Introduction to Speech Recognition

YUAN-FU LIAO

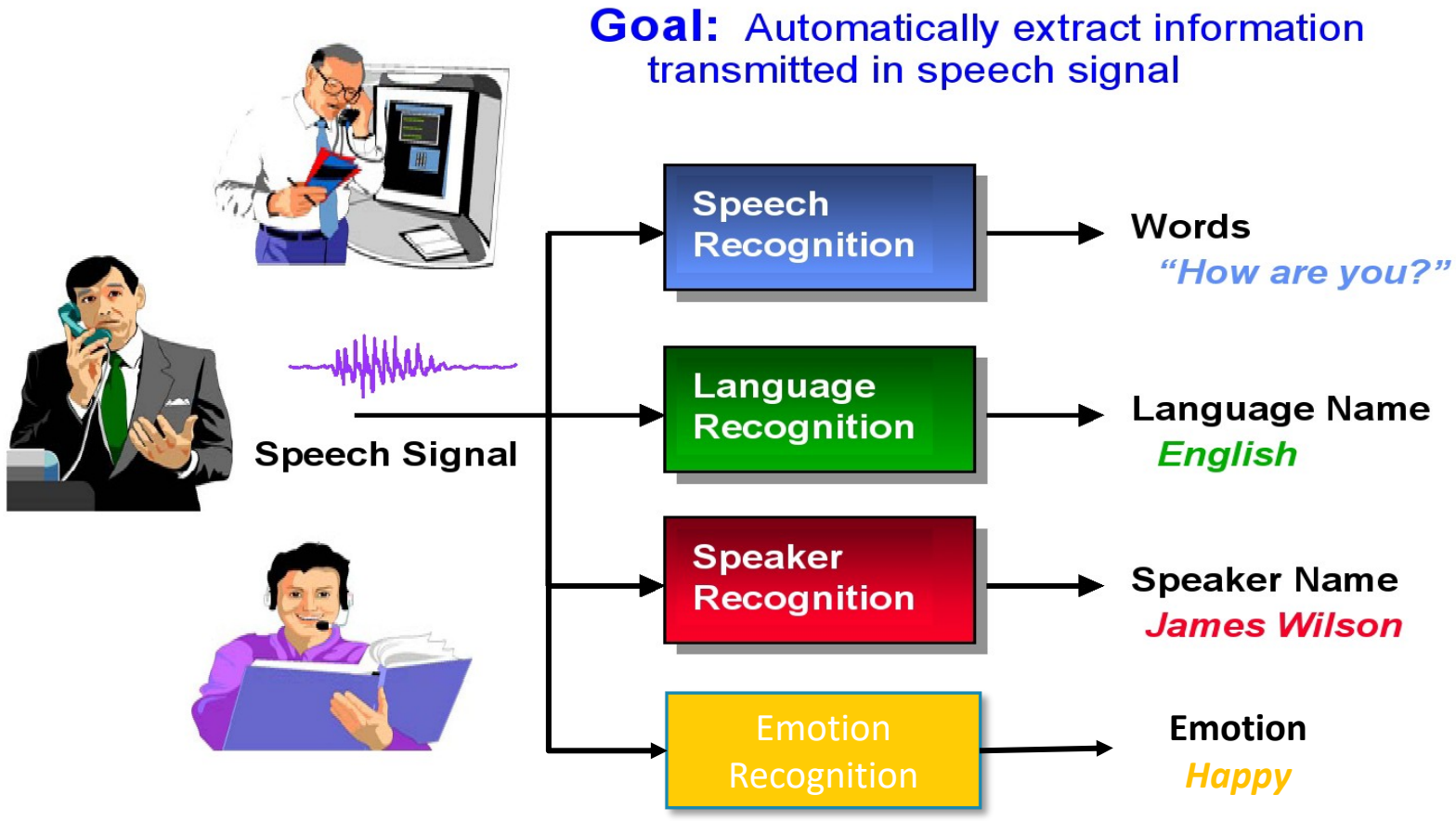
NATIONAL TAIPEI UNIVERSITY OF TECHNOLOGY

Outline

Automatic Speech Recognition (ASR)

- How does ASR work?
- General Architecture of Speech Recognition

Broad Objectives of Speech Recognition for Machines



Source : Reynolds et. al

Communication: A Thing Always Happens in Our Daily Life

What is Behind the Communication?

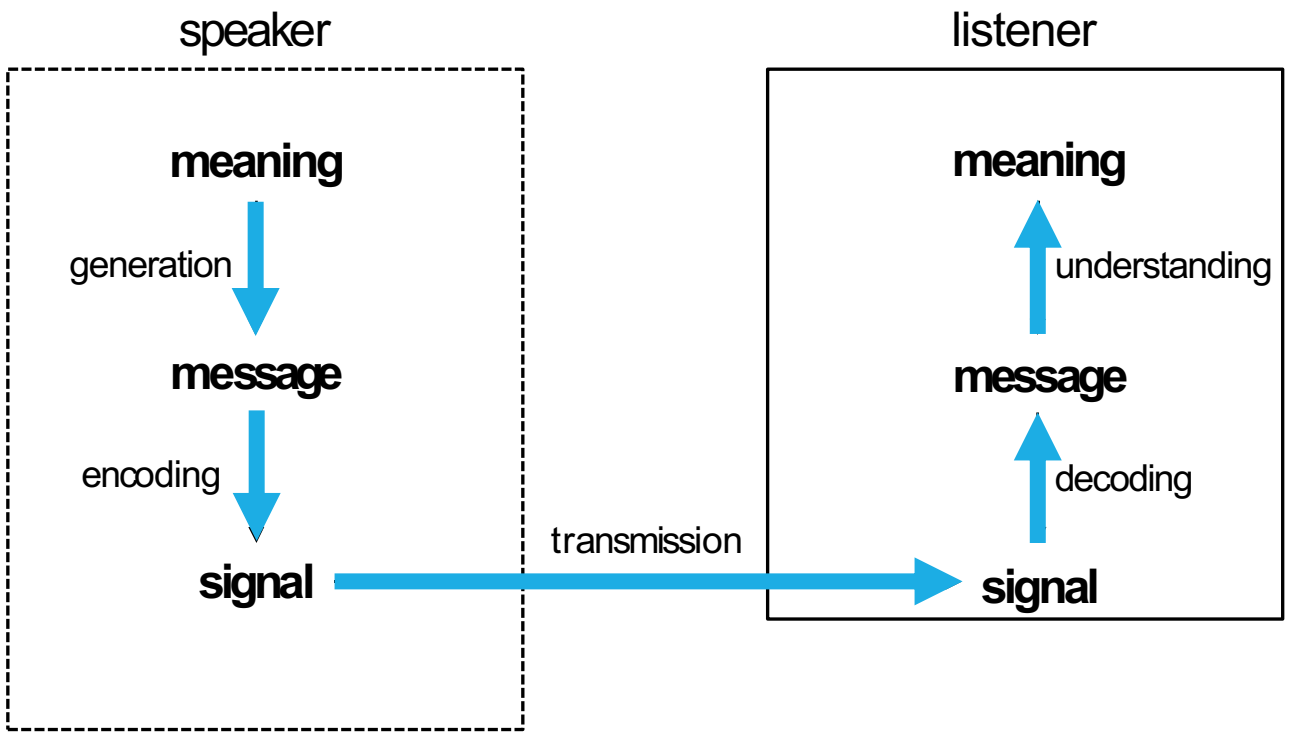
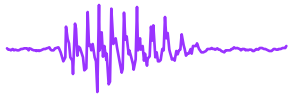


