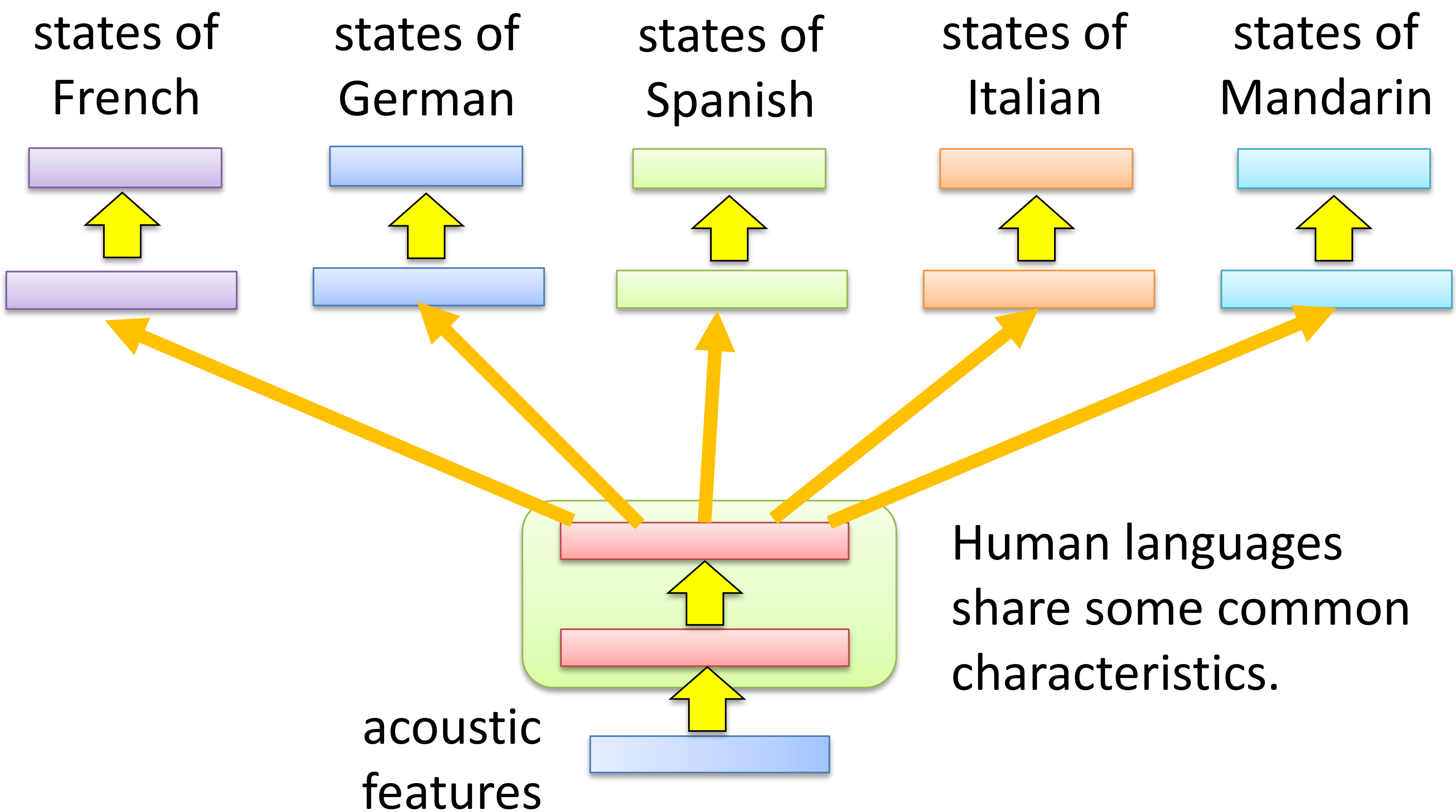
Multi-task Learning

Multitask Learning

- **The multi-layer structure makes DNN suitable for multitask learning**

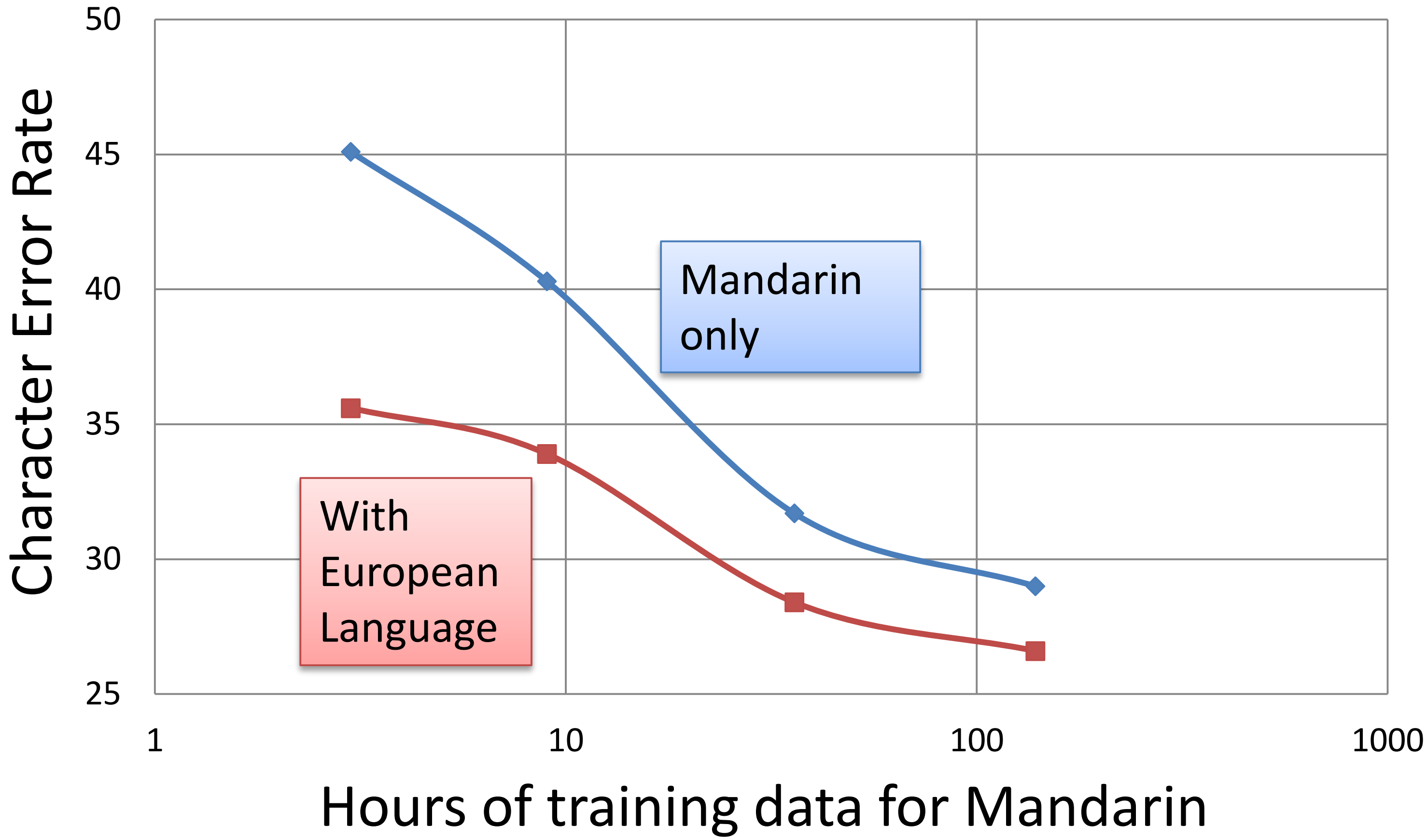


Multitask Learning - Multilingual



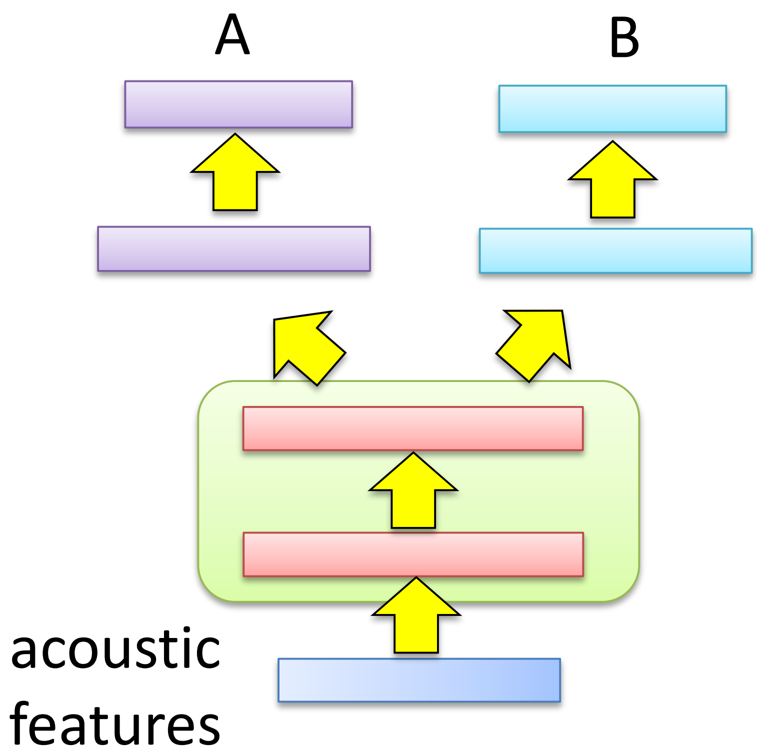
Human languages share some common characteristics.

Multitask Learning - Multilingual



