# 4 Speech Recognition

# Chapter 4 Speech Recognition

- 4.1 Overview of Speech Recognition
- 4.2 Acoustic Model
- 4.3 Language Model
- 4.4 Decoding Algorithm for Speech Recognition
- 4.5 Prospect of Speech Recognition

# 4.1 Overview of Speech Recognition

- 4.1.1 Basic Concepts of Speech Recognition
- 4.1.2 History and Main Contents of Speech Recognition

# What is speech recognition?

## Speech Recognition

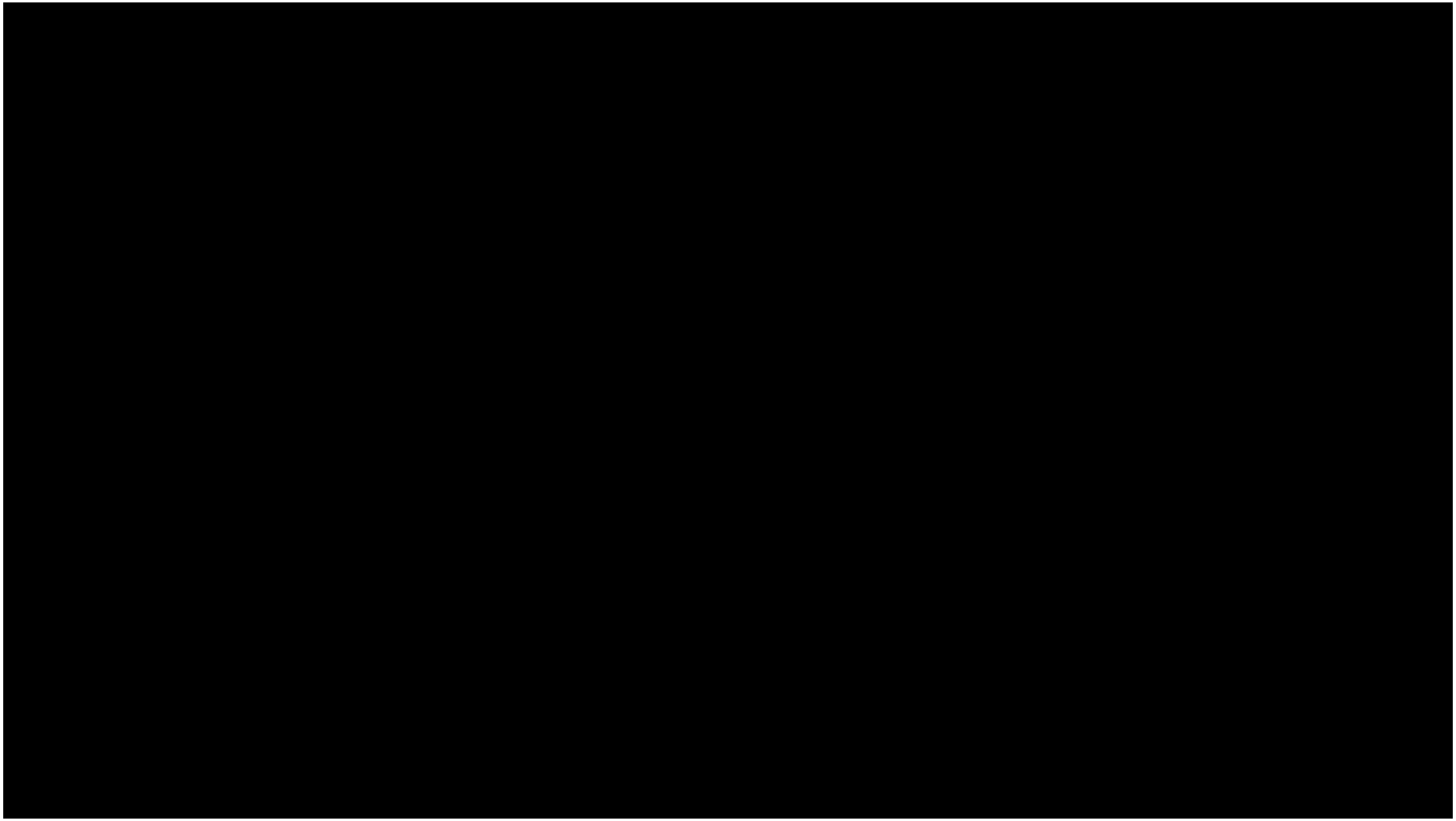
Convertion of spoken words into text (words, syllables, phones, etc.), also known as “Automatic Speech Recognition” (ASR).

ASR System

“How are you?”

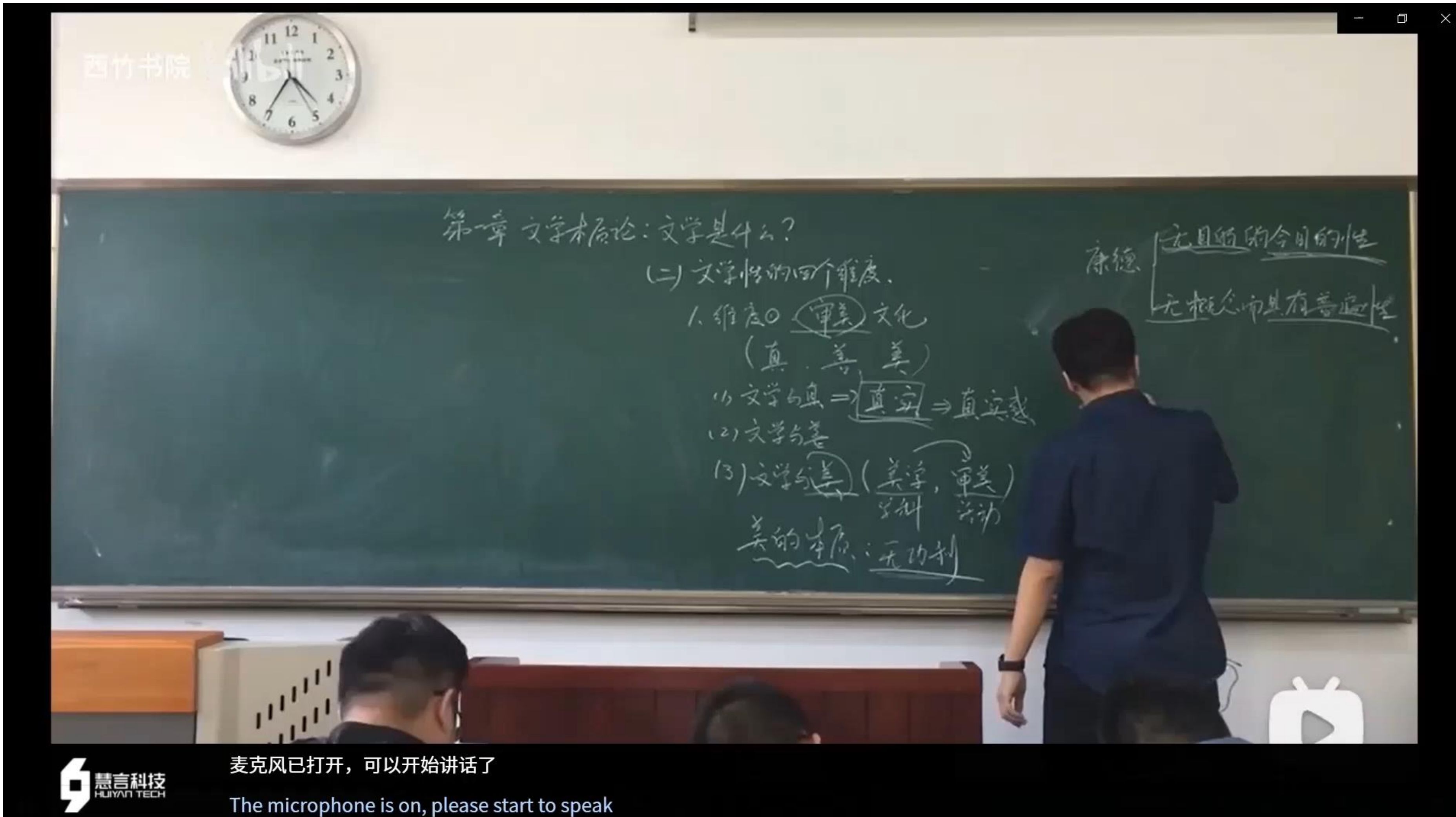
# What is speech recognition?