Figure. Processes involved in communication between two speakers

Speech Recognition

Goal: Automatically extract the string of words spoken from the speech signal



Speech Signal

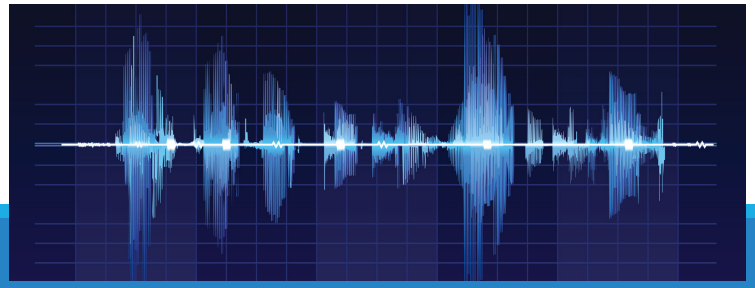
Speech Recognition

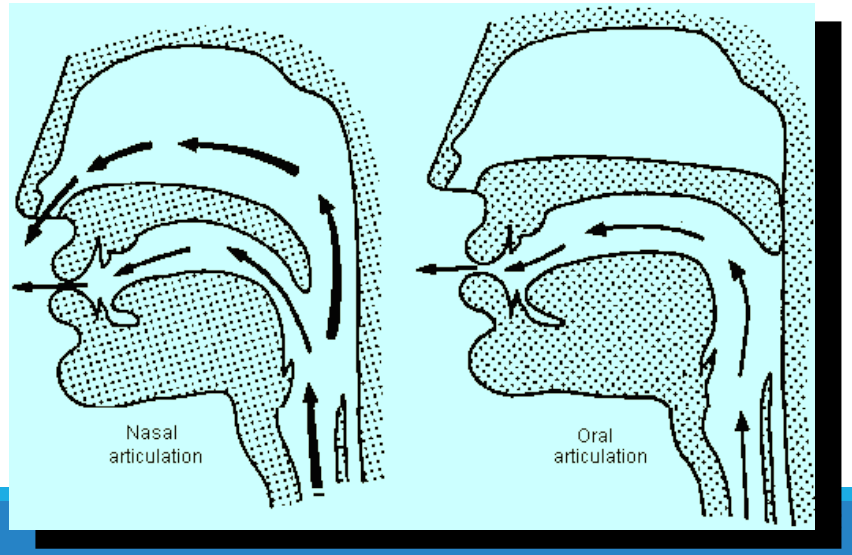
Words

"How are you?"

How is SPEECH produced?

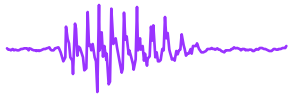
⇒ Characteristics of Acoustic Signal





Speech Recognition

Goal: Automatically extract the string of words spoken from the speech signal



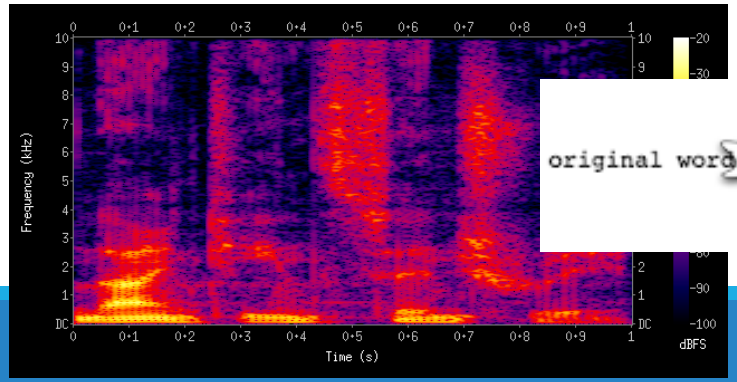
Speech Signal

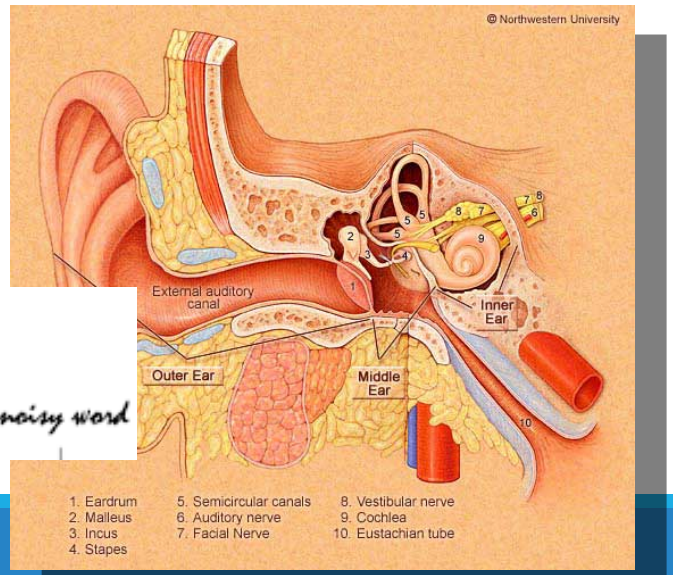
Speech Recognition

Words

"How are you?"

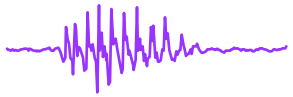
How is SPEECH perceived?
=> Important Features





Speech Recognition

Goal: Automatically extract the string of words spoken from the speech signal



Speech Signal

Speech Recognition

Words

"How are you?"

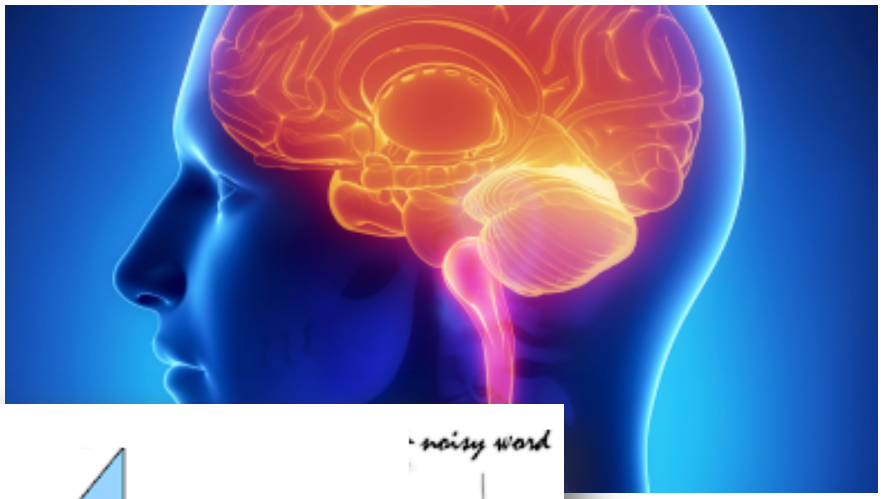
What Sentence is Spoken? => Language Model

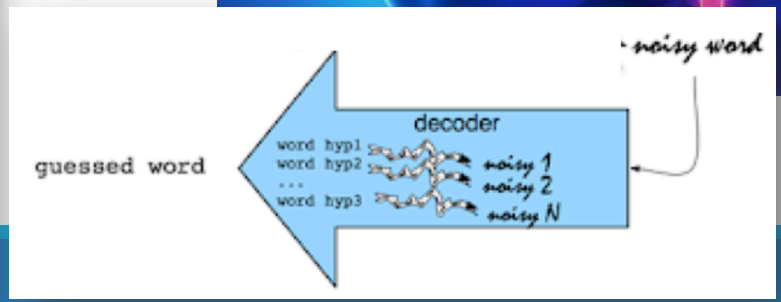
Alice was beginning to get very tired of sitting by her sister on the bank, and of having nothing to do: once or twice she had peeped into the book her sister was reading, but it had no pictures or conversations in it, 'and what is the use of a book,' thought Alice 'without pictures or conversation?'

