Automatic Speech Recognizer (ASR):

Bagaimana Membuat Komputer Mendengar?

Dessi Puji Lestari

Sekolah Teknik Elektro dan Informatika
Program Studi Informatika
Institut Teknologi Bandung

1st InaCL Workshop - 7 Januari 2016

Konten

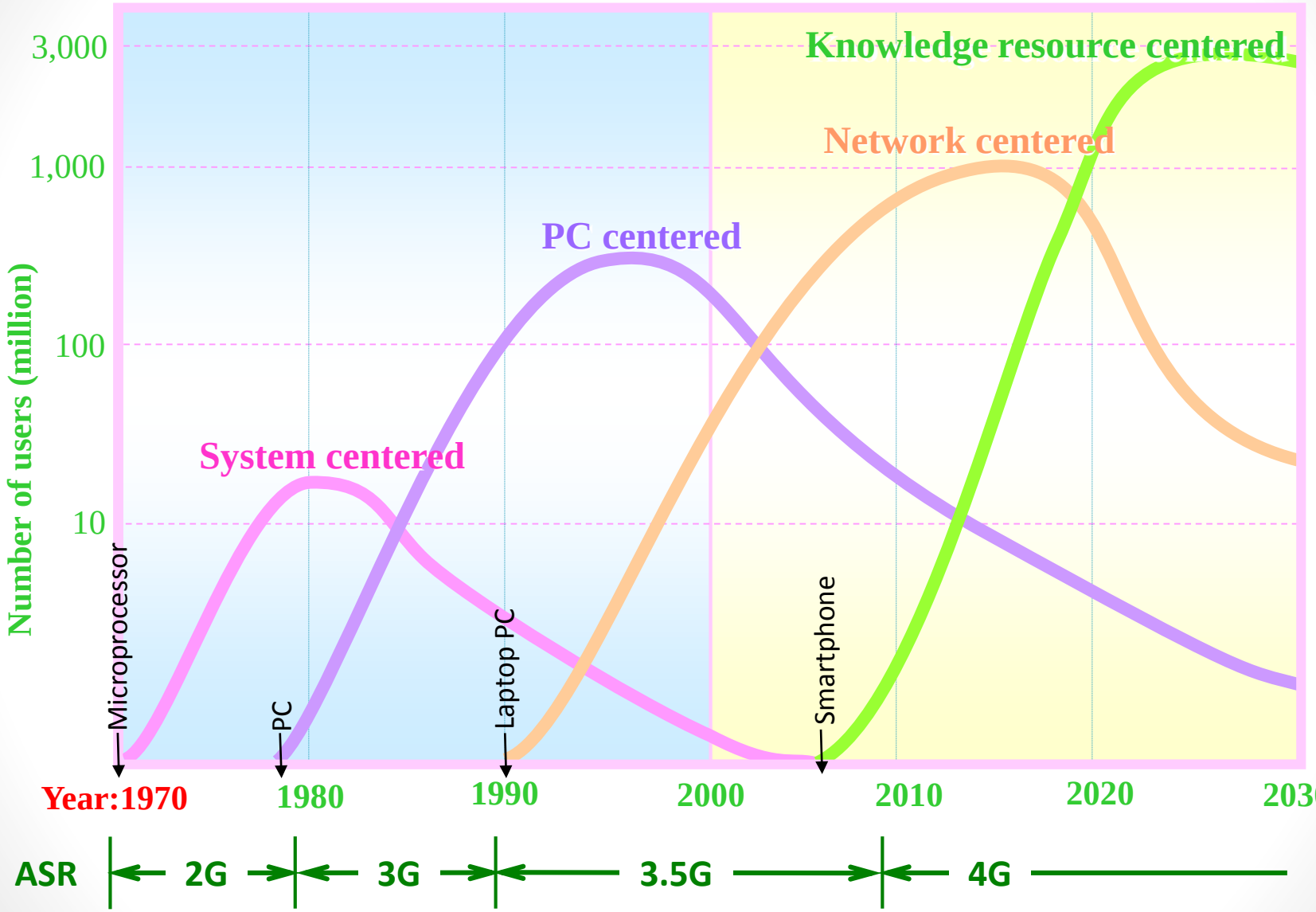
- Lingkup Pemrosesan Ucapan
- Sekilas Karakteristik Sinyal ucapan
- Proses Pengenalan Ucapan pada ASR
  - Penangkapan Bunyi
  - Ektraksi Fitur
  - Pemodelan
  - Pencarian Jawaban
- Studi Kasus

Pemrosesan Suara

- Merupakan aplikasi dari pemrosesan sinyal digital (*Digital Signal Processing*) untuk mengolah dan atau melakukan analisis terhadap sinyal suara.

- Aplikasi utama:
  - Speech Coding
  - Speech Enhancement
  - Speech Recognition (Speech to Text)
  - Speech Synthesis (Text to Speech)
  - Speaker Identification/Verification

Perkembangan Teknologi Informasi



(David C. Moschella: "Waves of Power")

Generasi Teknologi ASR

1950 1960 1970 1980 1990 2000 2010

1952 **1G** 1970

Heuristic approaches
(analog filter bank + logic circuits)

1970 **2G** 1980

Pattern matching
(LPC, FFT, DTW)

Prehistory ASR (1925)

1980 **3G** 1990

Statistical framework
(HMM-GMM, n-gram, neural net)

1990 **3.5G**

Discriminative approaches, robust training, normalization, adaptation, spontaneous speech, rich transcription