Huang, Jui-Ting, et al. "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers." *Acoustics, Speech and Signal Processing (ICASSP), 2013*

Multitask Learning
- Different auxillary Tasks



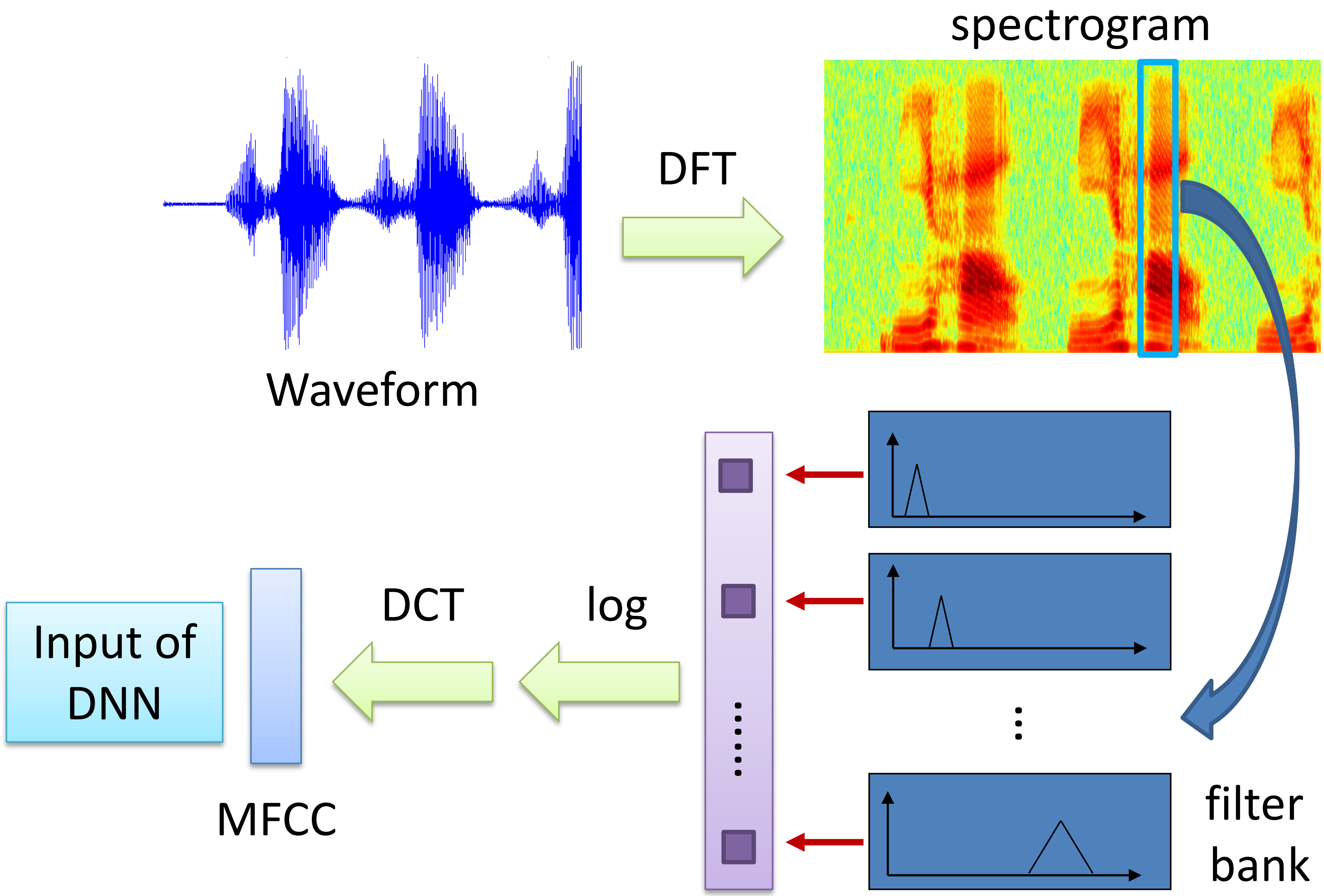
Main task	Auxillary task
A= state	B = phoneme
A= state	B = gender
A= state	B = grapheme (character)