$P(w_{i+1}=\text{of} \mid w_i=\text{tired}) = 1$
$P(w_{i+1}=\text{of} \mid w_i=\text{use}) = 1$
$P(w_{i+1}=\text{sister} \mid w_i=\text{her}) = 1$
$P(w_{i+1}=\text{beginning} \mid w_i=\text{was}) = 1/2$
$P(w_{i+1}=\text{reading} \mid w_i=\text{was}) = 1/2$

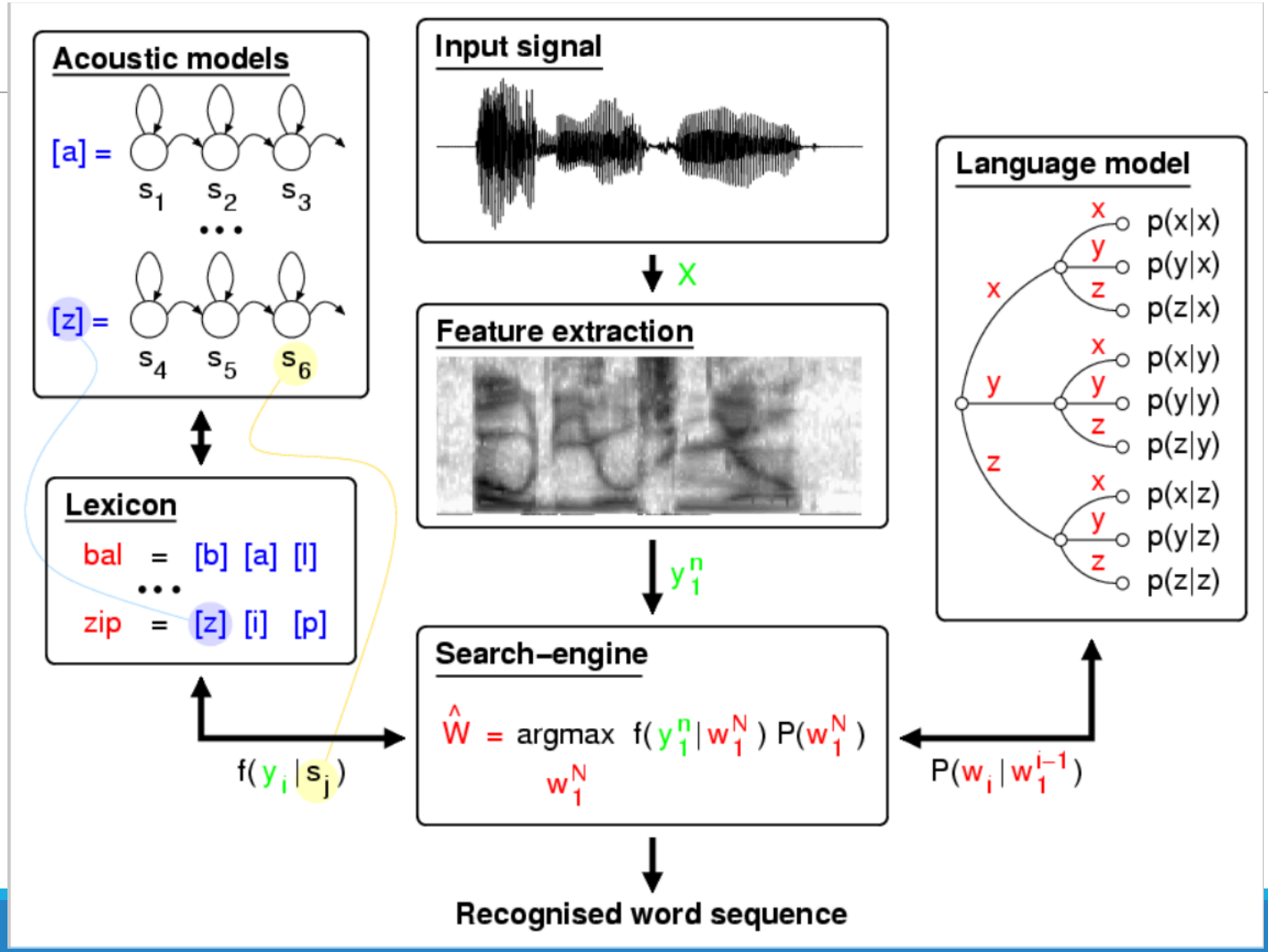
$P(w_{i+1}=\text{bank} \mid w_i=\text{the}) = 1/3$
$P(w_{i+1}=\text{book} \mid w_i=\text{the}) = 1/3$
$P(w_{i+1}=\text{use} \mid w_i=\text{the}) = 1/3$

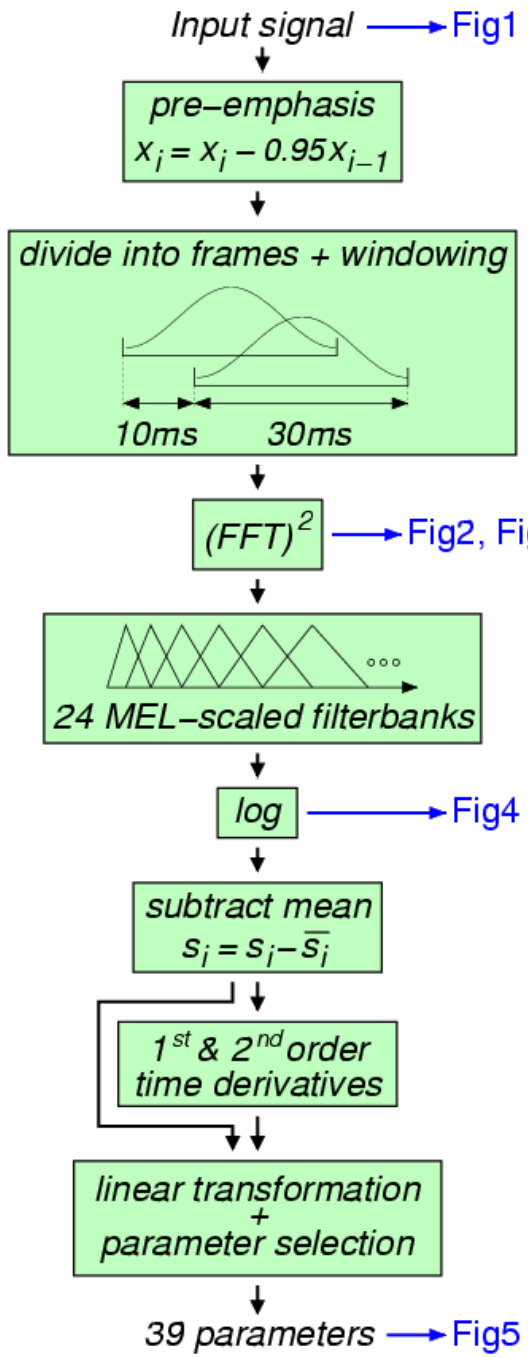
$$P(X|Y) = \frac{P(X,Y)}{P(Y)}$$

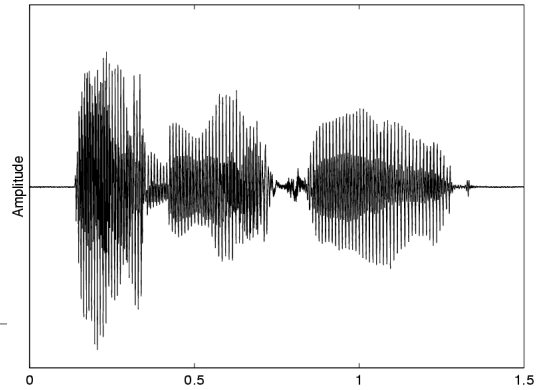


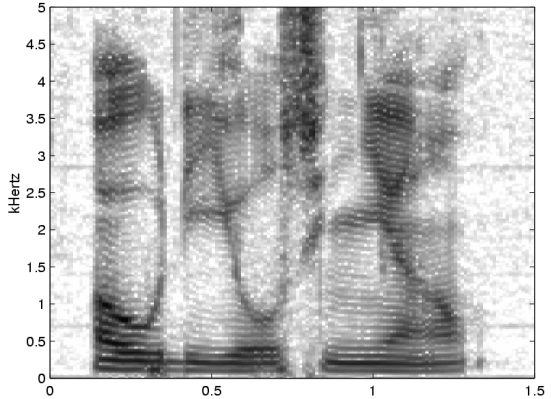


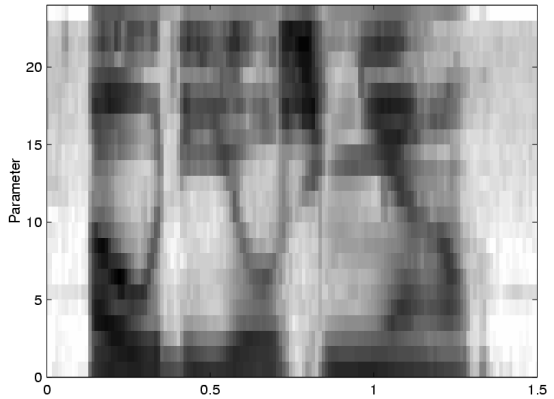
Automatic Speech Recognition: General Architecture