? **4G**

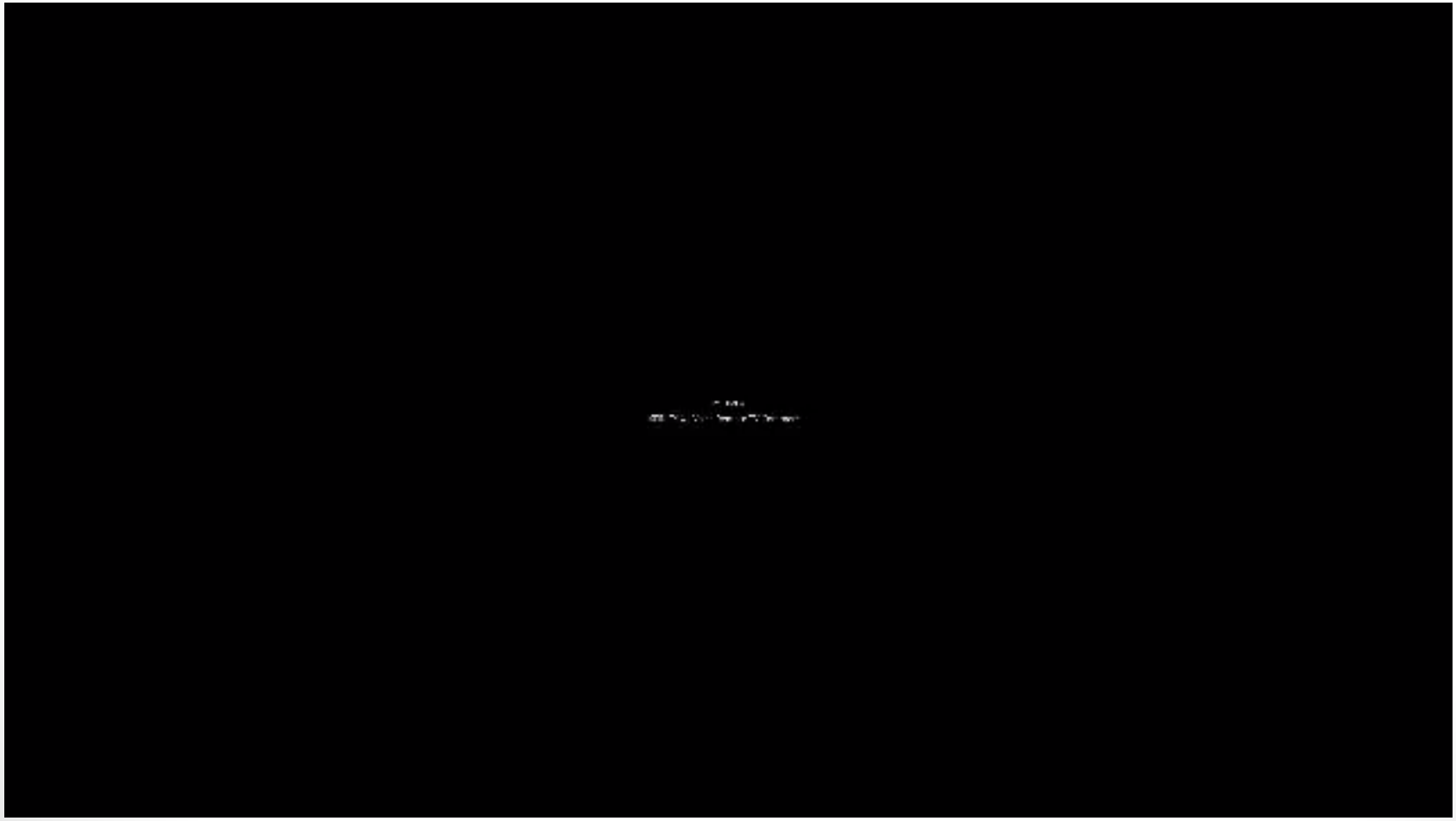
DNN-HMM

Extended knowledge processing

Radio Rex – 1920's ASR

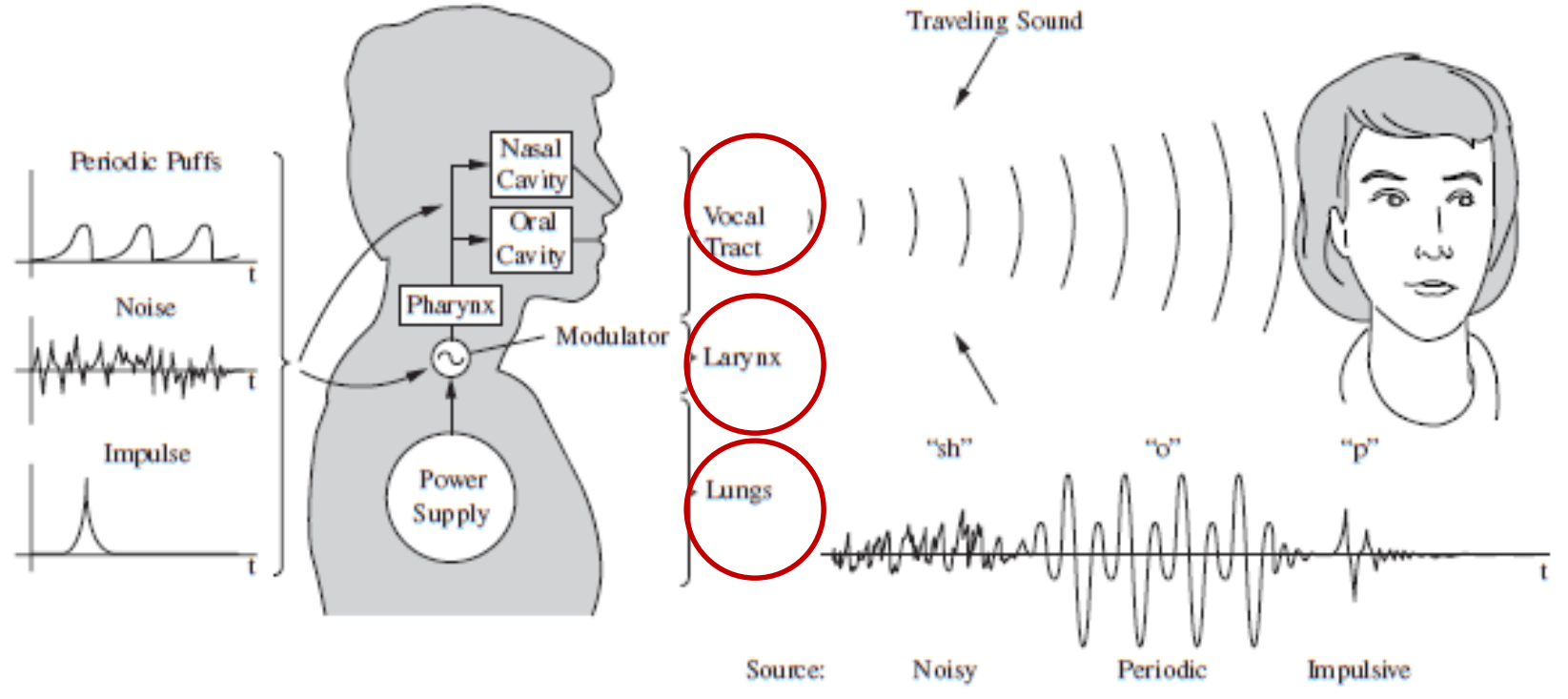
Mainan Boneka Anjing bernama"Rex" (Elmwood Button Co.) yang bisa dipanggil keluar rumahnya dengan menyebut namanya.

Contoh Voice Command : Iklan Remote TV



Karakteristik Sinyal Ucapan

- Sebuah sinyal ucapan terdiri atas rangkaian bunyi atau *phoneme*

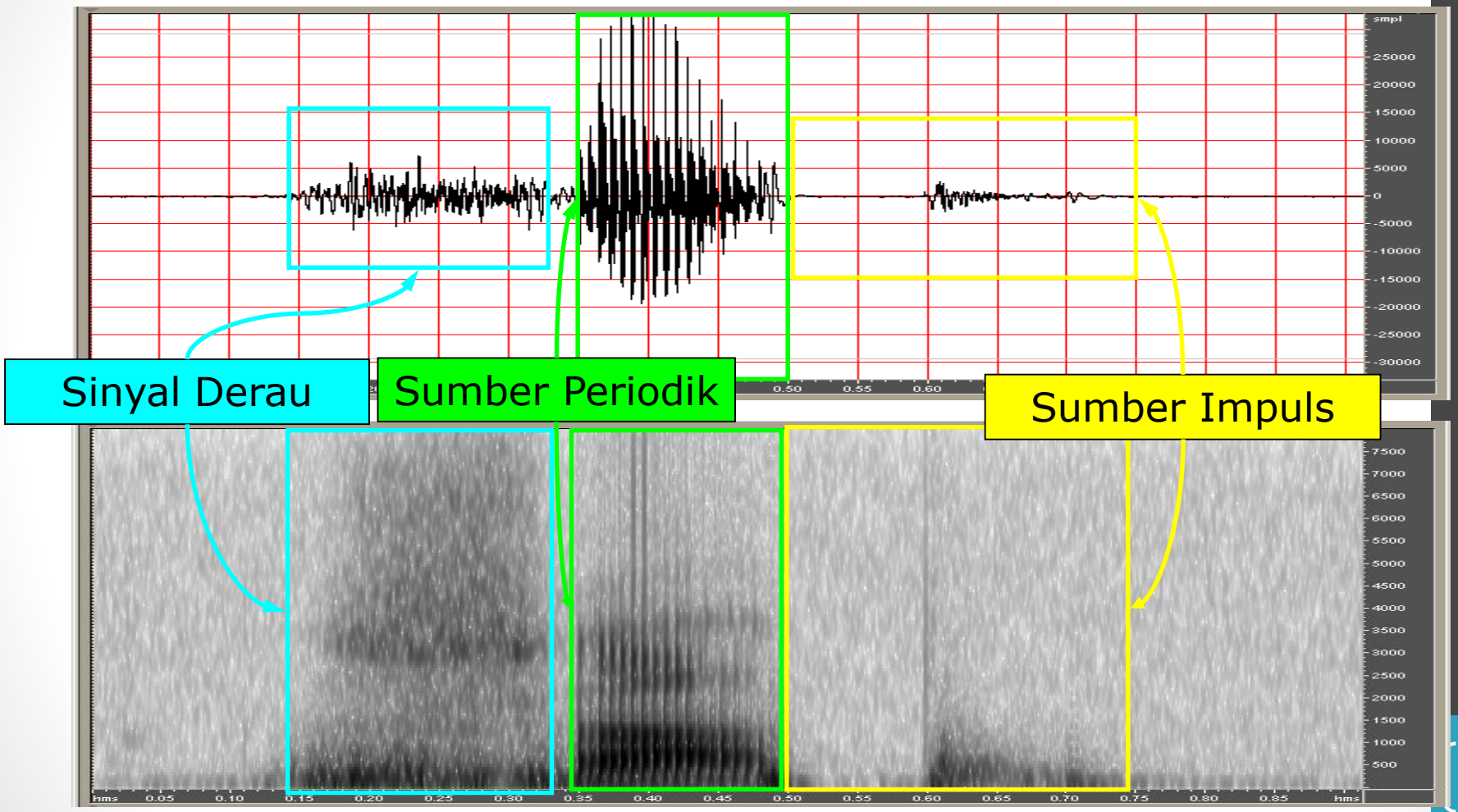


Perbedaan Bunyi

- Sumber bunyi (vocal cord) : pitch (F0)
  - Voiced : Ada getaran di pita suara. Contoh: semua bunyi huruf vokal,
  - Unvoiced : Tidak ada getaran di pita suara. Contoh : Bunyi ribut "s" , "f"