Dongpeng Chen, Mak, B., Cheung-Chi Leung, Sivadas, S., "Joint acoustic modeling of triphones and trigraphemes by multi-task learning deep neural networks for low-resource speech recognition," ICASSP 2014

Deep Learning for Acoustic Modeling

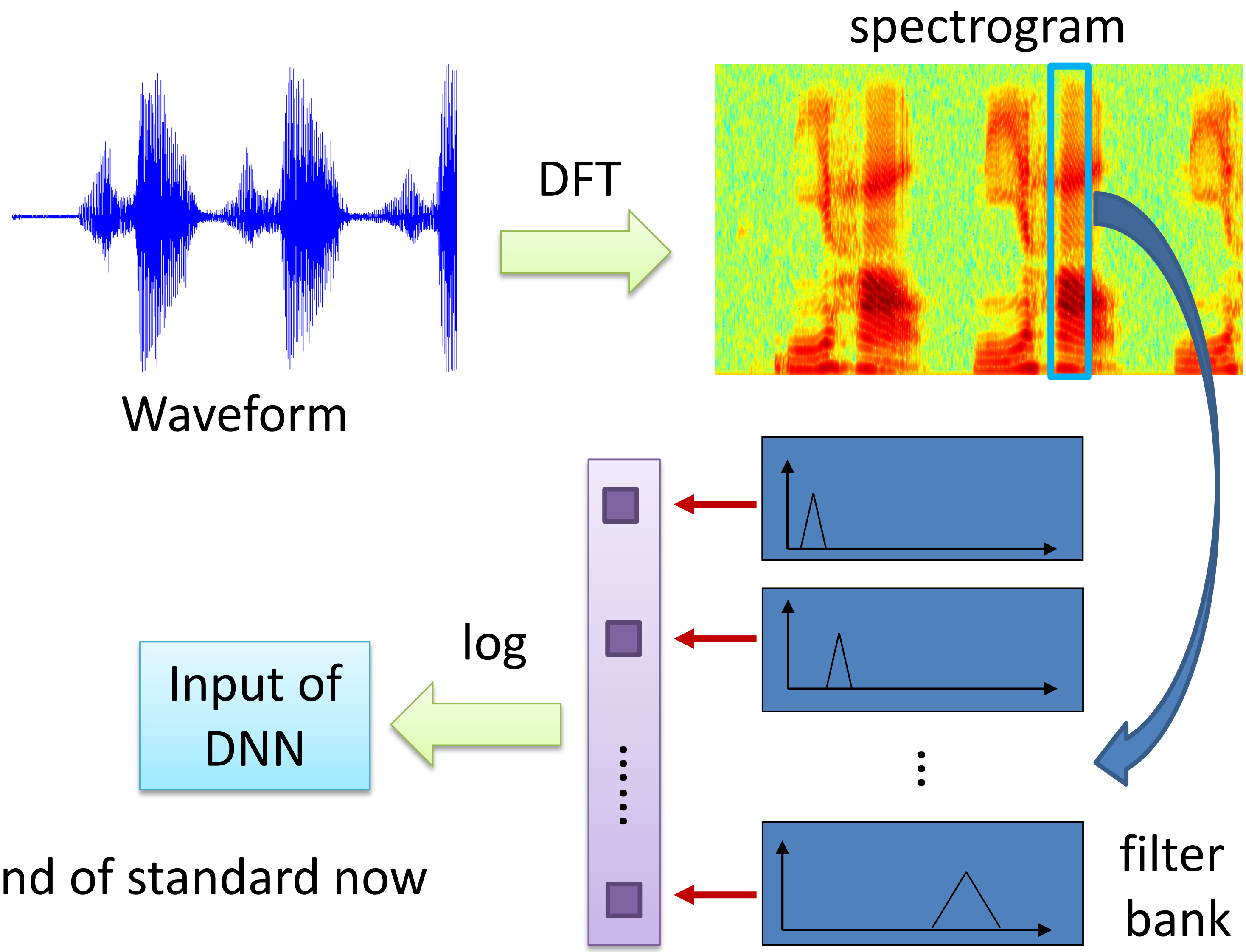
New acoustic features

MFCC



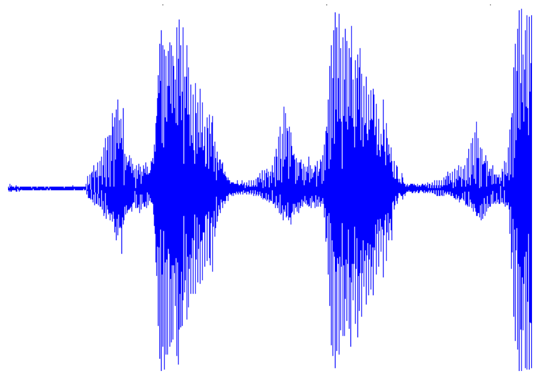
spectrogram
DFT
Waveform
Input of DNN
DCT
log
MFCC
filter bank