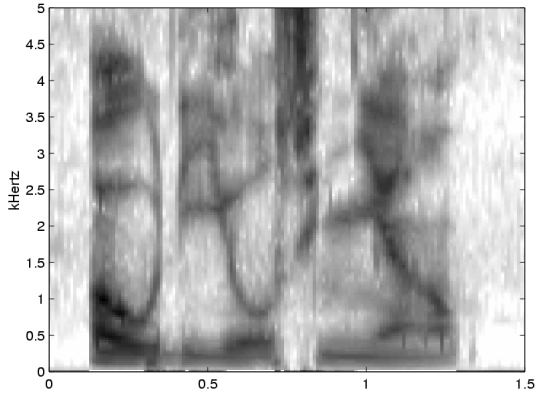


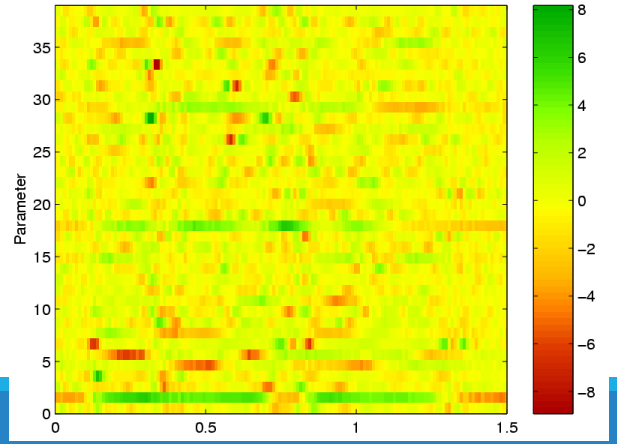

Input signal
Fig1
pre–emphasis
$x_i = x_i - 0.95x_{i-1}$
divide into frames + windowing
10ms
30ms
$(FFT)^2$
Fig2, Fig3
24 MEL–scaled filterbanks
log
Fig4
subtract mean
$s_i = s_i - \bar{s}_i$
1st & 2nd order time derivatives
linear transformation + parameter selection
39 parameters
Fig5