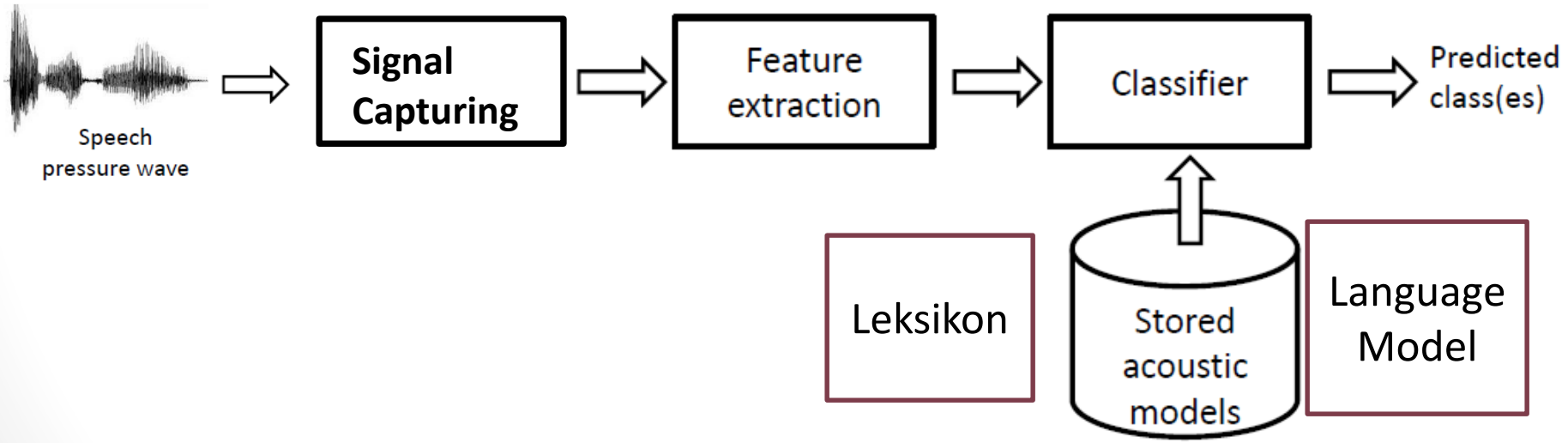
- Perbedaan konfigurasi pada *vocal tract : Frekuensi forman (F1, F2, dst)*
- Gabungan Keduanya
  - Fricative : Membuat friksi atau celah kecil. Contoh : bunyi "f"
  - Plosive : Menutup lalu membuka. Contoh : bunyi "p", "b", "t"
  - Nasal : Sebagian udara harus dilewatkan melalui hidung. Contoh : bunyi "m", "n", "ng", "ny"

Contoh Kata: “Shop”

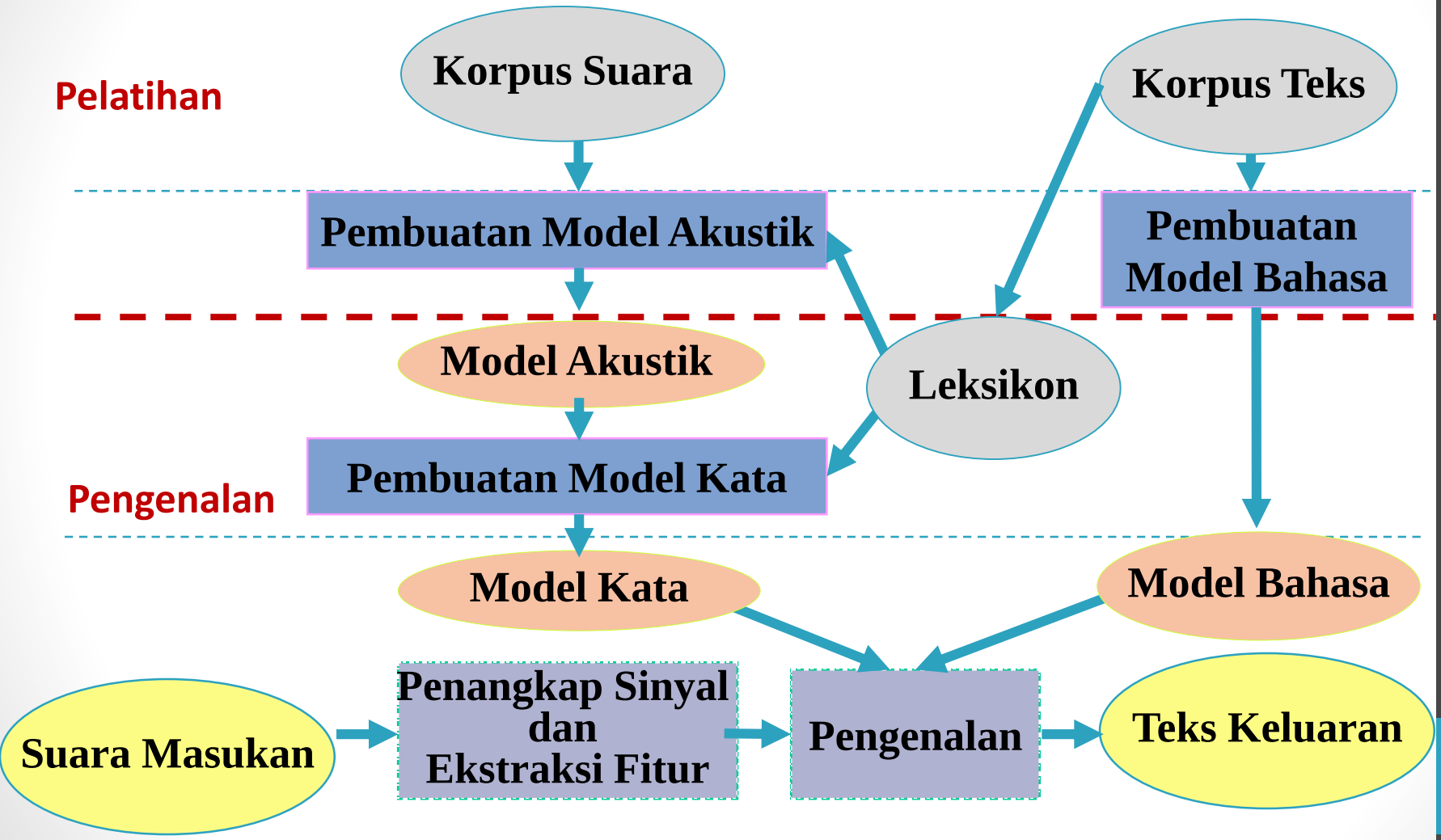


Proses Pengenalan Sinyal Suara

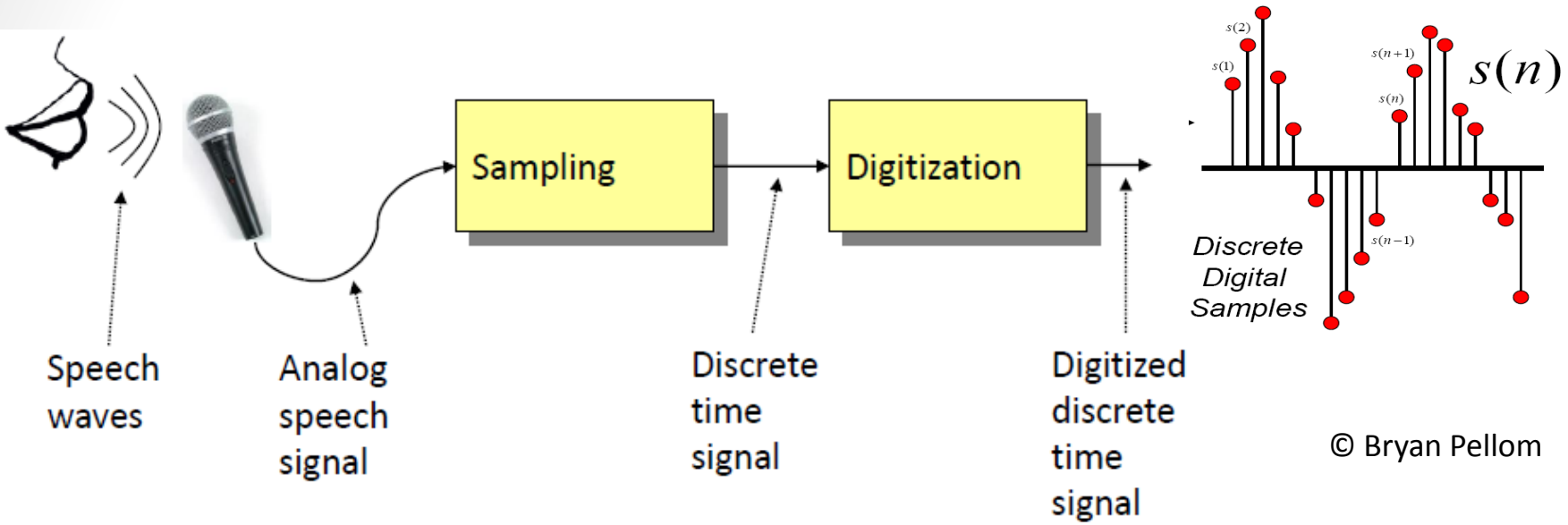
- *Bagaimana memutuskan satu bunyi termasuk ke dalam kelas fonem tertentu, membentuk rangkaian bunyi (kata), lalu menjadi kalimat utuh ?*