Filter-bank Output



- Kind of standard now

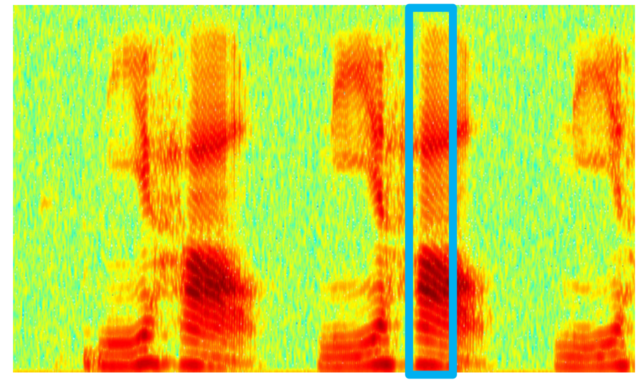
Spectrogram



Waveform

DFT

spectrogram



Input of DNN

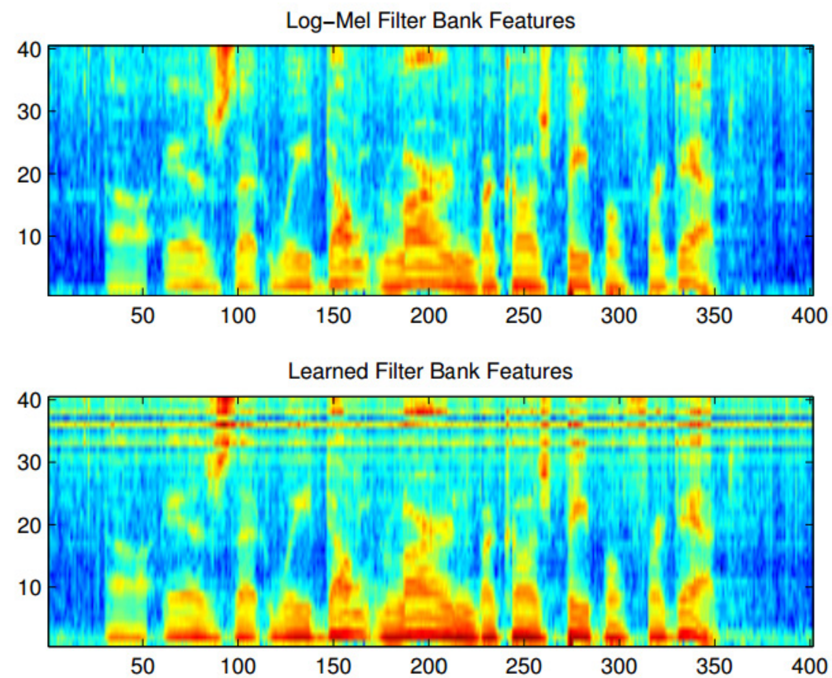
- common today

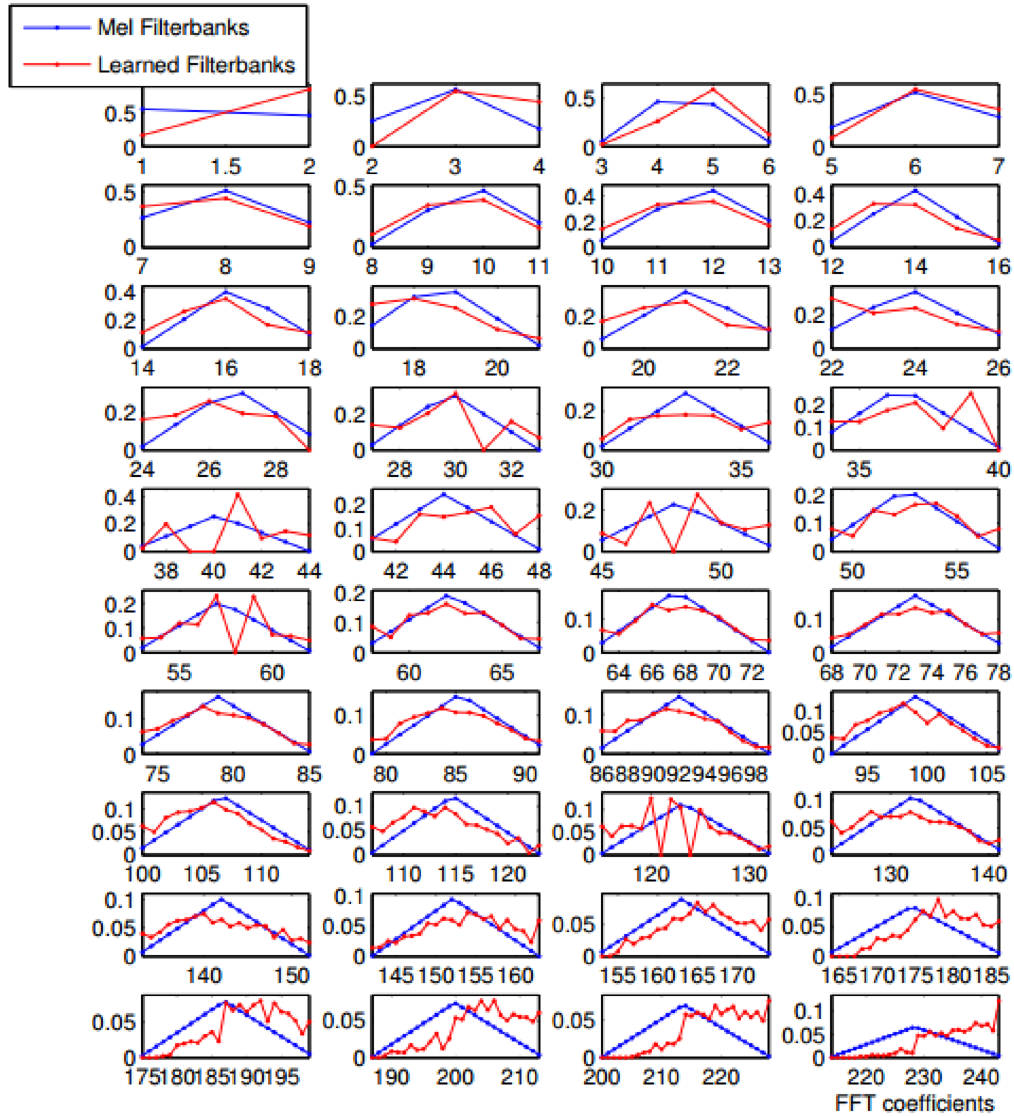
- 5% relative improvement over fitlerbank output

Sainath, T. N., et al., “Learning filter banks within a deep neural network framework,” In *ASRU, 2013*

Spectrogram

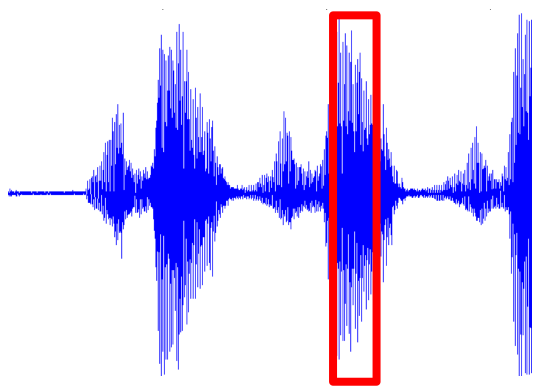
- **Learning fbanks within DNN**





Sainath, T. N., et al., "Learning filter banks within a deep neural network framework," In *ASRU, 2013*

Waveform?