The recognized text can be

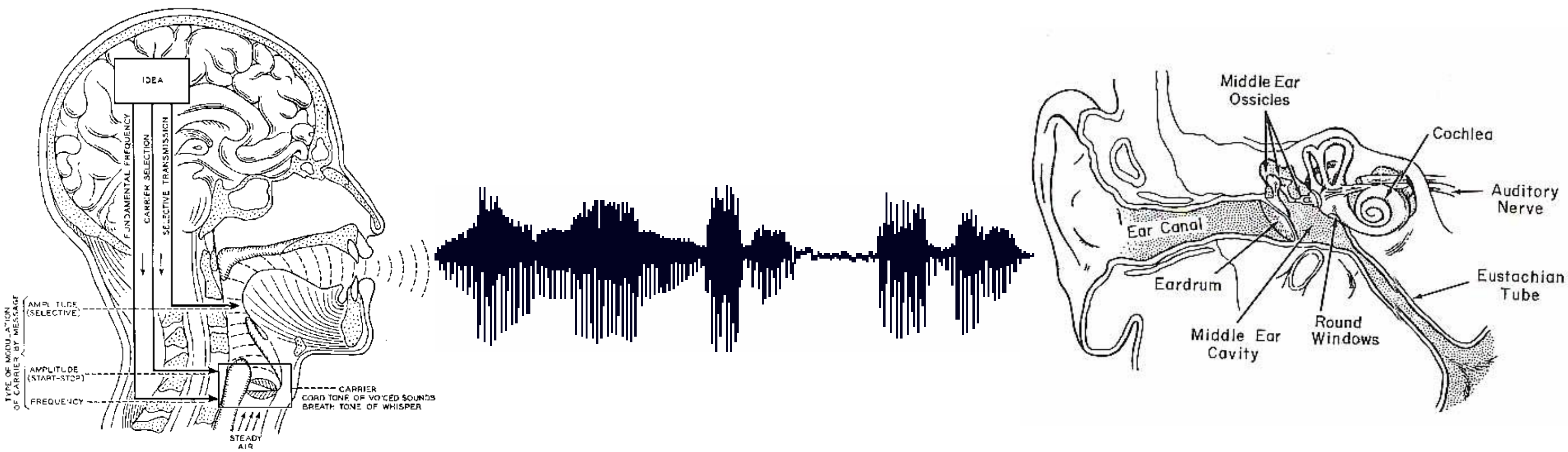


# What is speech recognition?

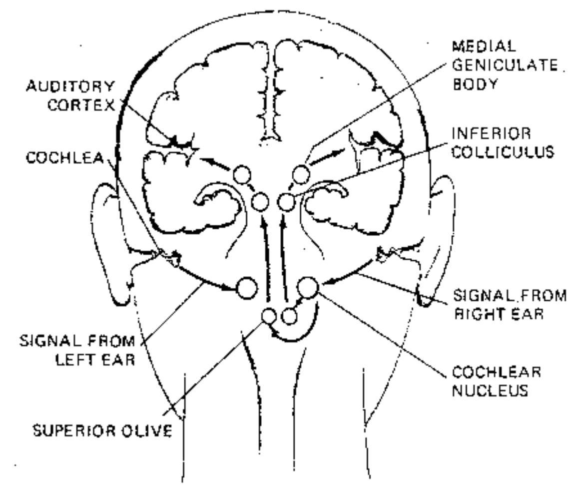
The recognized text can be



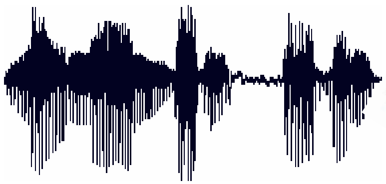
# How do humans recognize speech?




- Articulation produces sound waves which the ear conveys to the brain for processing.

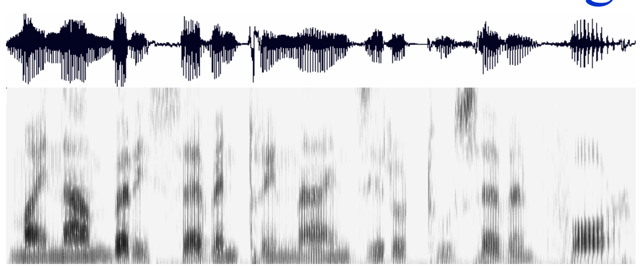


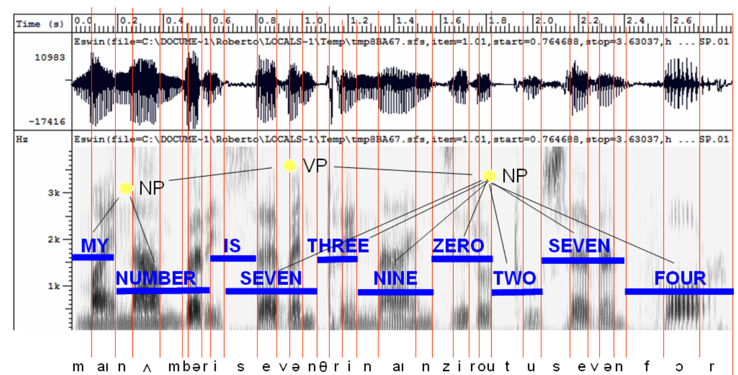
# How might computers do it?