Konfigurasi ASR



Penangkapan Sinyal



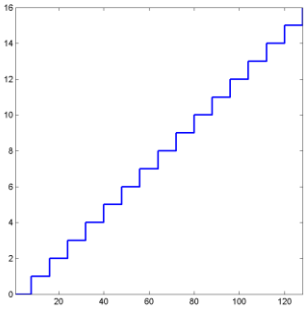
© Bryan Pellom

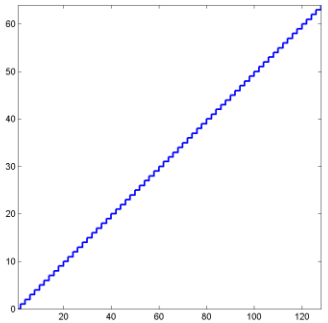
- Penangkapan sinyal dilakukan dengan menggunakan mikrofon
- Cara kerja: meniru sistem pendengaran manusia
  - Terdapat sebuah membran yang bergerak disebabkan gelombang yang menekan
  - Mekanisme transduksi mengkonversikan pergerakan membran menjadi sinyal yang berarti

Frekuensi Sampling dan Bit Kuantisasi

Media	Tingkat Sampling
Telephone (bandwith 300Hz –3.3kHz))	8 KHz
Microphones (bandwith 8KHz)	16 KHz
CD	44.1 KHz per channel

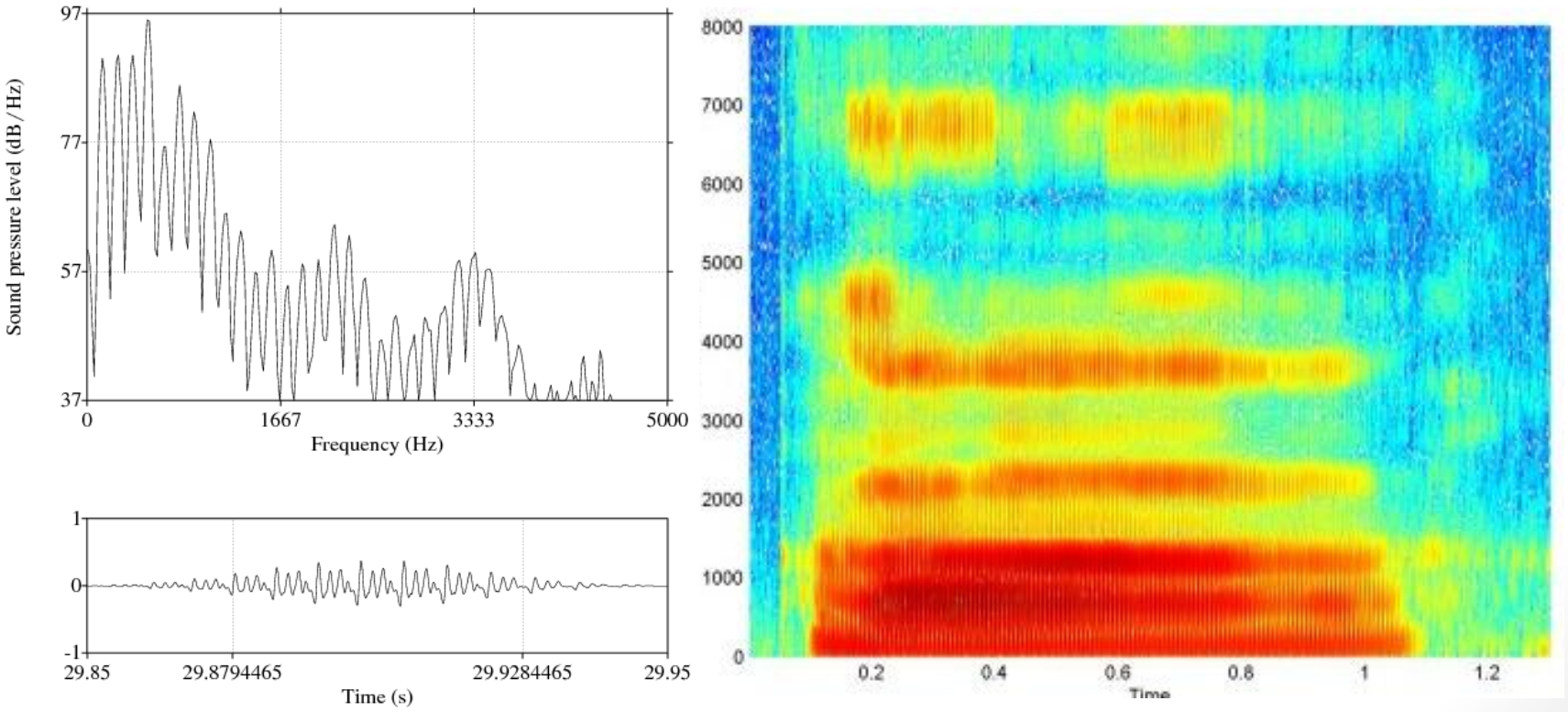
- Kuantisasi Amplitudo : PCM (Pulse Code Modulation) 8 atau 16 bit





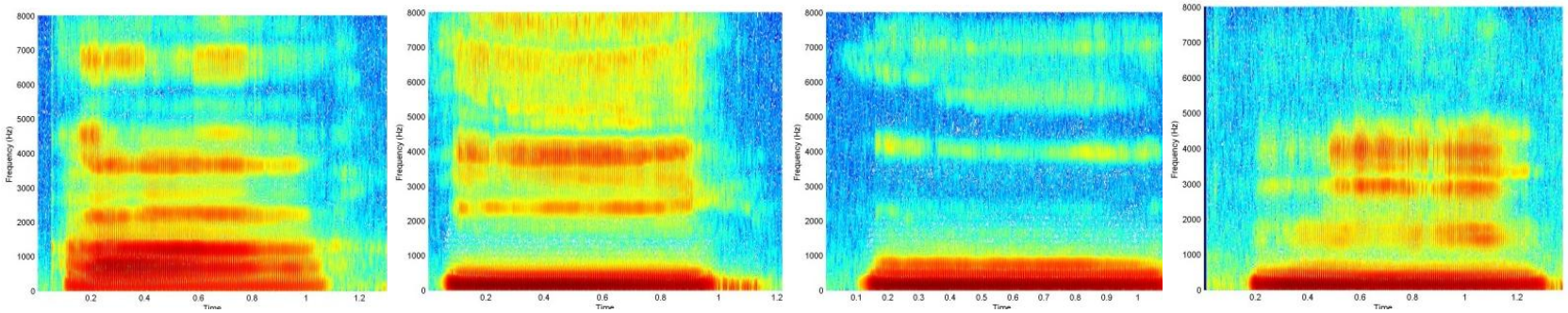
Fitur Penting di dalam Sinyal Unyapan:
Pola Frekuensi dari Spektral

- Analisis dilakukan dengan melihat energi pada berbagai frekuensi di dalam sebuah sinyal sebagai fungsi dari waktu (spektrogram)