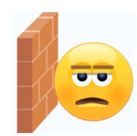
Waveform

Input of DNN

- If success, no Signal & Systems 

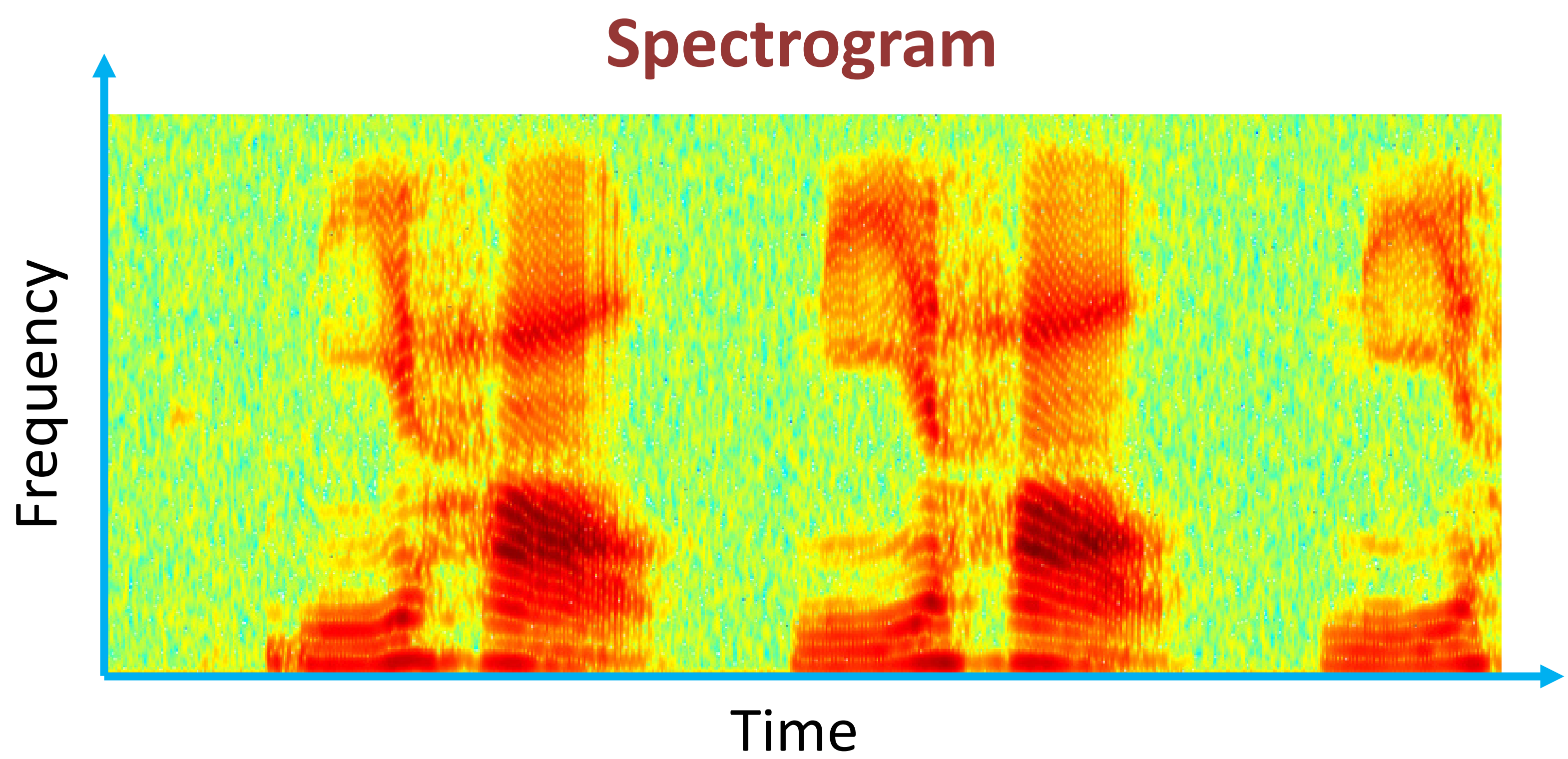
- People tried, but not better than spectrogram yet

Tüske, Z et al., "Acoustic modeling with deep neural networks using raw time signal for LVCSR," In *INTERPSEECH 2014*

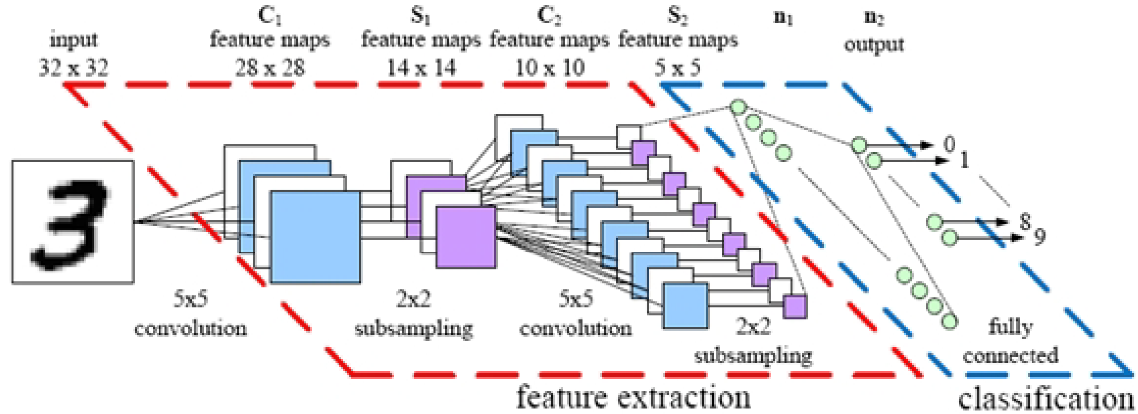
- Still need to take Signal & Systems …… 

Convolutional
Neural Network (CNN)