Amplitude

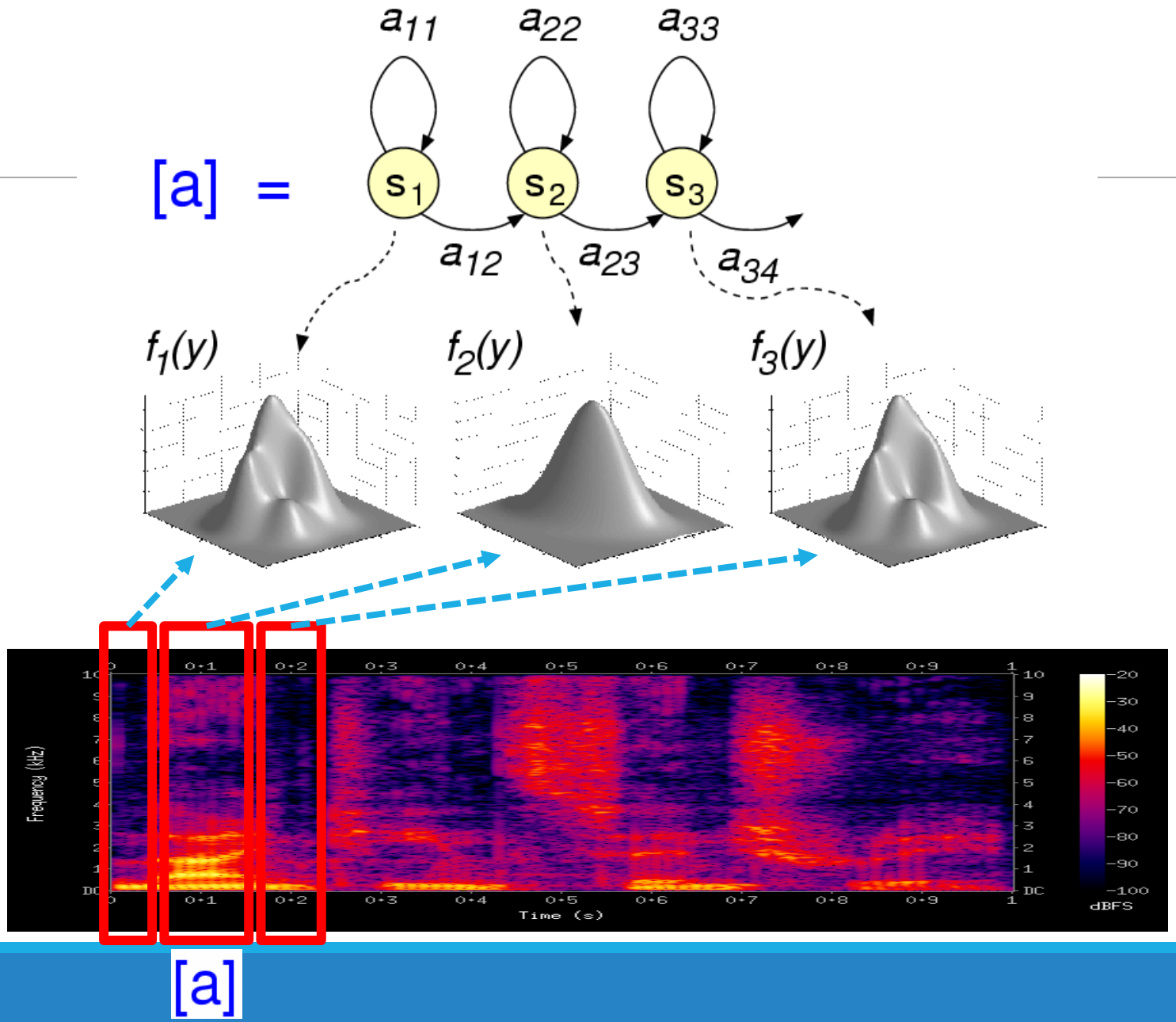

kHertz

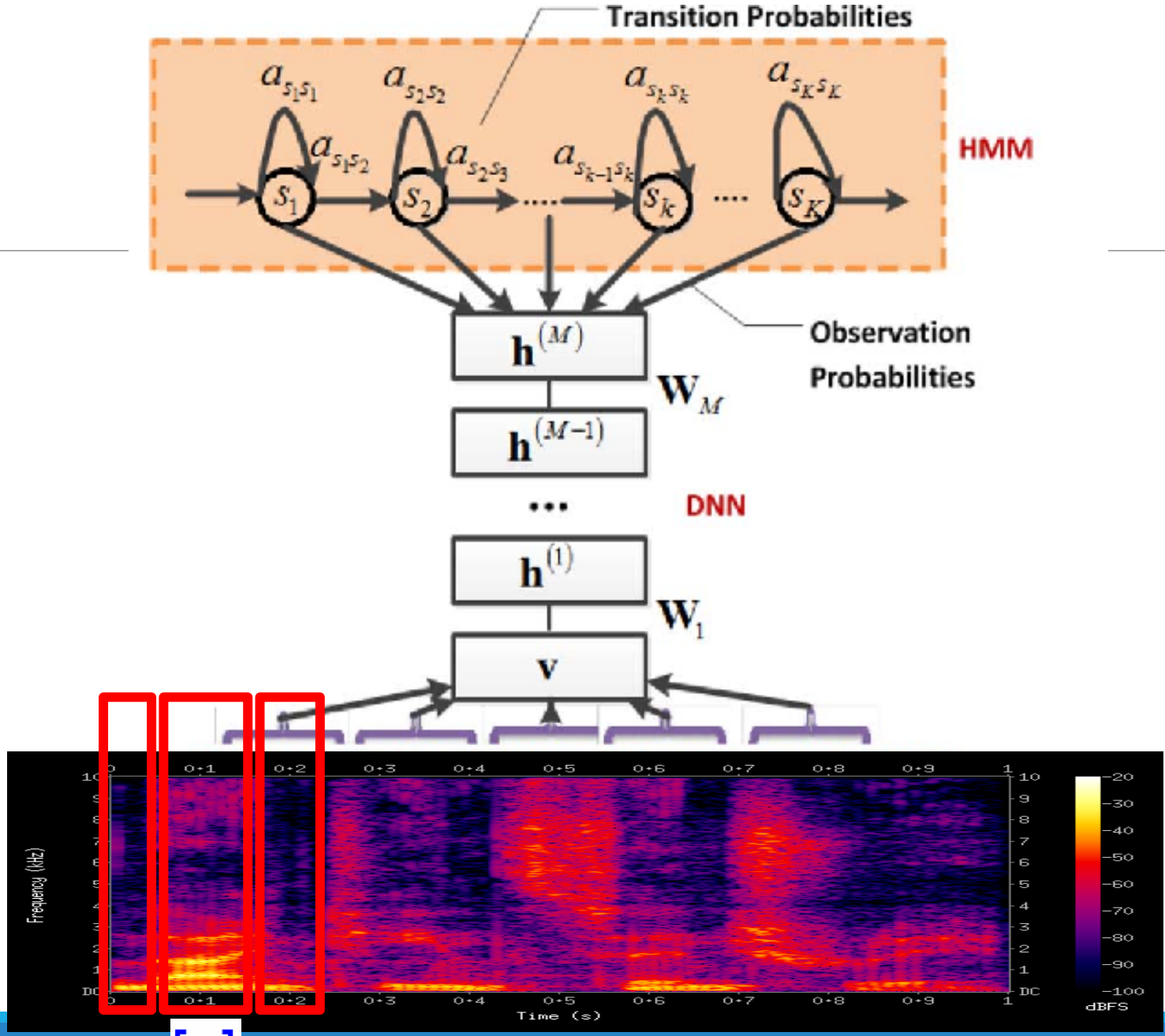

Parameter


kHertz


Parameter

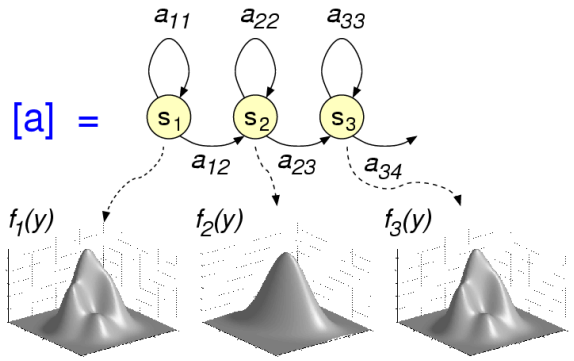
Hidden Markov Models



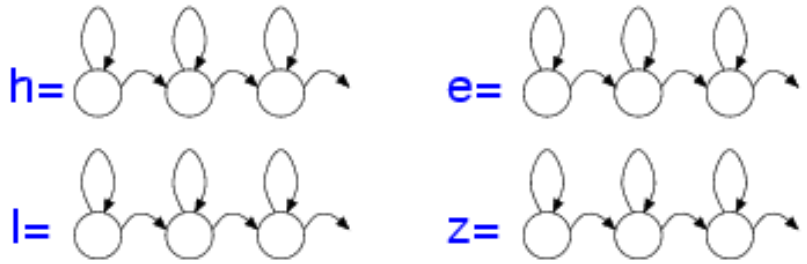

Transition Probabilities
HMM
Observation Probabilities
DNN
Frequency (kHz)
Time (s)
dBFS

[a]

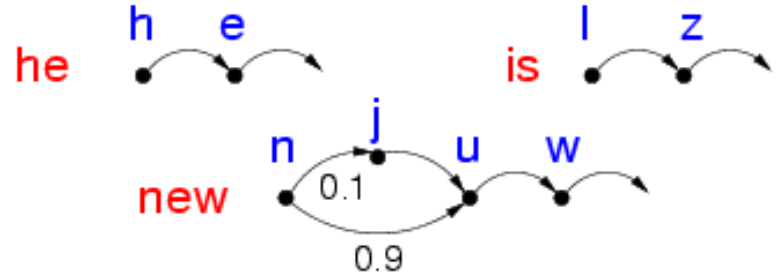
Hidden Markov Models



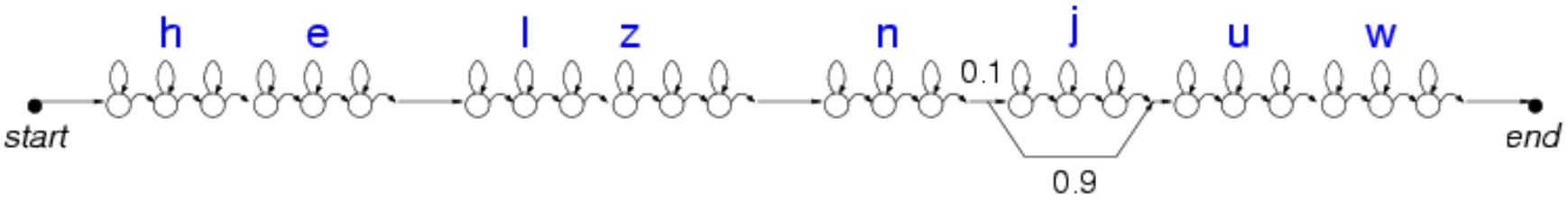
HMM phone models



Lexicon



Sentence model: 'he is new'



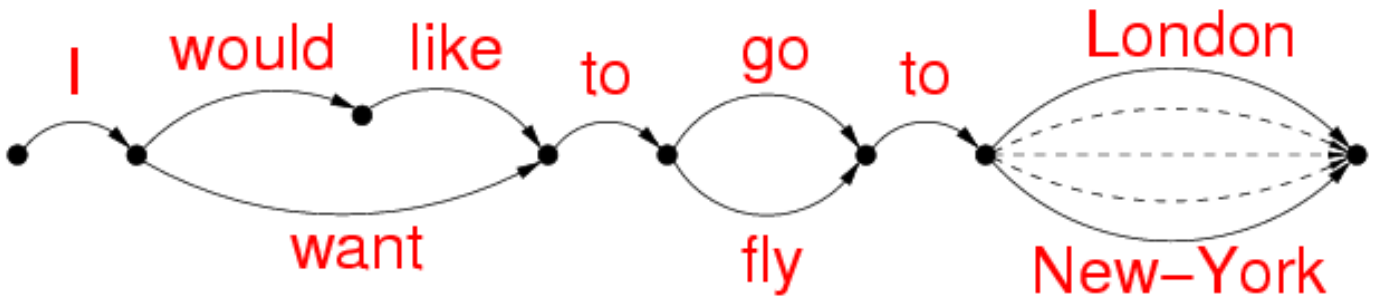
Grammar:

$$\langle \text{sentence}_1 \rangle = \text{I} \left\{ \begin{matrix} \text{would like} \\ \text{want} \end{matrix} \right\} \text{to} \left\{ \begin{matrix} \text{go} \\ \text{fly} \end{matrix} \right\} \text{to} \ \langle \text{airport} \rangle$$

$$\langle \text{sentence}_2 \rangle = \cdots$$

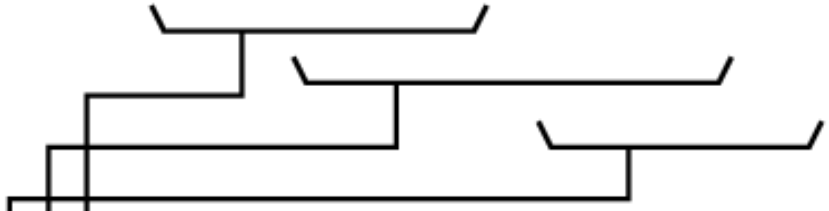
$$\langle \text{airport} \rangle = \{ \text{London}, \ \text{New–York}, \ldots \}$$

Finite–state representation:



N–gram language models

... the united states of ???