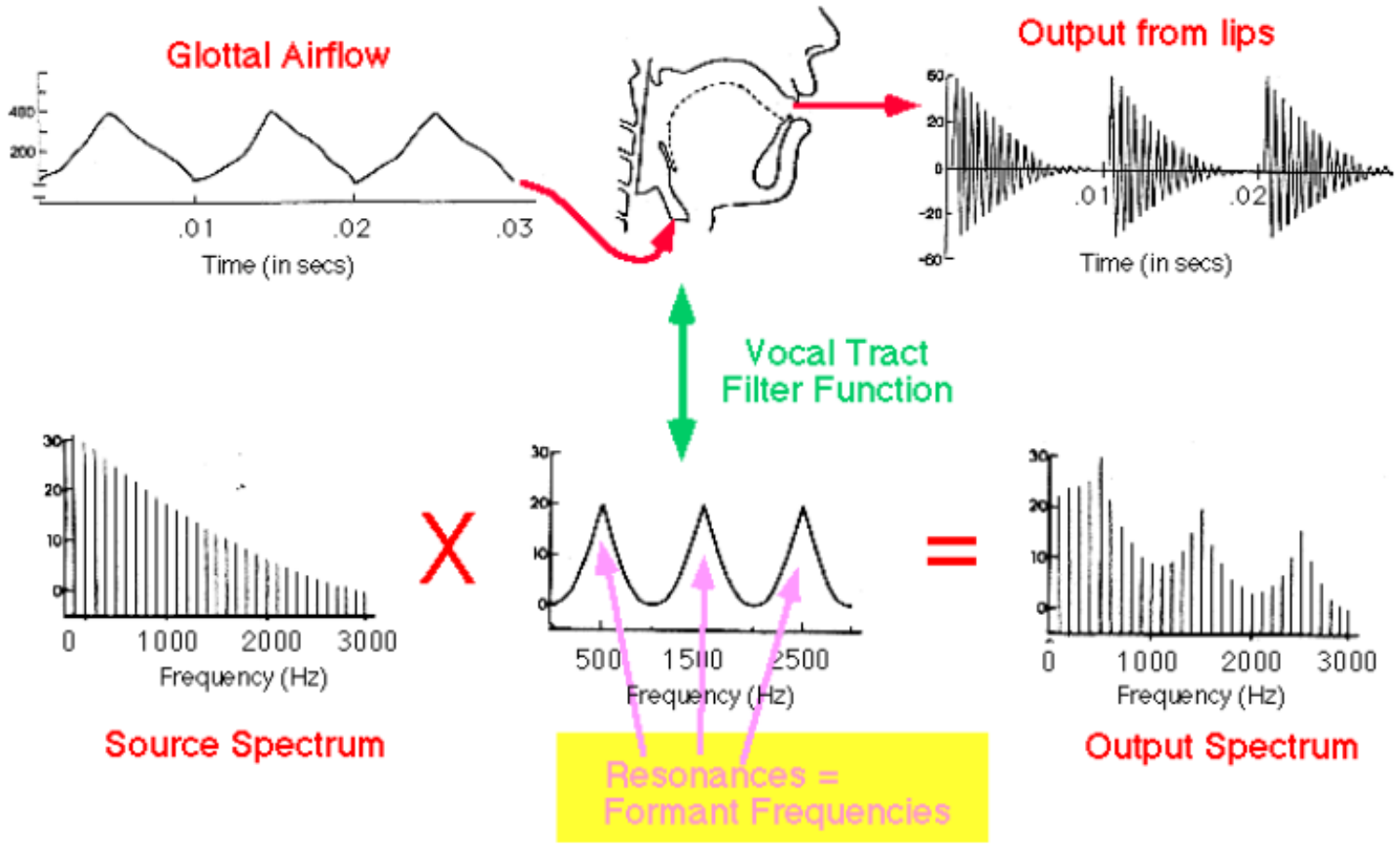
Pola Frekuensi

- Instans yang berbeda untuk sebuah bunyi yang sama memiliki pola yang mirip
- Fitur yang dihasilkan harus menangkap struktur spektral seperti ini



AA IY UW M

Sumber dan Filter Bunyi



Mel Frequency Cepstrum Coefficients (MFCC)

- MFCCs dapat secara akurat merepresentasikan *envelope of the short time power spectrum*.

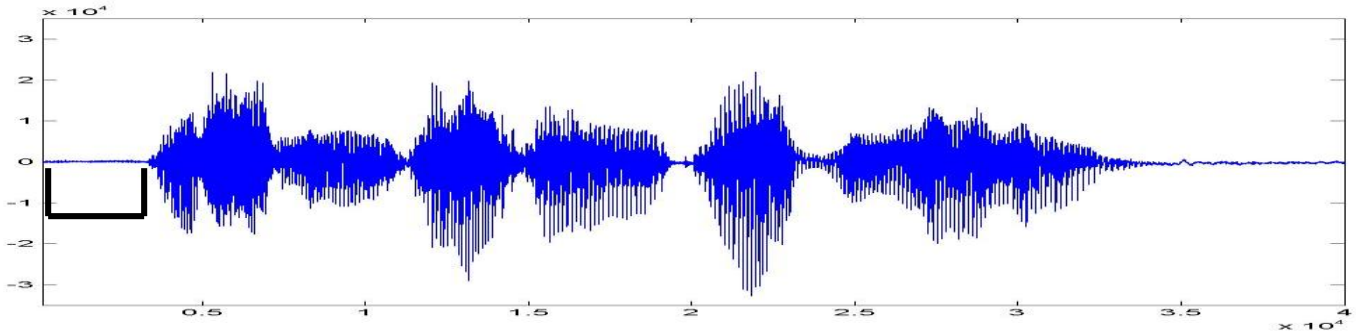
  (Davis and Mermelstein, 1980)

  - Menjadi *state-of-the-art* hingga saat ini.

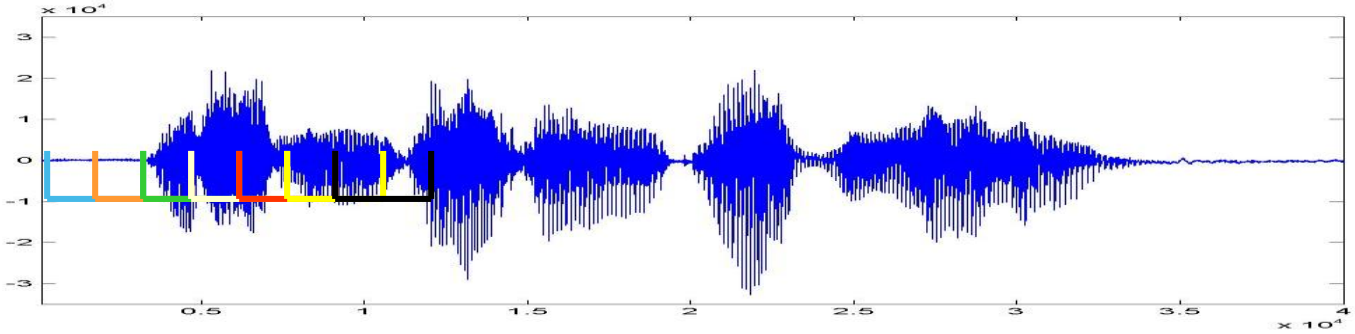
- Sebelumnya digunakan:
  - Linear Prediction Coefficients (LPCs)
  - Linear Prediction Cepstral Coefficients (LPCCs)

Signal Framing

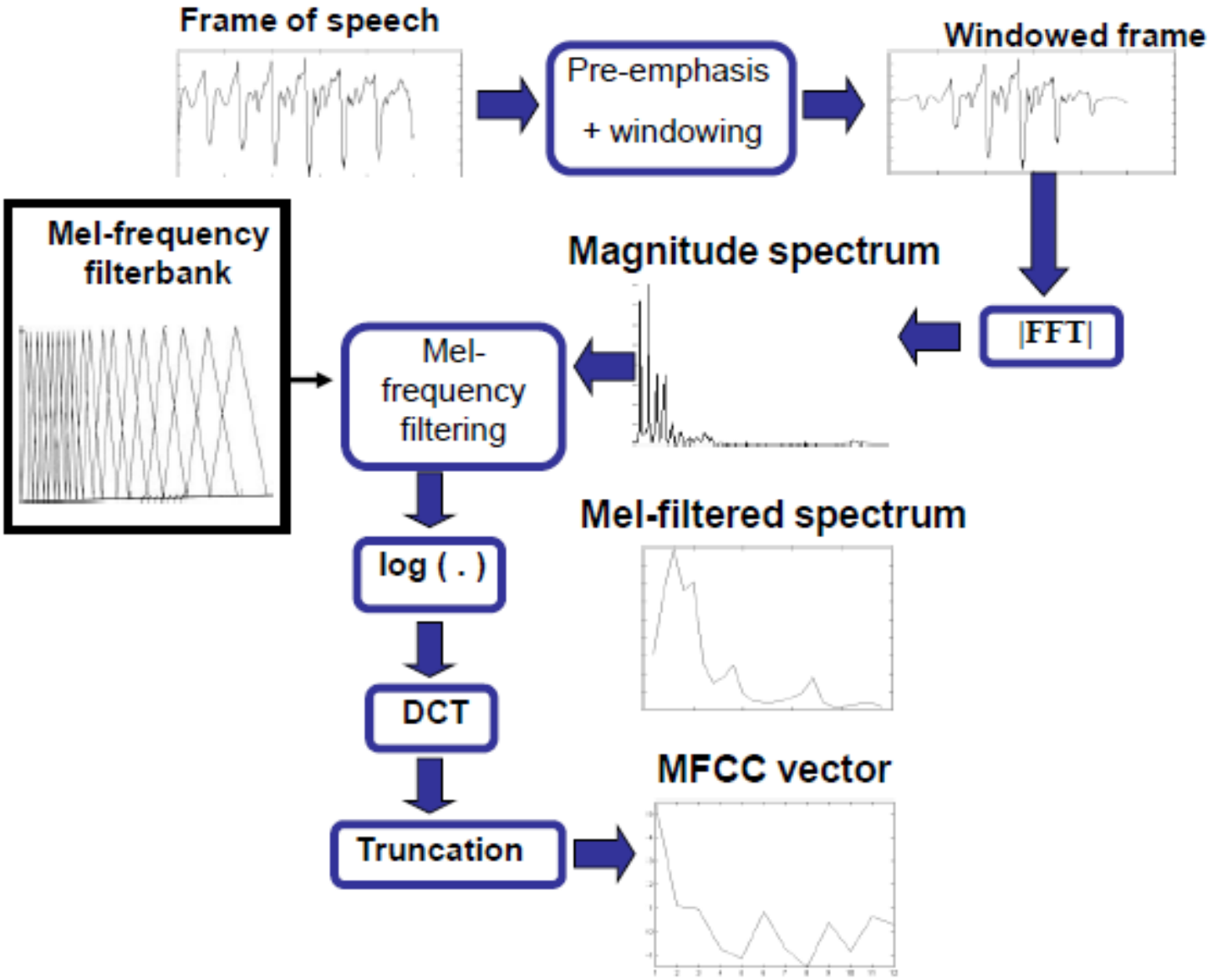
- Sinyal diproses per segmen atau frame
  - Ukuran segmen 20-25 ms.