Acoustic waveform

Acoustic signal






- Digitization
- Acoustic analysis of the speech signal
- Linguistic interpretation

**Speech recognition**

# Types of Speech Recognition

- Different classes based on types of utterances they are able to recognize

  - 1. Isolated Words
      - “Listen/Not-Listen” states

“打开空调”

  - 2. Connected Words
      - Connected digit recognition

“139-1234-5678”(Phone Number)

      - “run-together”

“现在 我 简单 介绍 语音 识别 的 基本 概念。”

  - 3. Continuous Speech
      - Natural speech

Large Vocabulary Continuous Speech Recognition (LVCSR)

“现在我简单介绍语音识别的基本概念。”

# Types of Speech Recognition

- Different classes based on different vocabulary size
  - 1. Small vocabulary
    - About 10-1000 words
  - 2. Medium vocabulary
    - About 1000-10000 words
  - 3. Large vocabulary
    - More than 10000 words

Large Vocabulary Continuous Speech Recognition (LVCSR)

# Types of Speech Recognition

- Different classes based on speaker dependency

  - 1. Speaker dependent
    - Training/Adapting model using user’s speech
    - Time consuming for individual speaker/user.

  - 2. Speaker independent
    - Using same model for all speakers/users
    - Need various speakers and styles for training data

# ASR Metric: Word Error Rate (WER)

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

$$\text{Word Error Rate} = \frac{\text{Insertions} + \text{Substitutions} + \text{Deletions}}{\text{Total Words in Correct Transcript}} * 100\%$$

Alignment example:

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

WER = (1+2+0)/6 *100%= 50%

# 4.1 Overview of Speech Recognition

- 4.1.1 Basic Concepts of Speech Recognition
- 4.1.2 History and Main Contents of Speech Recognition

# Trends of ASR

## Modeling approaches:

| Before mid 70's | Mid 70's – mid 80's | After mid 80's |
|---|---|---|
| Heuristic | Template matching | Mathematical |
| Rule-based and declarative | Deterministic and data-driven | Probabilistic and data-driven |

**Pattern Recognition Approach**

| | | |
|---|---|---|
| **Isolated word** | **Isolated word**<br>**Connected speech** | **LVCSR**<br>HMM-based<br>DL-based |

# Rule-based approach

- Use knowledge of phonetics（语音学） and linguistics （语言学）  to guide search process

- Rules express everything (anything) that might help to decode:
  - Phonetics, phonology（音韵学）, phonotactics（音位结构学）
  - Syntax（语法）
  - Pragmatics（语用学）

# Rule-based approach

- Typical approach is based on “blackboard” architecture:
  - At each decision point, lay out the possibilities
  - Apply rules to determine which sequences are permitted

- Poor performance due to
  - Difficulty to express rules
  - Difficulty to make rules interact
  - Difficulty to know how to improve the system

| | | | |
|---|---|---|---|
| | k | iː | ʃ |
| s | h | iə | tʃ |
| ʃ | p | ɪ | h |
| | t | | s |

# Pattern Recognition Approach

- 2 steps:
  - Pattern Training
  - Pattern Comparison

- Forms:
  - Speech Template — Deterministic
  - Statistical Model (HMM, DNN) — Probabilistic

- Goal to determine identity of unknown speech according to how well patterns match

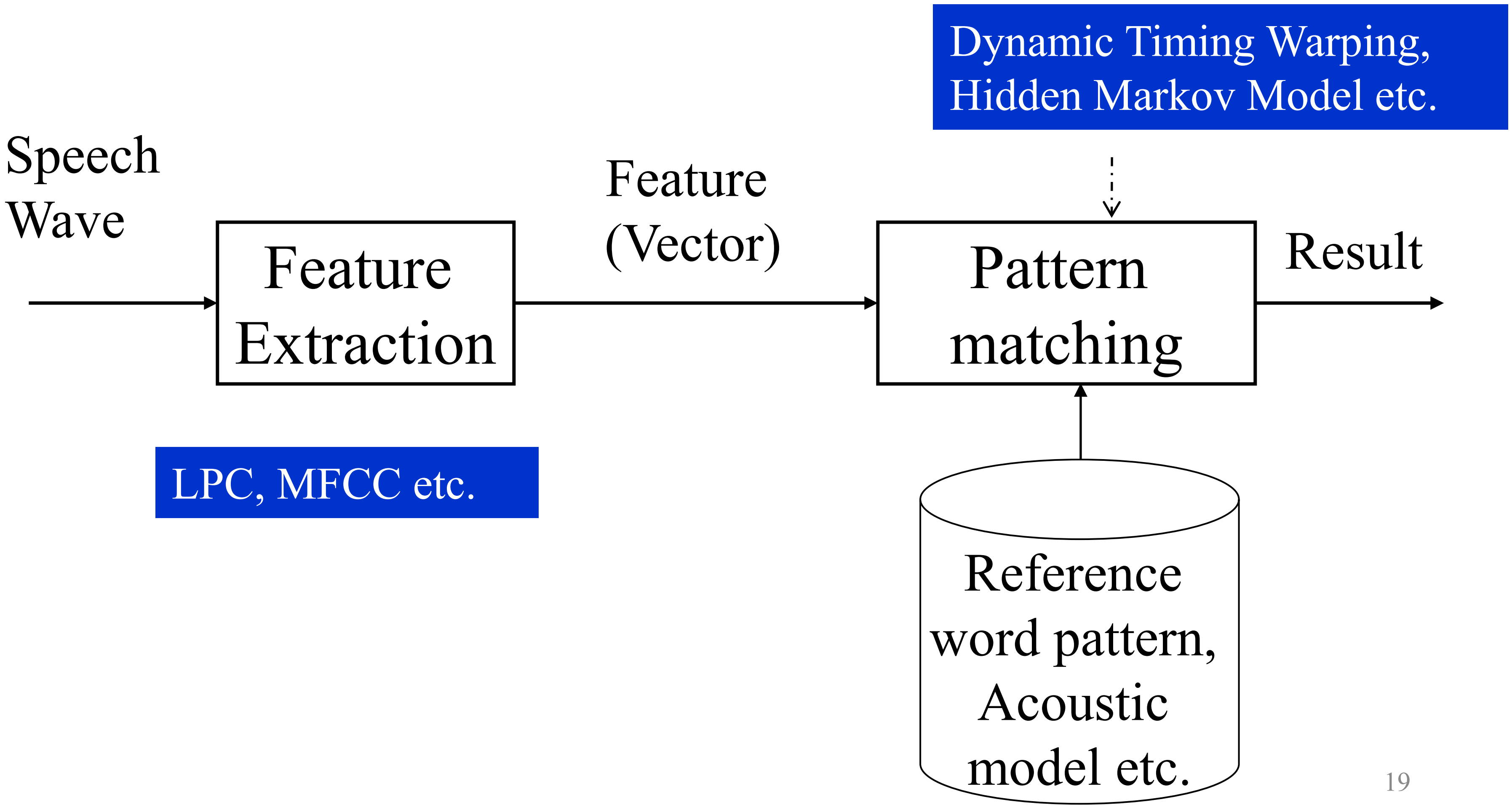
# Methods in Pattern Comparison Approach