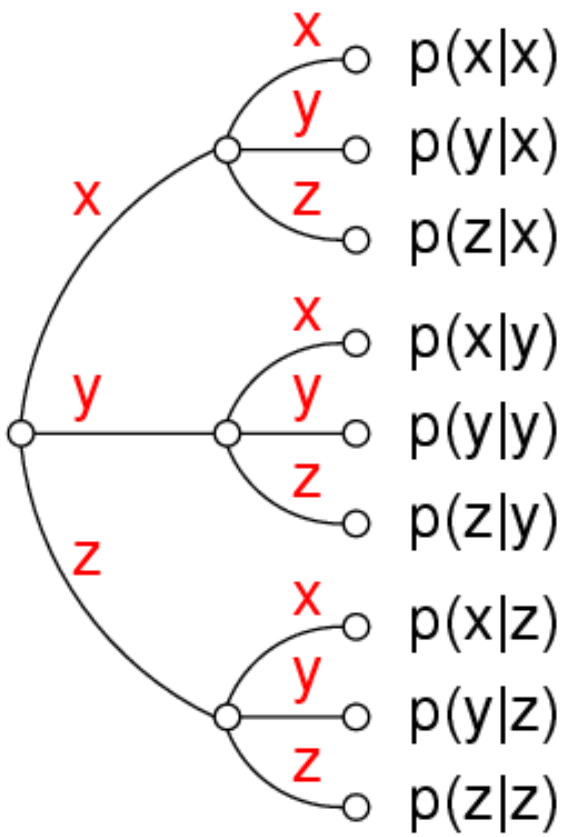


P(states | the united) = . . .

P(of | united states) = . . .

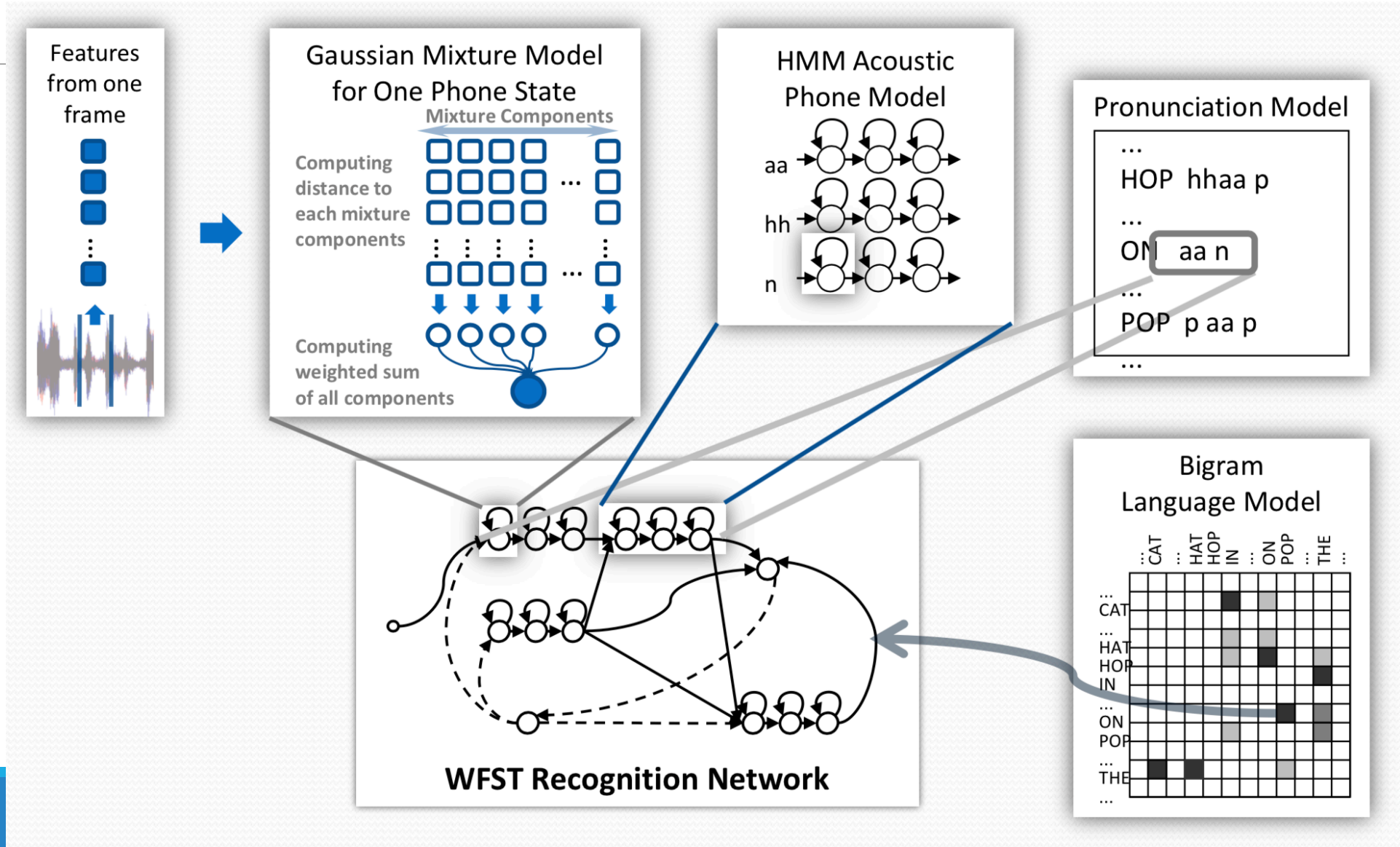
P(America | states of) = . . .

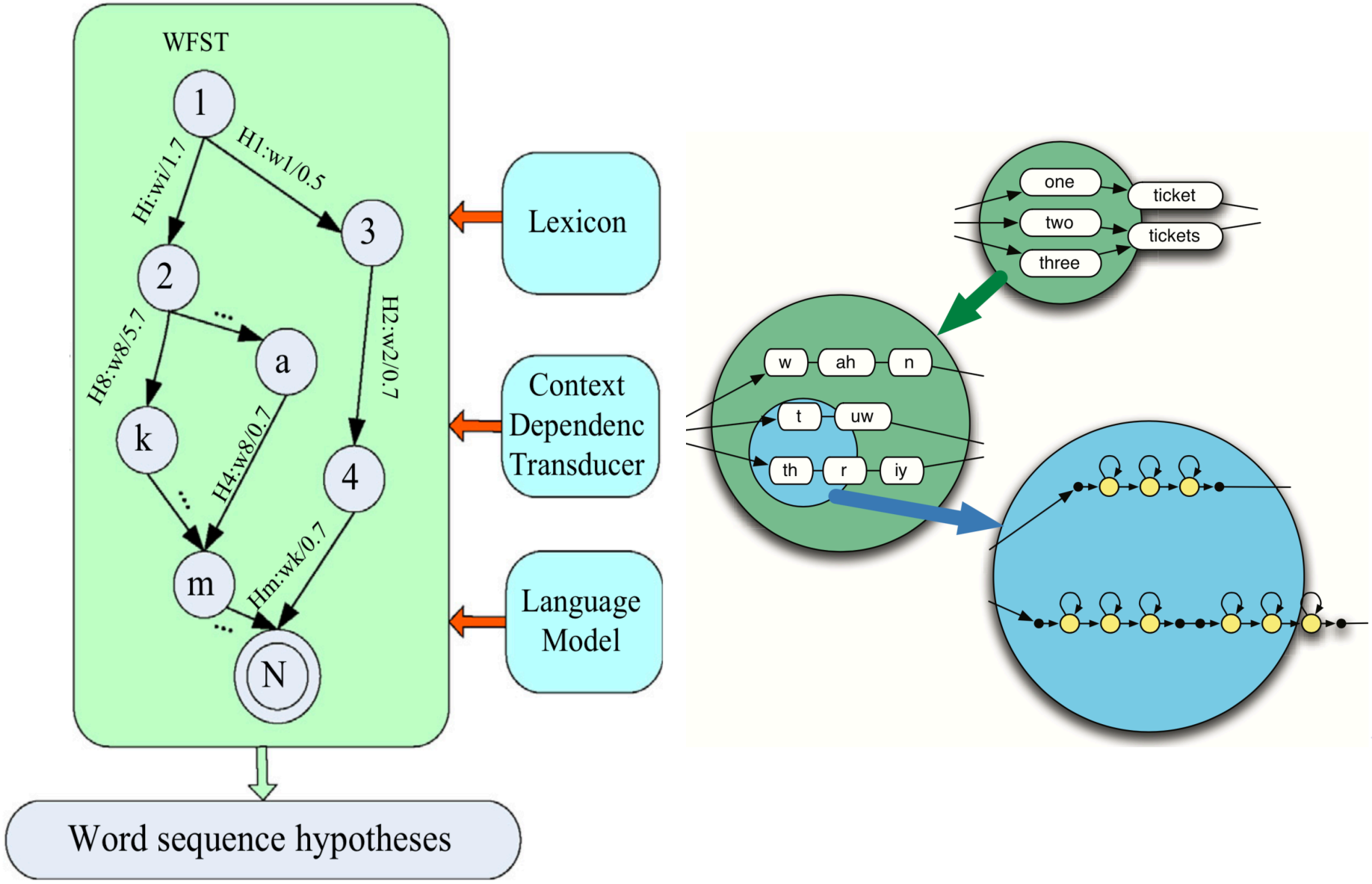
P(Belgium | states of) = . . .