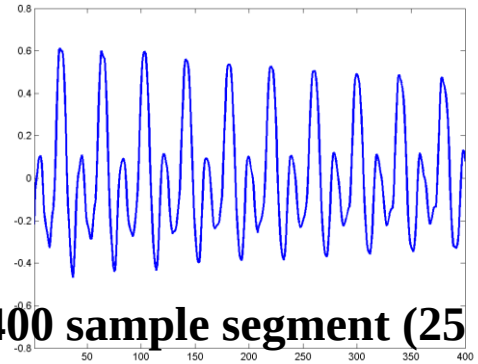
- Segmen beririsan sebanyak 10-15 ms.



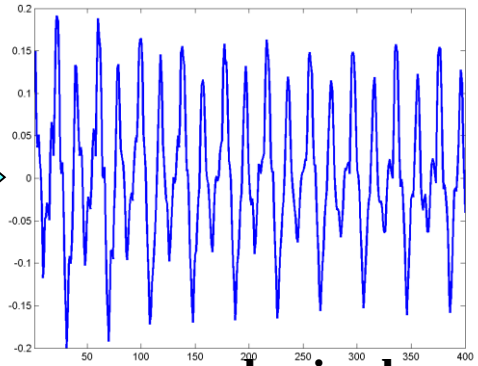
MFCC Diagram dengan FFT



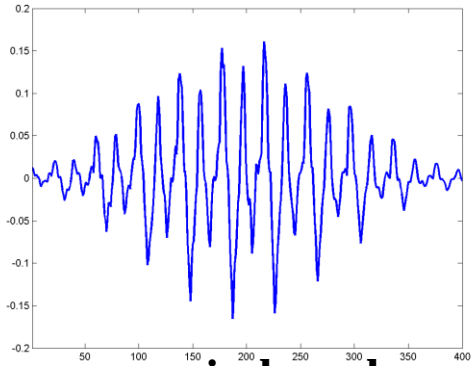
Contoh Visualisasi Ekstraksi Fitur MFCC



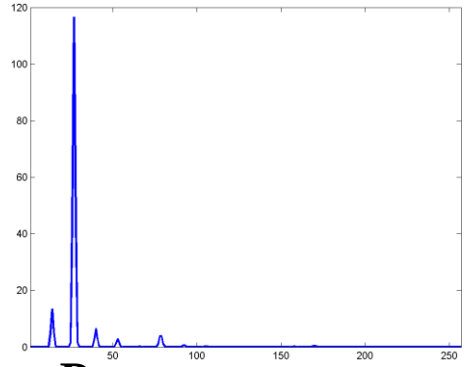
400 sample segment (25 ms) from 16khz signal