- Template Based Approach
  - Patterns stored as dictionary of words
  - Match unknown utterance with reference templates
  - Select best matching pattern

- Statistical Approach (HMM、DNN)
  - Probabilistic Models
  - Uncertainty and Incompleteness

# Diagram of data-driven ASR

Example of isolated word recognition



# Template-based approach

# Template-based approach

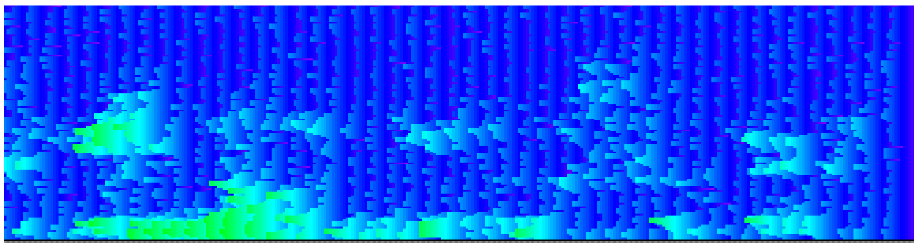
- Store examples of units (words, phonemes), then find the example that most closely fits the input

- Extract features from speech signal, then it's "just" a complex similarity matching problem, using solutions developed for all sorts of applications

- OK for discrete utterances, and a single user

# Pattern matching with different length

Input pattern

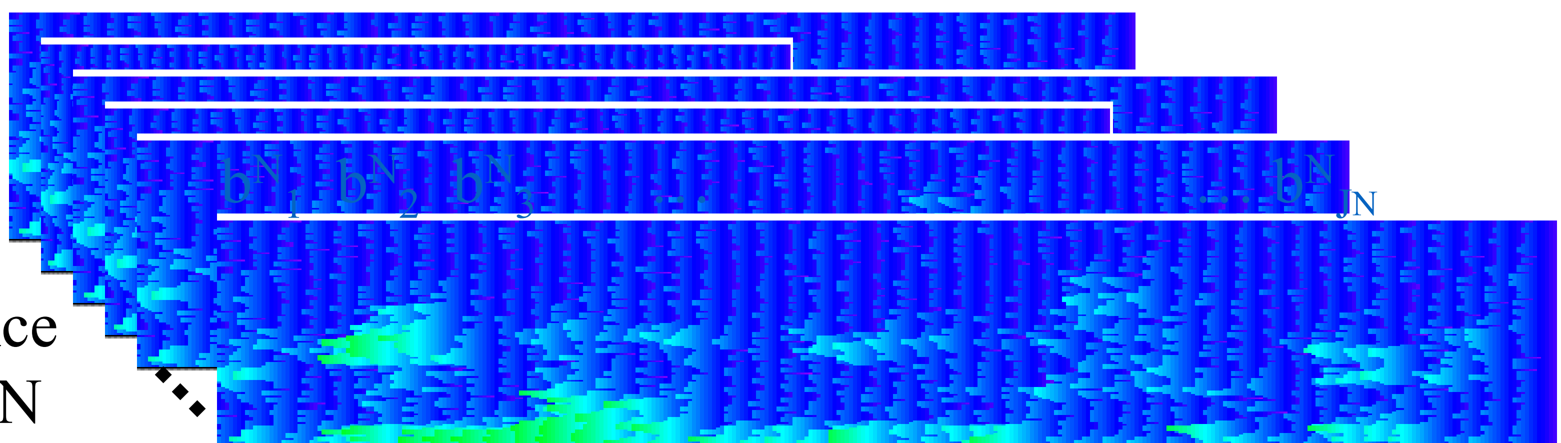



Different length

↓

How to compare?