- Speech can be treated as images



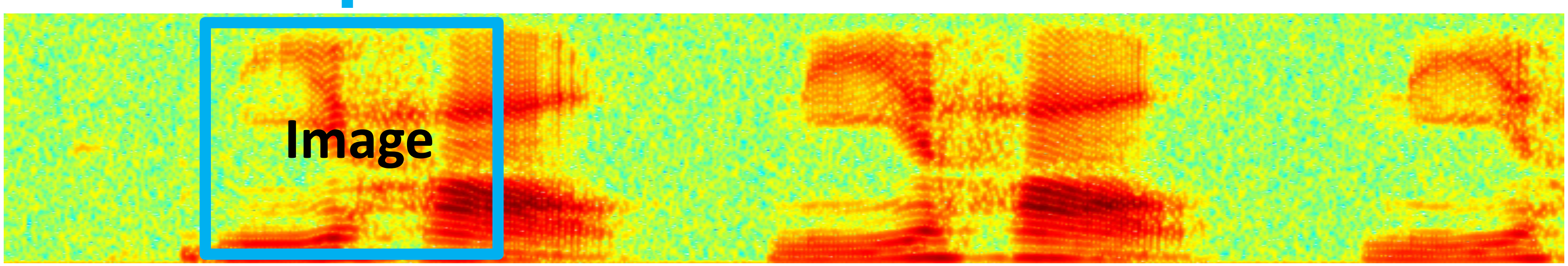
CNN

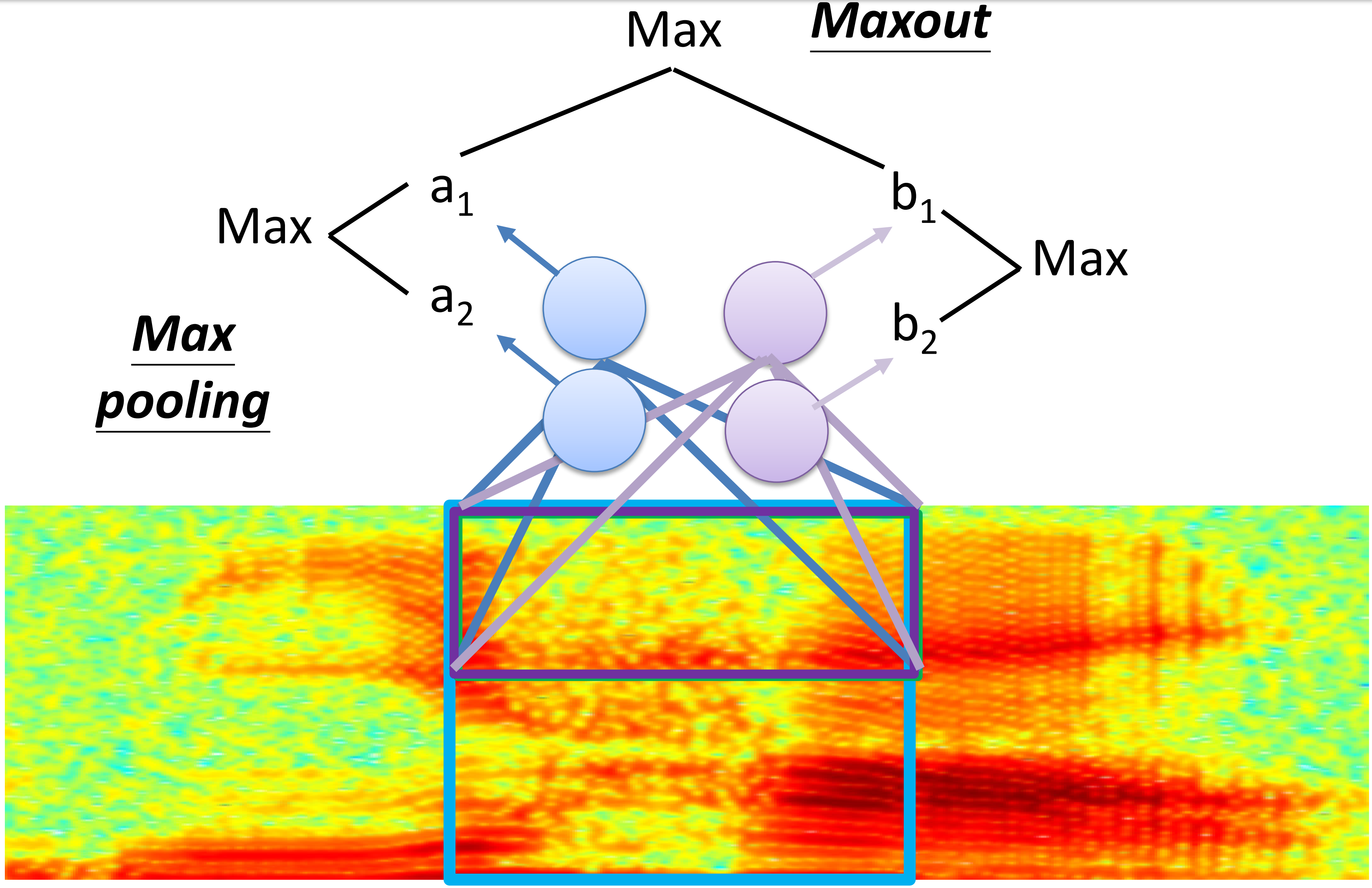


Probabilities of states

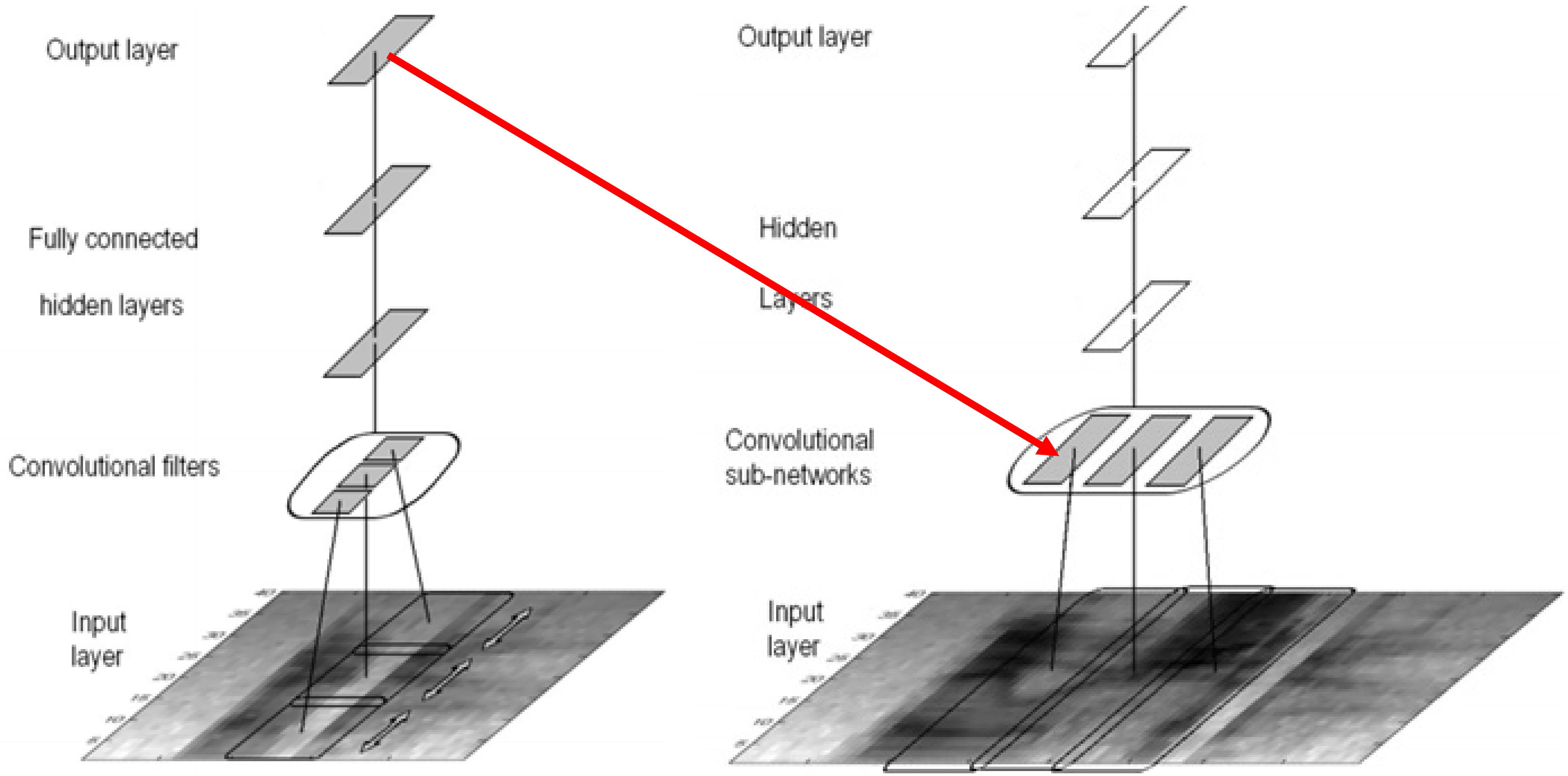
CNN