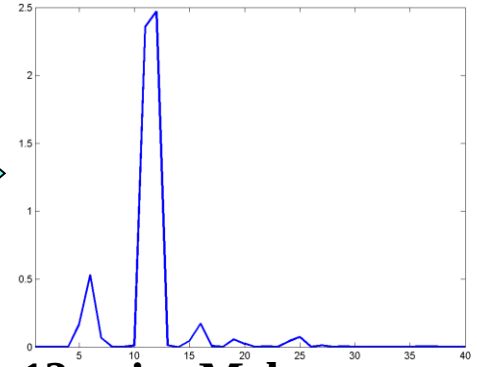
preemphasized



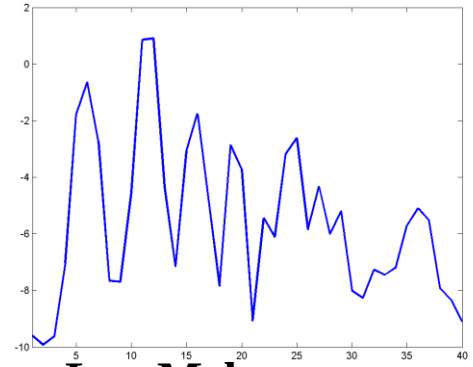
windowed



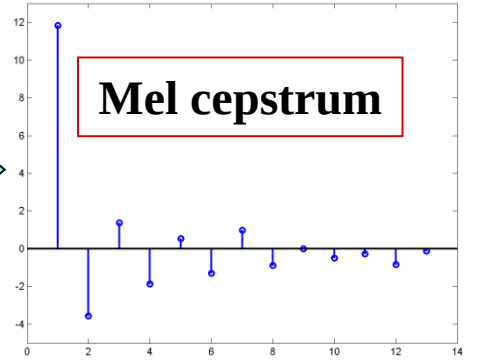
Power spectrum



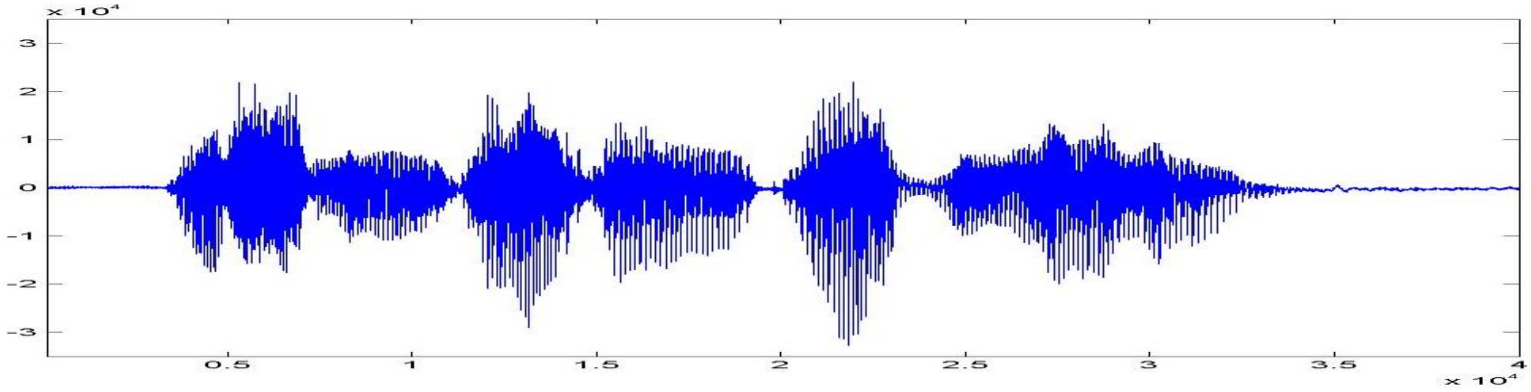
12 point Mel spectrum

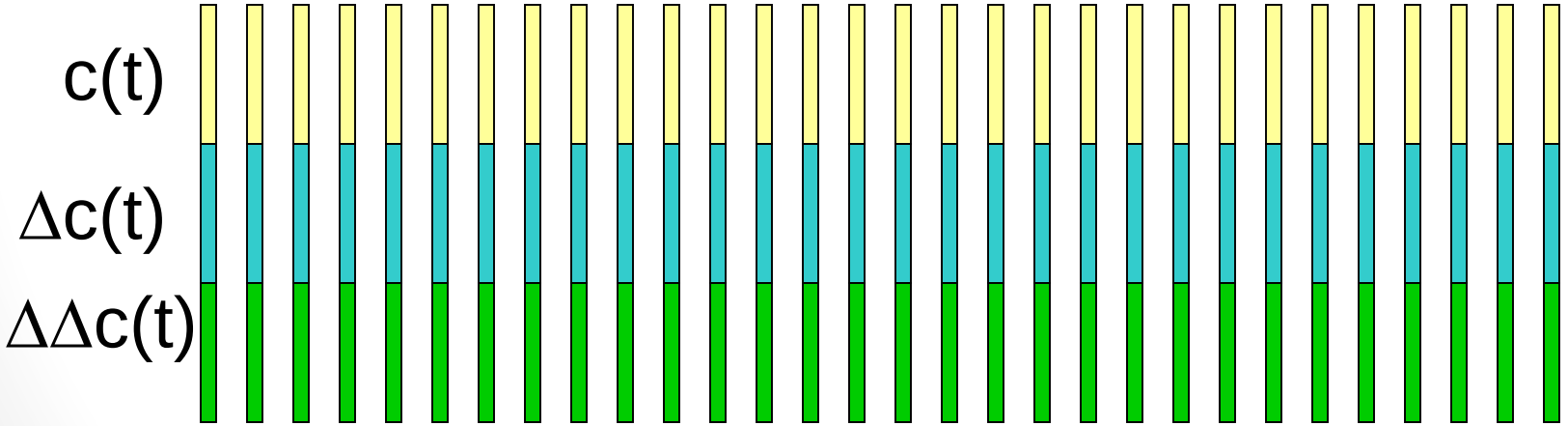


Log Mel spectrum



Fitur MFCC



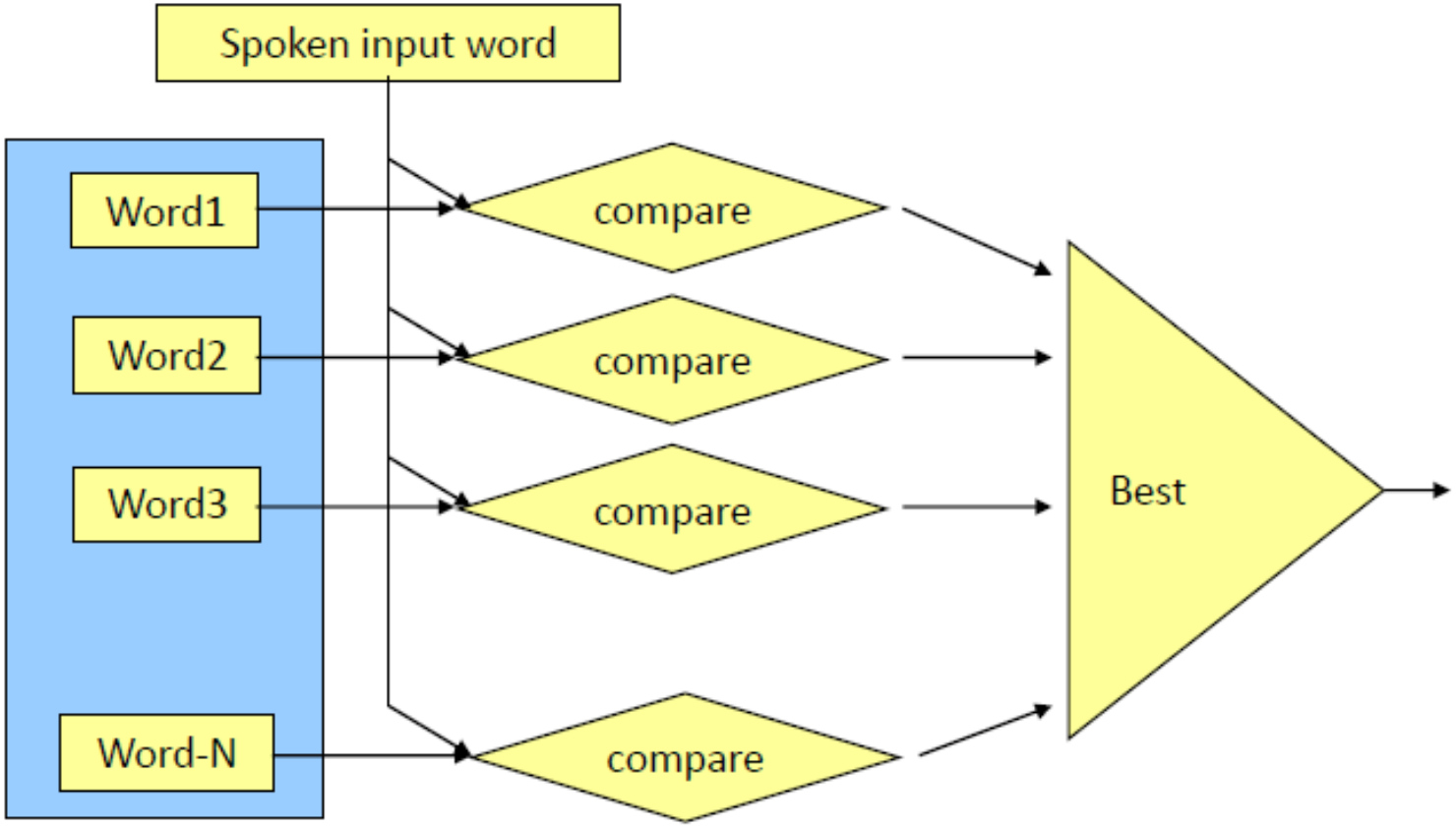


Fitur MFCC

- Biasanya MFCCs memiliki 39 Fitur

39	Fitur MFCC
12	Koefisien Cesptral
12	Delta Koefisien Cesptral
12	Delta Delta Koefisien Cesptral
1	Koefisien Energi
1	Delta Koefisien Energi
1	Delta Delta Koefisien Energi

Classifier Kata/Bunyi/Kalimat



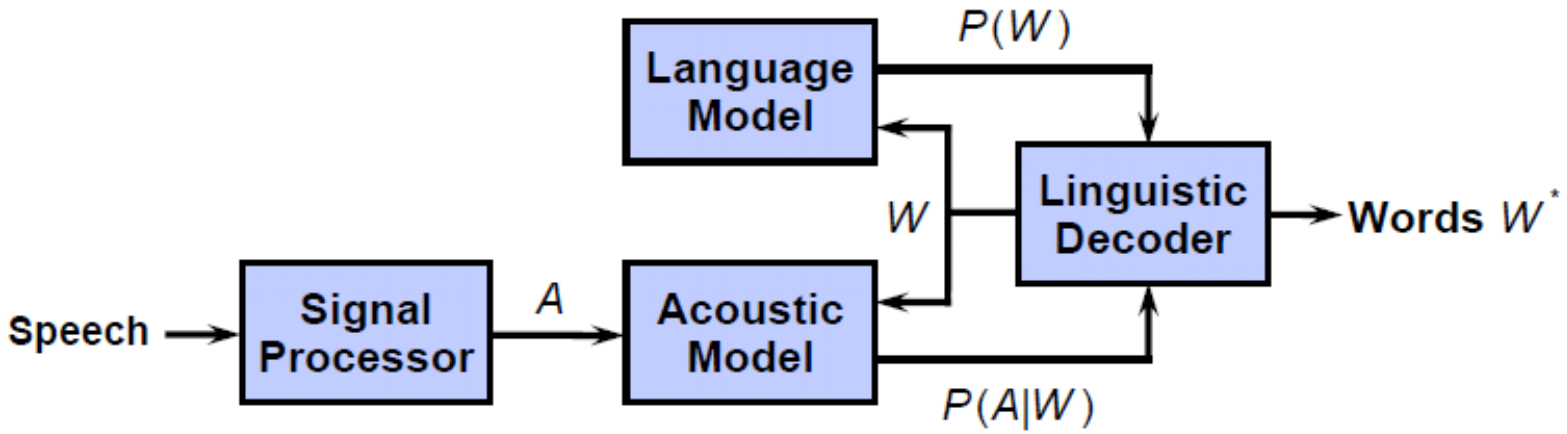
Classifier Berbasis Model

- **Given acoustic observations, *A*, choose word sequence, *W**, which maximizes *a posteriori* probability, *P(W|A)***

$$W^* = \arg\max_W P(W \mid A)$$

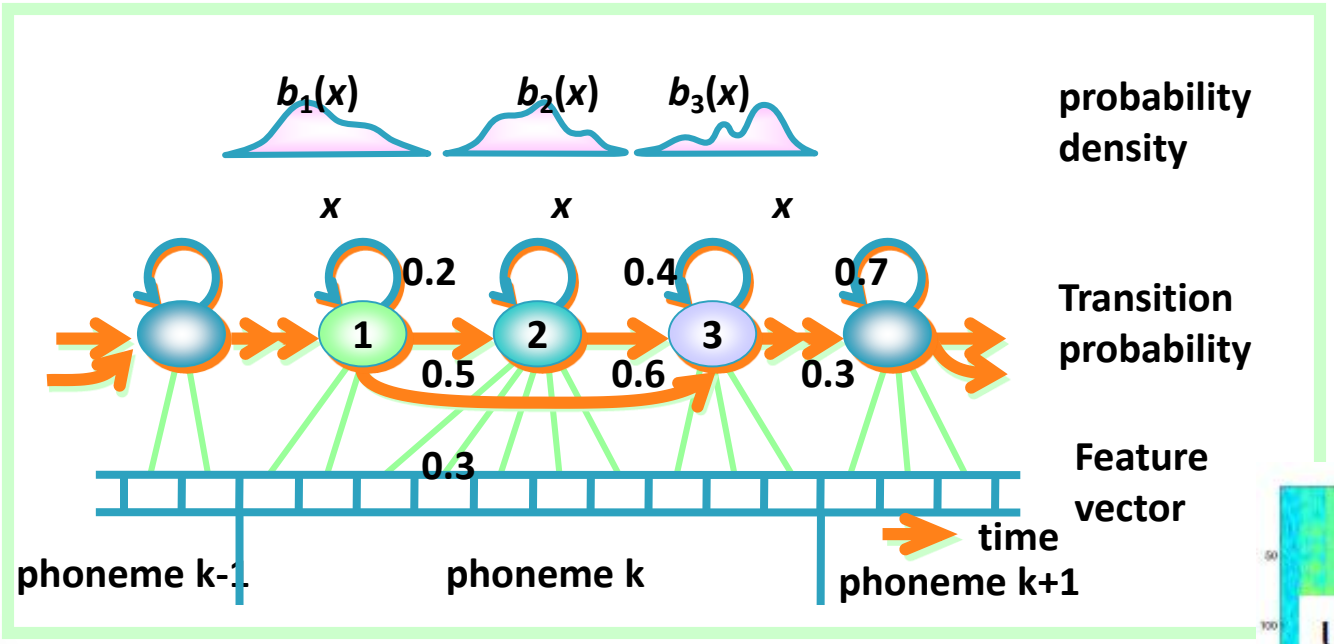
- **Bayes rule is typically used to decompose *P(W|A)* into acoustic and linguistic terms**