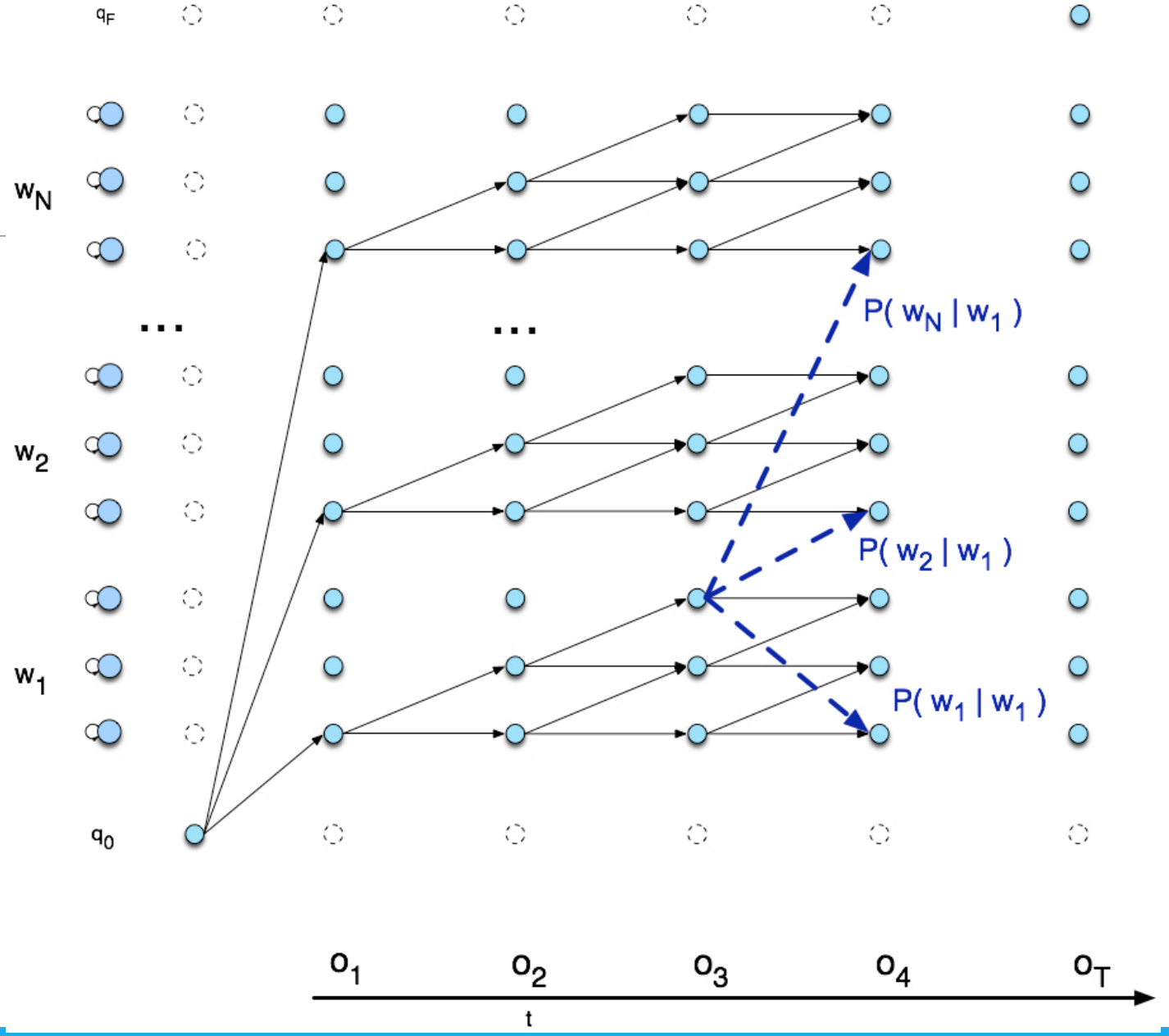
Predict the next word, give a set of predecessor words