Replace DNN by CNN




Max
Maxout
a_1
a_2
Max
b_1
b_2
Max
Max pooling

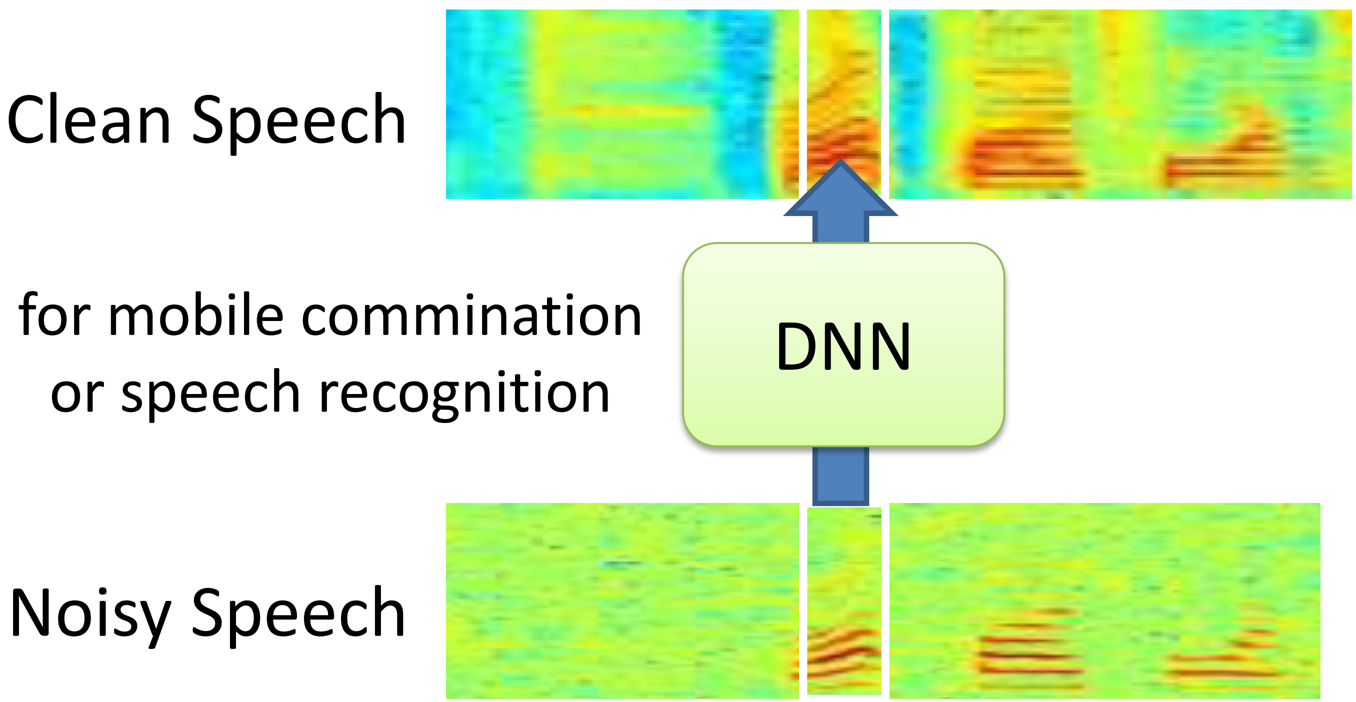
CNN



Tóth, László. "Convolutional Deep Maxout Networks for Phone Recognition”, Interspeech, 2014.