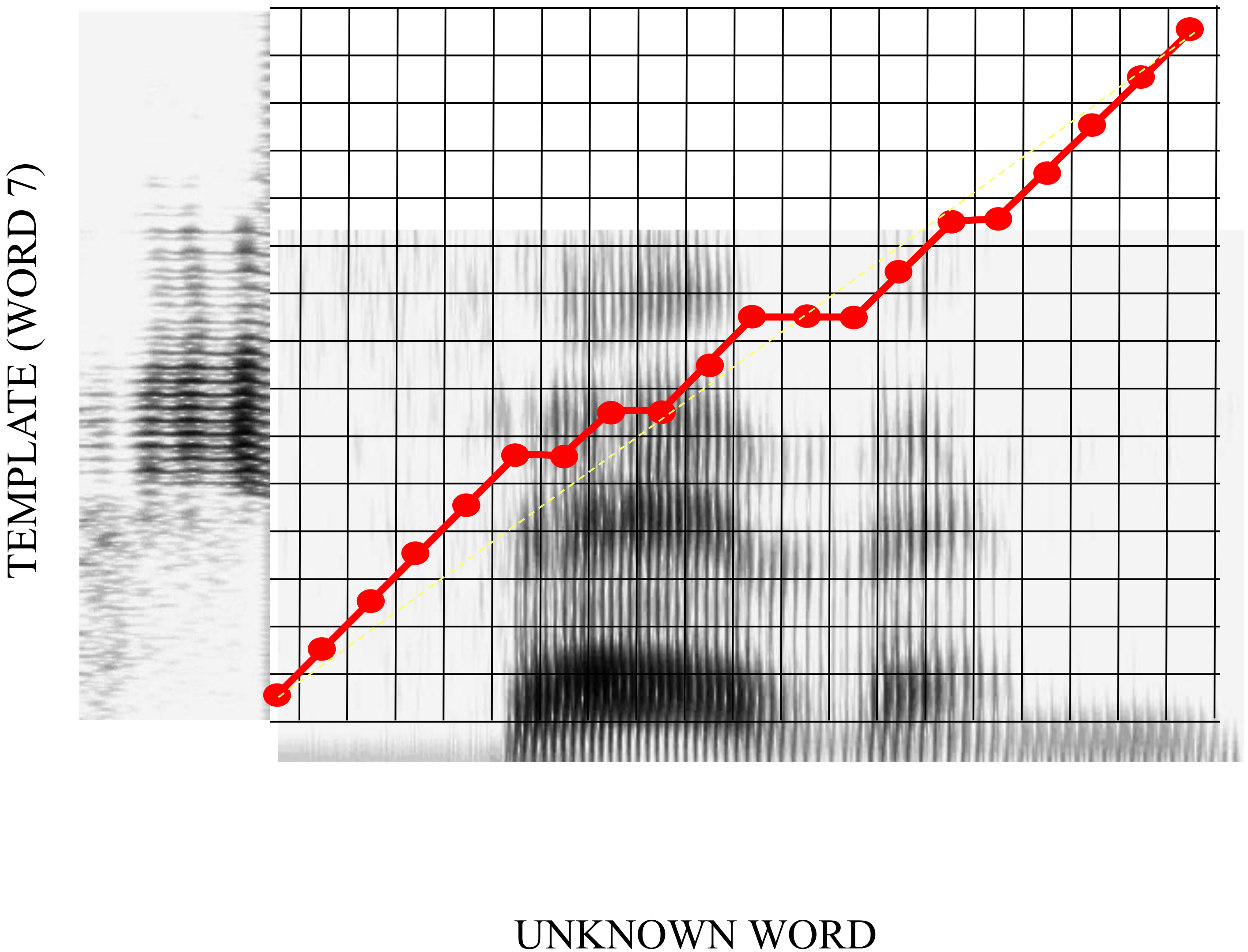
How about linear alignment?




Reference Pattern 1

Reference Pattern N

# Dynamic Time Warping (DTW) /Dynamic Programming (DP) Matching

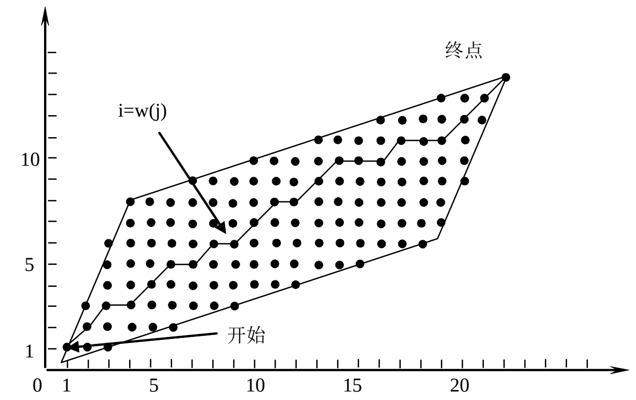


# DTW Algorithm

## (1) Initialization:

$$i = j = 1, \quad g(1, \quad 1) = 2d(\boldsymbol{x_i}, \boldsymbol{y_j})$$

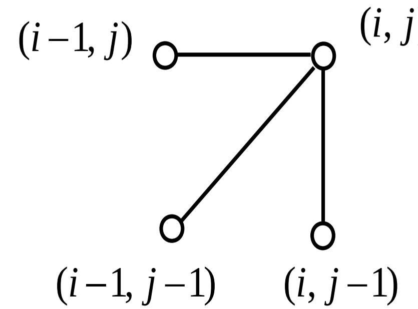
$$g(i,j) = \begin{cases} 0 & (i,j) \in \mathrm{Reg} \\ huge & (i,j) \notin \mathrm{Reg} \end{cases}$$




Constrained Region

**g(i,j):** minimum partial accumulated distance

# DTW Algorithm




## (2) Recursion:

Local distance

Path constraints

Minimum partial
accumulated distance

$$\mathrm{g}(i,j) = \min \begin{Bmatrix} g(i-1,j) + d\left(\boldsymbol{x_i}, \boldsymbol{y_j}\right) W(1) \\ g(i-1,j-1) + d\left(\boldsymbol{x_i}, \boldsymbol{y_j}\right) W(2) \\ g(i,j-1) + d\left(\boldsymbol{x_i}, \boldsymbol{y_j}\right) W(3) \end{Bmatrix}$$

$$i = 2,3,\cdots,I; j = 2,3,\cdots,J; (i,j) \in \mathrm{Reg}$$

$$W(1) = W(3) = 1, \quad W(2) = 2$$

## (3) Termination:

$$\frac{g(I,J)}{\sum W}$$

# Problem of template-based approach

- Hard to distinguish very similar templates
- And quickly degrades when acoustic conditions between input and templates are different
- Therefore needs techniques to mitigate these degradations:
  - More subtle matching techniques
  - Multiple templates which are aggregated
- Taken together, these suggested …

# Statistic-based approach

# Statistic-based approach

- Can be seen as extension of template-based approach, using more powerful mathematical and statistical models

- Sometimes seen as “anti-linguistic” approach

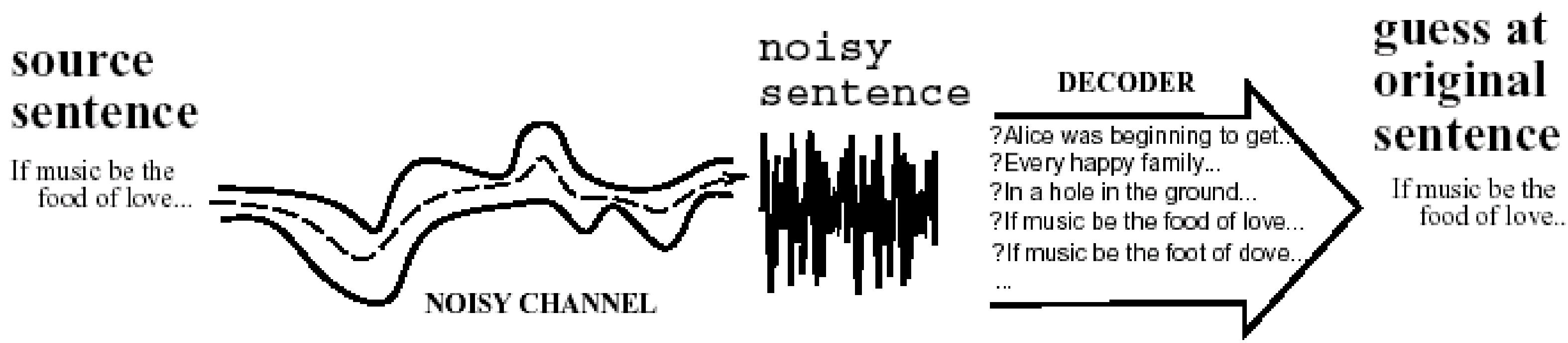
# Statistic-based approach

- Collect a large corpus of transcribed speech recordings

- Train the computer to learn the correspondences ("machine learning")