Automatic Speech Recognizer

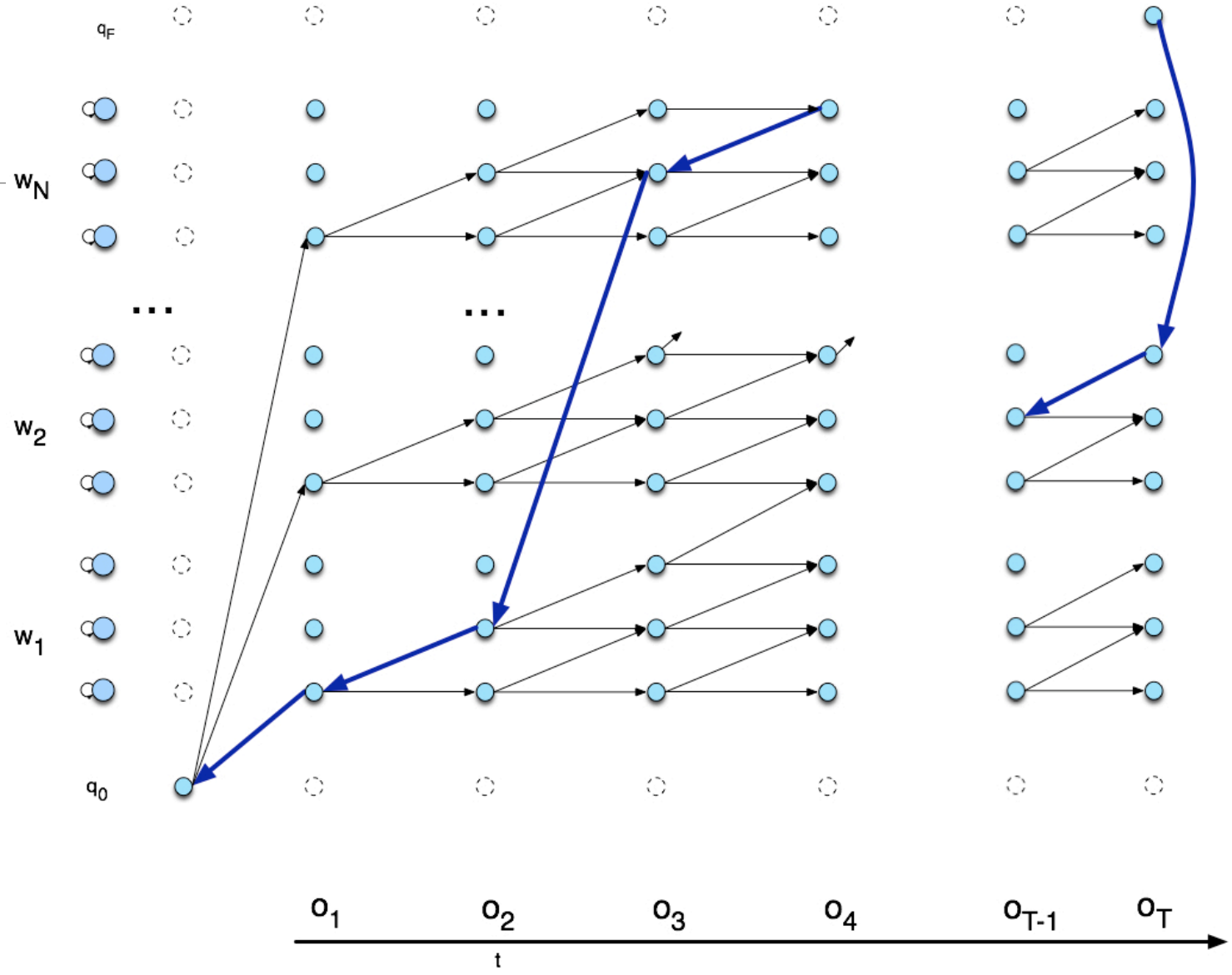



WFST
1
2
3
4
a
k
m
N
Hi:wi/1.7
H1:w1/0.5
H2:w2/0.7
H8:w8/5.7
H4:w8/0.7
Hm:wk/0.7
Lexicon
Context
Dependenc
Transducer
Language
Model
Word sequence hypotheses
one
two
three
ticket
tickets
w
ah
n
t
uw
th
r
iy

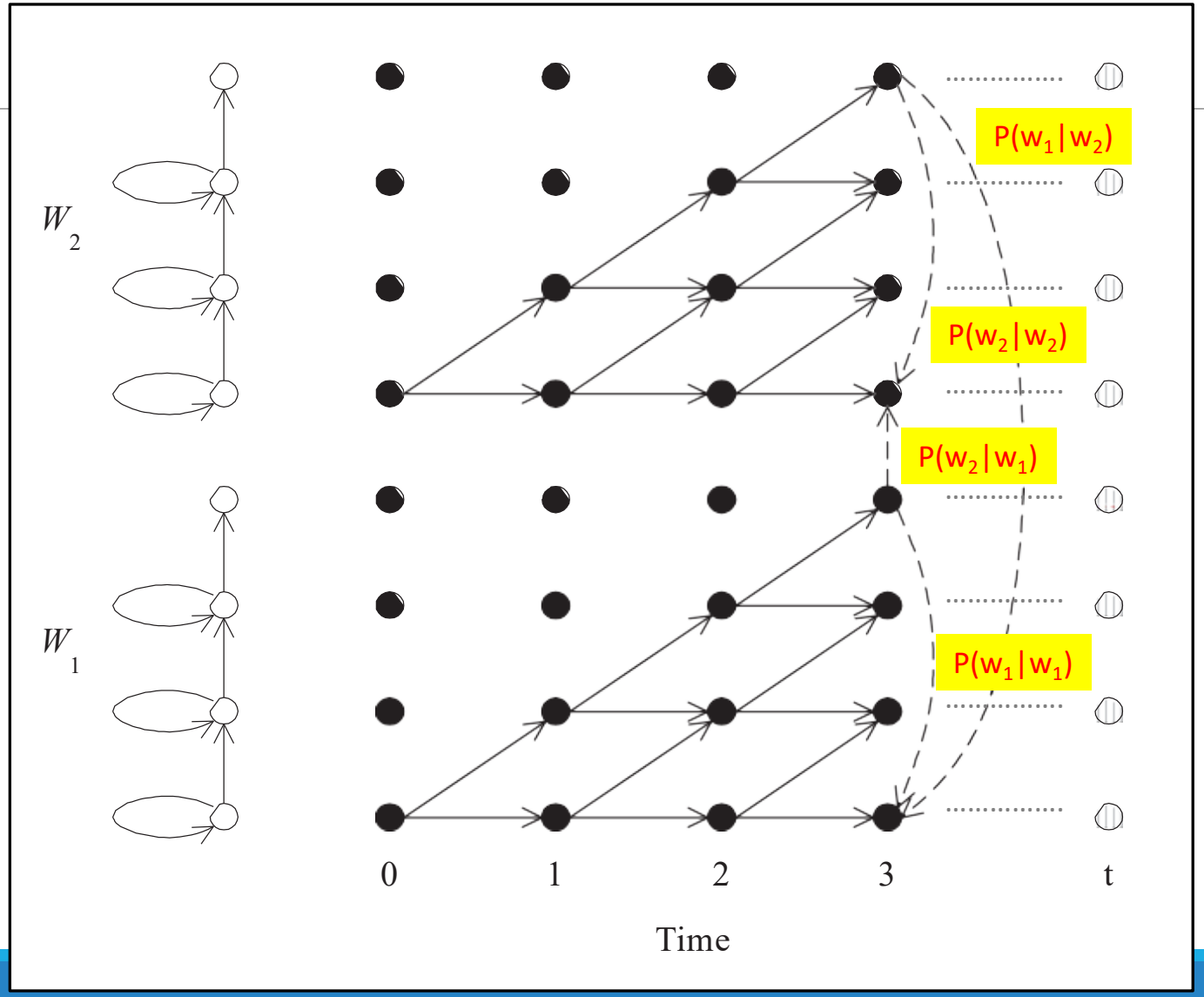
Viterbi Trellis



Viterbi Backtrace

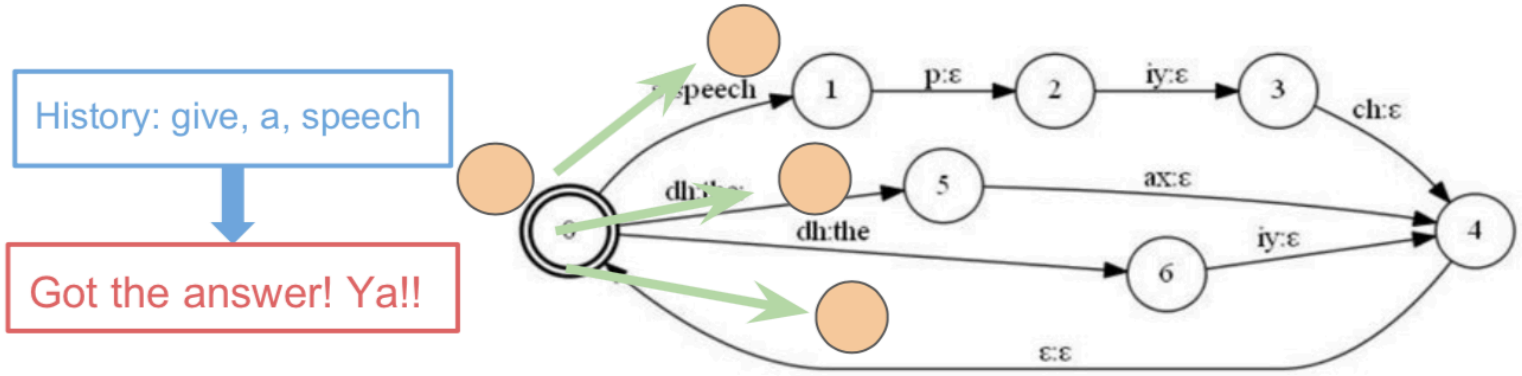


Viterbi Path with Complex Models



Token & Beam Pruning

- The loop of Decode:
  - Token Copy
  - Update AM/LM scores
  - Record the **`highest_score`** over all tokens
  - If **`Token.score < (highest_score - beam)`**, kill it. → beam pruning
- Finally, we choose the token with the highest score.
- Output it history words as answer.



Automatic Speech Recognizer