Applications in
Acoustic Signal Processing

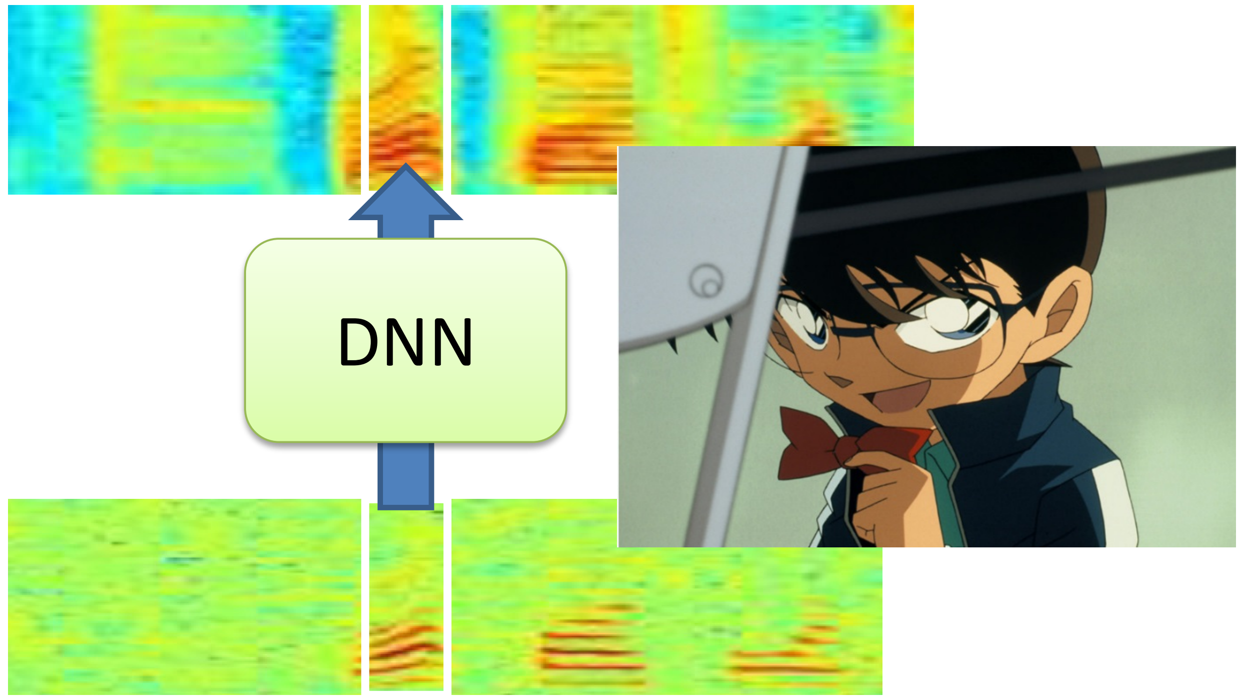
DNN for Speech Enhancement



- Demo for speech enhancement:
  http://home.ustc.edu.cn/~xuyong62/demo/SE_DNN.html

DNN for Voice Conversion

Female



Male

- Demo for Voice Conversion: https://candyvoice.com/demos/voice-conversion?lang=en

Google Assistant 電話訂餐(1/2)

- google IO 2018 assistant call DEMO1 (彩色=AI)

Google Assistant 電話訂餐(2/2)

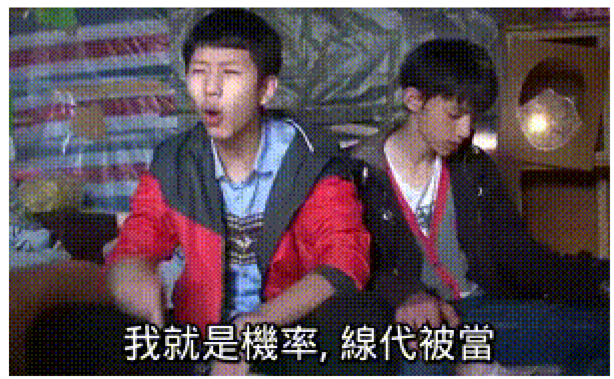
- google IO 2018 assistant call DEMO2 (彩色=AI)

Concluding Remarks

Concluding Remarks

It’s an interesting time!

- 大學的基礎科目很重要！
  - 機率統計,線性代數, Machine Learning, … etc
- Deep learning integrated into standard speech toolkits
  - Kaldi, HTK, … etc
- Rich variety of models and topologies supported by:
  - large quantities of training data
  - GPU-based training (and parallel implementations)
  - array of ML tools: TensorFlow, CNTK... etc



Questions

Thank you

To learn more about Delta, please visit www.deltaww.com.

References

[1] L. Baum and J. Eagon, "An Inequality with Applications to Statistical Estimation for Probabilistic Functions of Markov Processes and to a Model for Ecology," Bull Amer Math Soc, vol. 73, pp. 360–363, 1967.

[2] C. M. Bishop, Pattern Recognition and Machine Learning. Springer Verlag, 2006.

[3] H. Bourlard and N. Morgan, "Connectionist speech recognition: A hybrid approach," 1994.

[4] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," CoRR, vol. abs/1508.01211, 2015.

[Online]. Available: http://arxiv.org/abs/1508.01211

[5] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," CoRR, vol. abs/1506.07503, 2015.

[Online]. Available: http://arxiv.org/abs/1506.07503

[6] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint arXiv:1412.3555, 2014.

[7] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," CoRR, vol. abs/1506.02216, 2015.

[Online]. Available: http://arxiv.org/abs/1506.02216

[8] M. Gales and S. Young, "The application of hidden Markov models in speech recognition," Foundations and Trends in Signal Processing, vol. 1, no. 3, 2007.

[9] A. Graves, "Sequence transduction with recurrent neural networks," CoRR, vol. abs/1211.3711, 2012.

[Online]. Available: http://arxiv.org/abs/1211.3711

[10] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in Proceedings of the 23rd international conference on Machine learning. ACM, 2006, pp. 369–376.

[11] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in 2013 IEEE international conference on acoustics, speech and signal processing. IEEE, 2013, pp. 6645–6649.

[12] G. E. Hinton, "Products of experts," in Proceedings of the Ninth International Conference on Artificial Neural Networks (ICANN 99), 1999, pp. 1–6.

References

[13] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82–97, 2012.

[14] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[15] F. Jelinek, Statistical methods for speech recognition, ser. Language, speech, and communication. Cambridge (Mass.), London: MIT Press, 1997.

[16] H.-K. Kuo and Y. Gao, "Maximum entropy direct models for speech recognition," IEEE Transactions Audio Speech and Language Processing, 2006.

[17] L. Lu, X. Zhang, K. Cho, and S. Renals, "A study of the recurrent neural network encoder-decoder for large vocabulary speech recognition," in Proc. INTERSPEECH, 2015.

[18] T. Robinson and F. Fallside, "A recurrent error propagation network speech recognition system," Computer Speech & Language, vol. 5, no. 3, pp. 259–274, 1991.

[19] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1," D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, ch. Learning Internal Representations by Error Propagation, pp. 318–362.

[20] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015, pp. 4580–4584.

[21] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997.

[22] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," CoRR, vol. abs/1505.00387, 2015.

[Online]. Available: http://arxiv.org/abs/1505.00387

[23] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," vol. 13, no. 2, pp. 260–269, 1967.

[24] Research Talk by Mark Gales, in LXMLS, 2017. http://lxmls.it.pt/2017/talk.pdf

[25] NTU Course by Hung-Yee Lee (ppt)

[26] Standford Course: Speech Processing https://web.stanford.edu/class/cs224s/lectures/

Research Road Map

- Speech-to-text (Automatic Speech Recognition)
- Text -to-speech (Speech Synthesis)
- Speaker Recognition & Diarization
- Voice Conversion
- Speech Separation
- Speech Enhancement
- Speech Emotion Recognition
- Music Information Retrieval
- Spoken Language Understanding
- …