- At run time, apply statistical processes to search through the space of all possible solutions, and pick the statistically most likely one

# Noisy Channel in a Picture




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

# Noisy Channel Model

- In speech recognition, you observe an acoustic signal ($A=a_1,\ldots,a_n$) and you want to determine the most likely sequence of words ($W=w_1,\ldots,w_n$): $P(W \mid A)$

- Assume that the acoustic signal (A) is already segmented
- $P(W \mid A)$ could be computed as

$$P(W \mid A) = \prod_{a_i} \max_{w_i} P(w_i \mid a_i)$$

- Problem: Finding the most likely word corresponding to an acoustic representation depends on the context

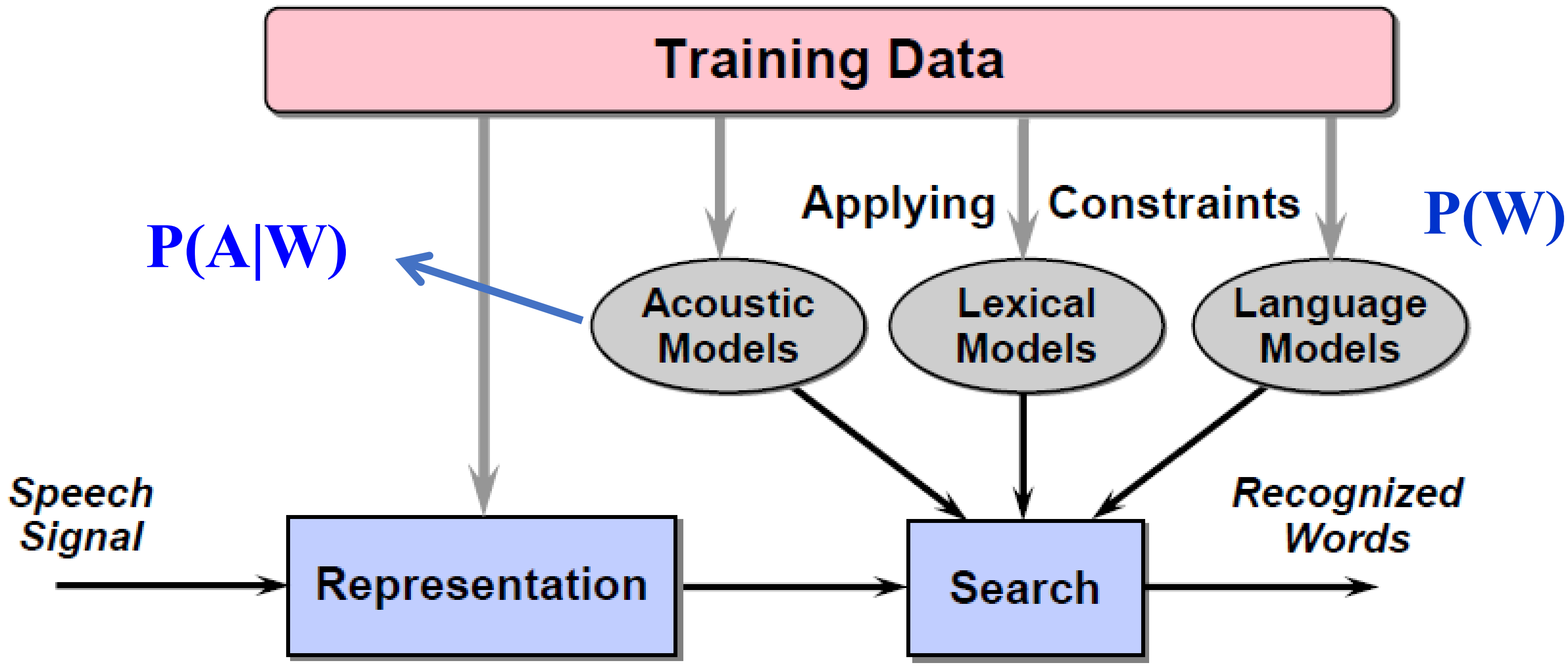
# Noisy Channel Model

- Given a candidate sequence W we need to compute P(W) and combine it with P(W | A)
- Applying Bayes' rule:

$$\arg\max_W P(W \mid A) = \arg\max_W \frac{P(A \mid W)P(W)}{P(A)}$$

- The denominator P(A) can be dropped, because it is constant for all W

# Diagram of ASR system

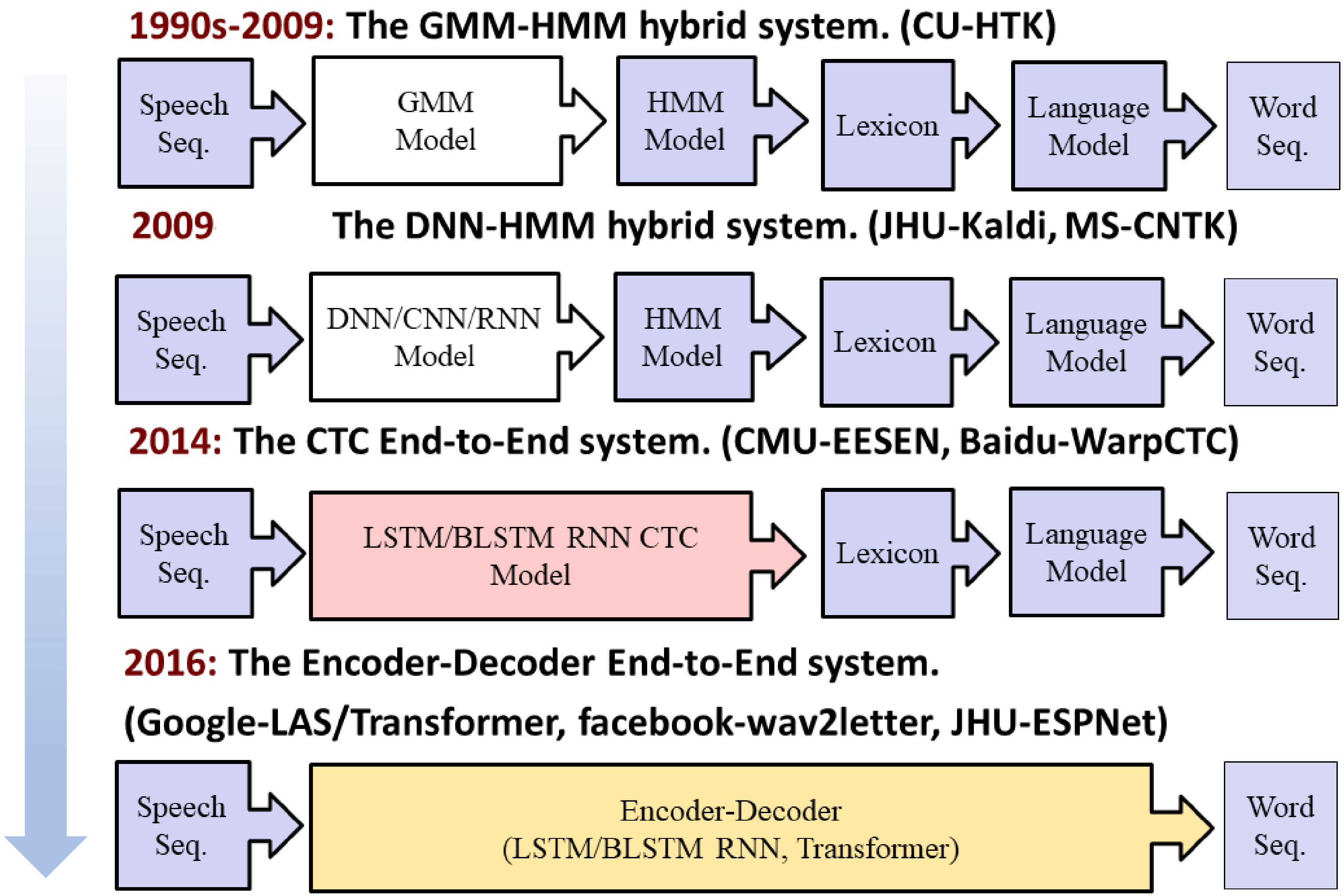



Feature extraction

Front-end processing

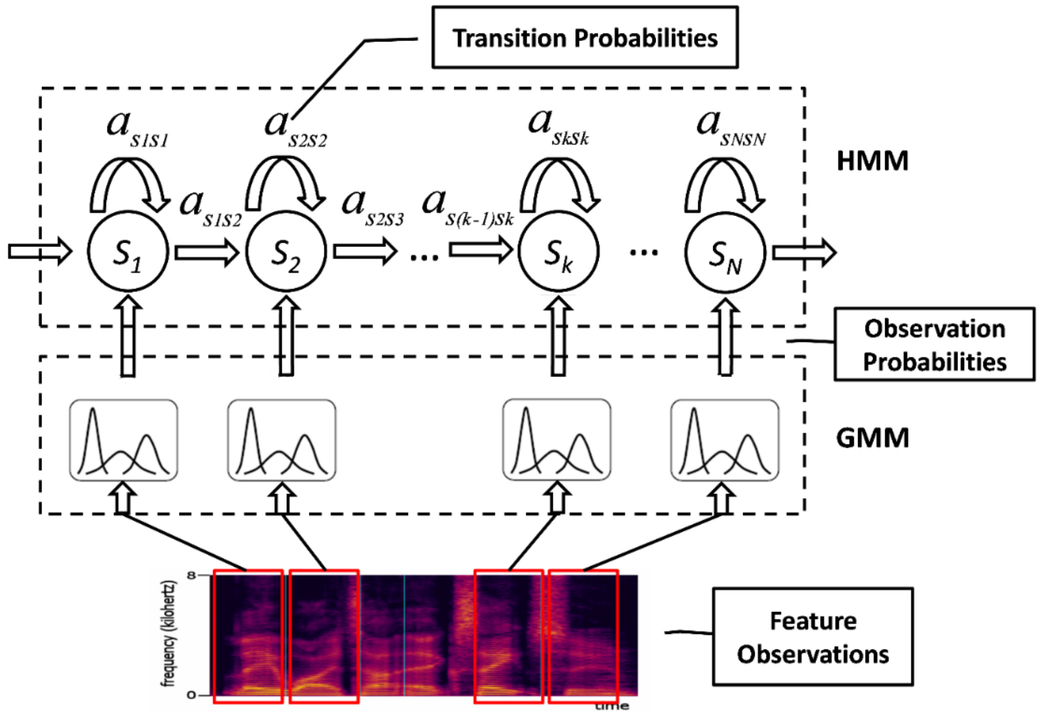
Back-end processing

# Evolution of speech recognition systems (Statistic model-based)



# Acoustic Model

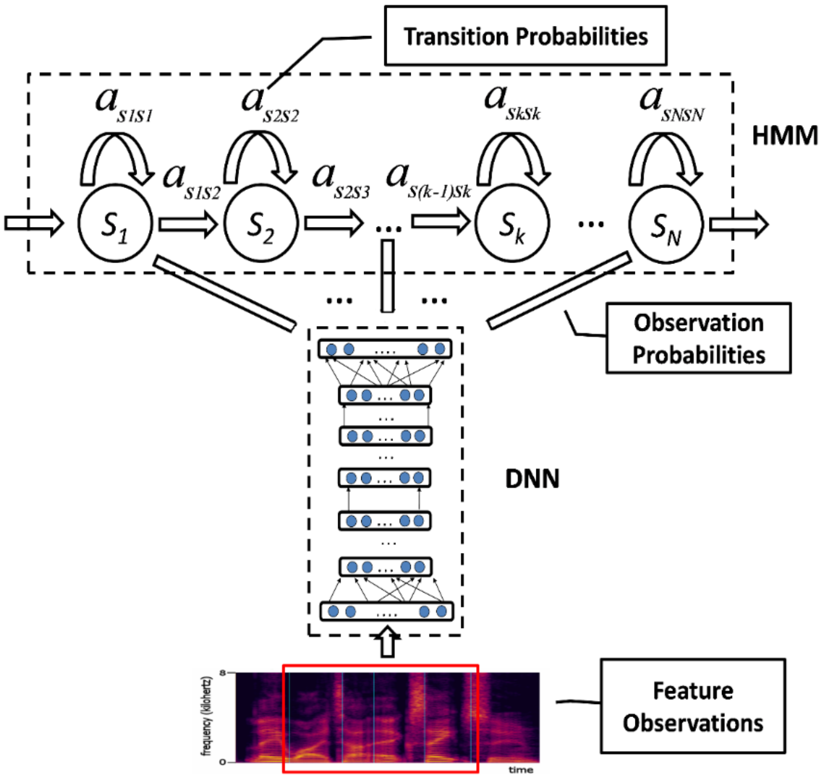
- GMM-HMM




- L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. Proc. IEEE, 77(2):257–286, 1989.

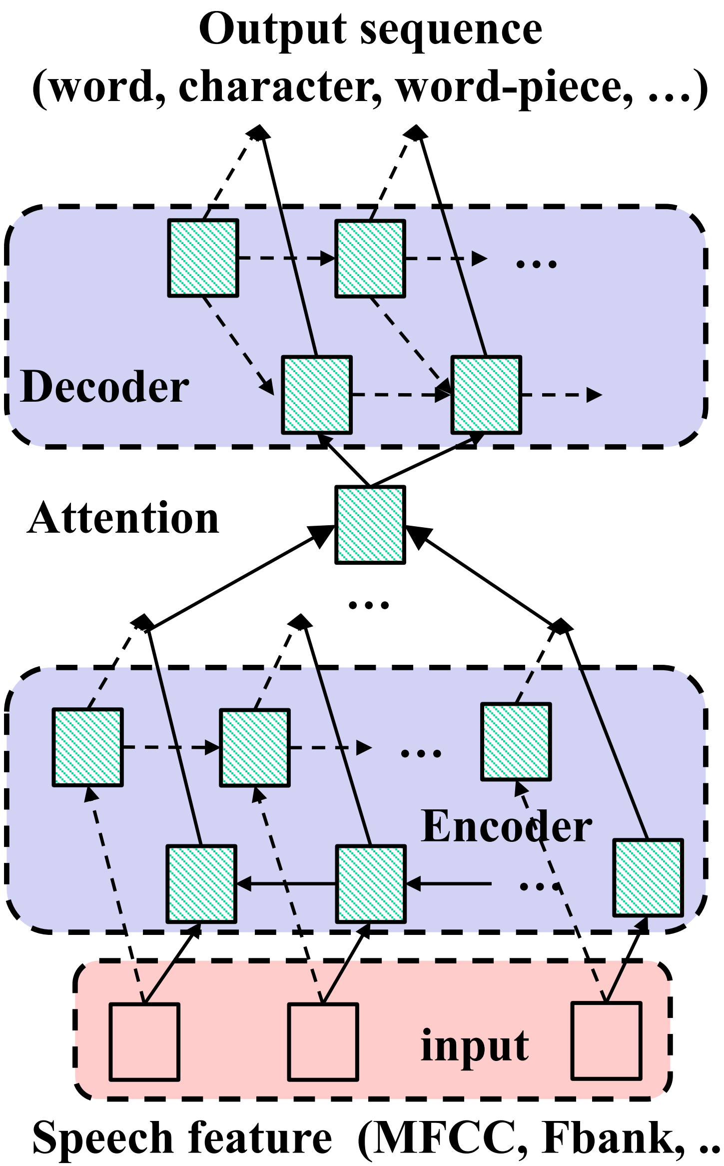
# Acoustic Model

- DNN-HMM



# End-to-End Model

- Jointly learn acoustic model and language model



# Language model

- Finite state automaton grammar

- N-gram models: (Models likelihood of word given previous word(s))
  - Build the model by calculating bigram or trigram probabilities from text training corpus
  - Smoothing issues

- Recurrent Neural Network-based model

# Decoding Algorithm

- Calculation of best hypothesis of word sequence

$$P(W \mid Y) = \underset{\{w_1^N\}\{t_1^N\}}{\arg\max}\{\sum_{n-1}^{N} \log(P_{acoust}(y_{t_{n-1}+1}^{t_n} \mid w_n)) + \lambda \cdot \sum_{n-1}^{N} \log(P_{lang}(w_n \mid w_{n-1}) + \delta)\}$$

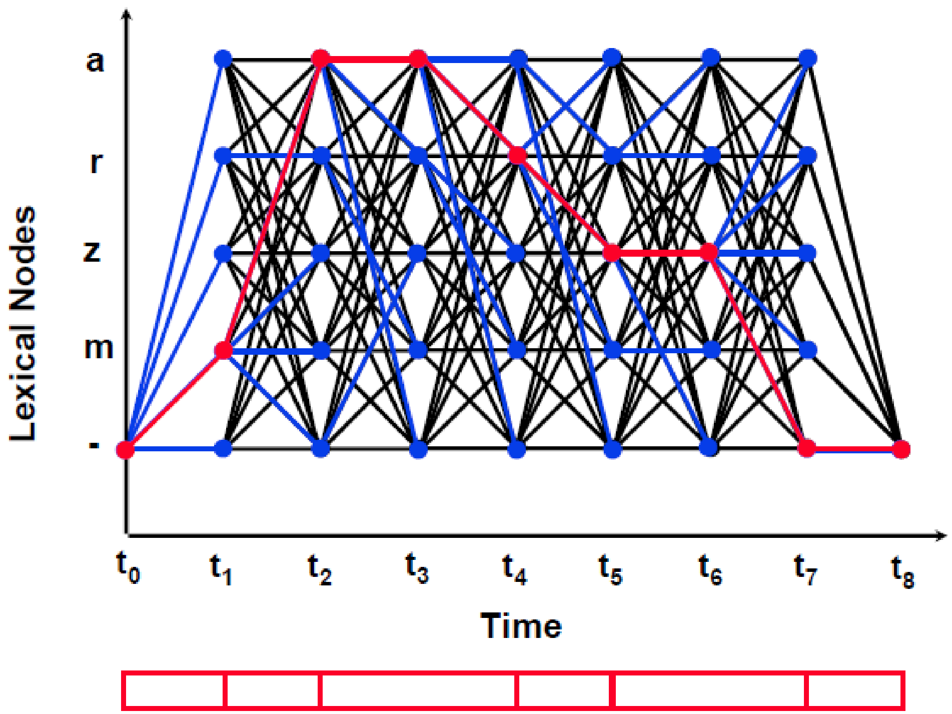
$P_{acoust}(y_{t_{n-1}+1}^{t_n} \mid w_n)$ ： likelihood of AM

$P_{lang}(w_n \mid w_{n-1})$ ： likelihood of LM

$\lambda$ ： Weighting $\quad \delta$ ： Penalty of insertion

# Decoding Algorithm

- Viterbi Search




- WFST (Weighted finite state transducer)