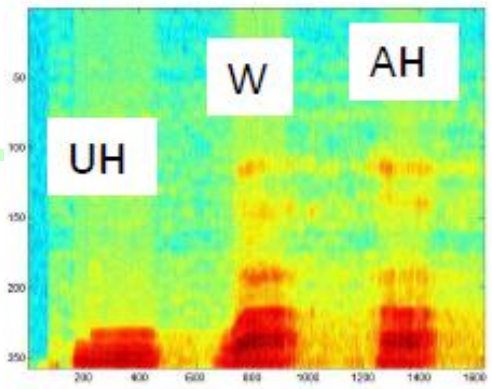
$$P(W \mid A) = \frac{P(A \mid W)P(W)}{P(A)}$$



Model Akustik: HMM

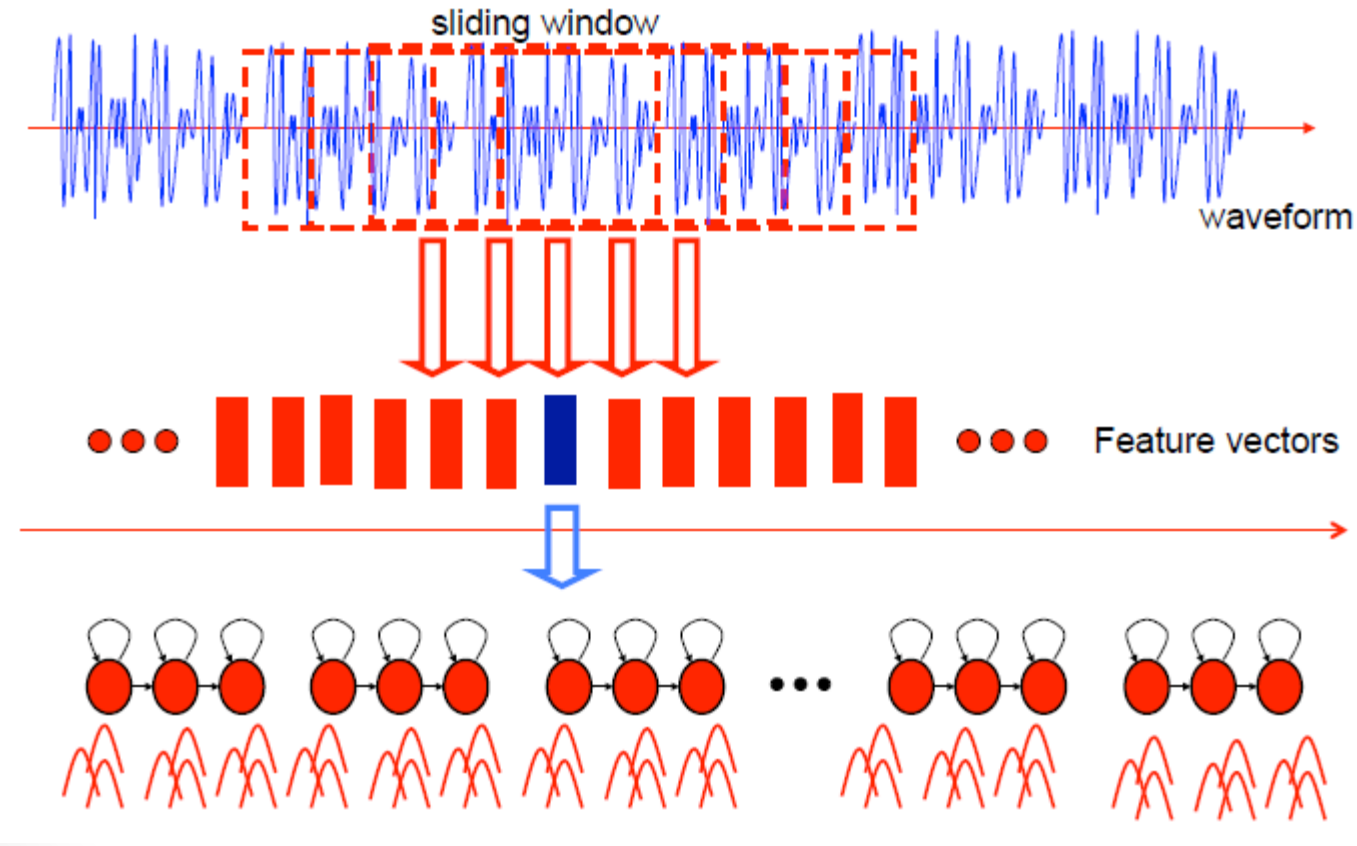
- $P(A|W)$ merupakan probabilitas memproduksi sebuah observasi akustik A jika diberikan rangkaian kata W.
- Probabilitas ini biasanya direpresentasikan dengan Hidden Markov Models (HMMs).





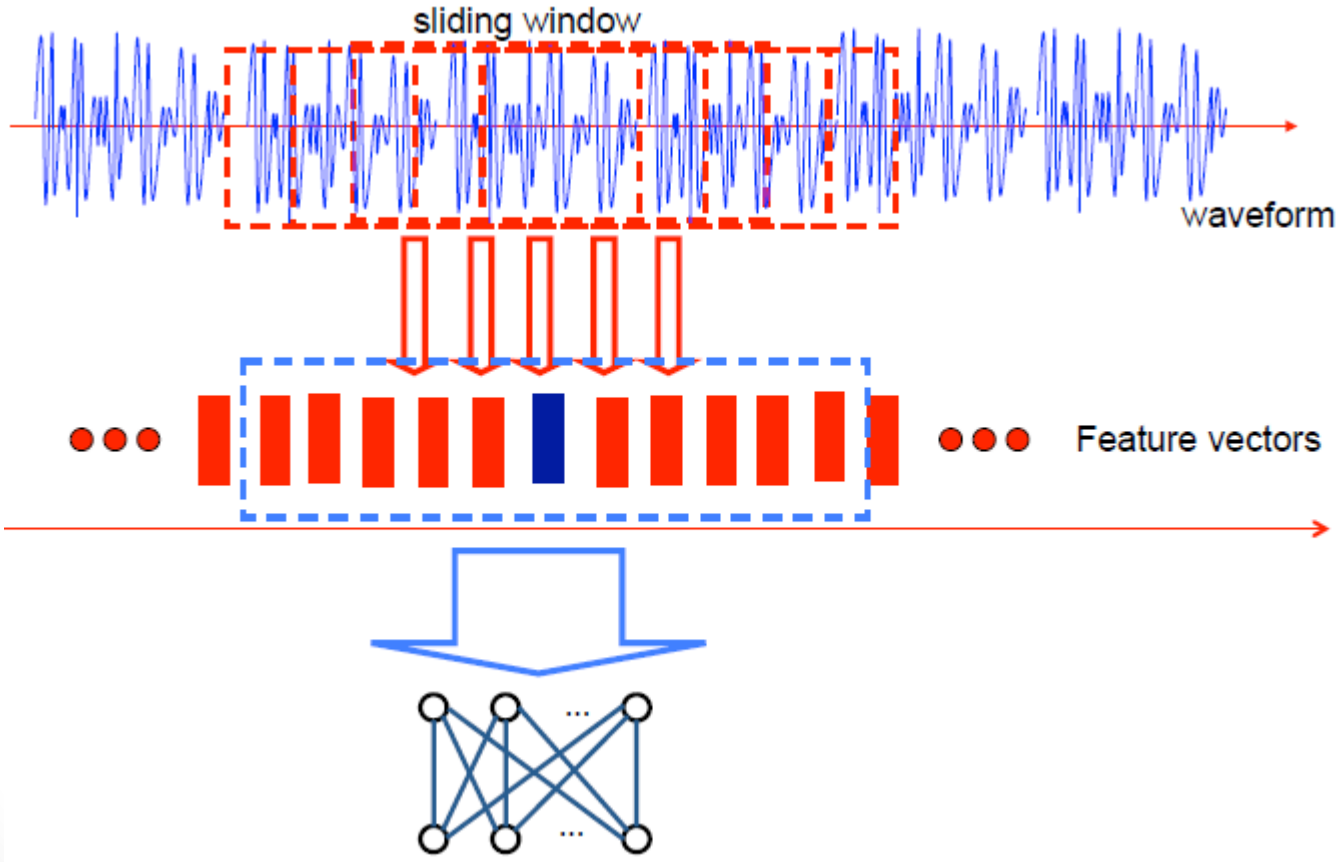
Representasi Probabilitas Output (1)

- Gaussian Mixture Model (n-mixture komponen)

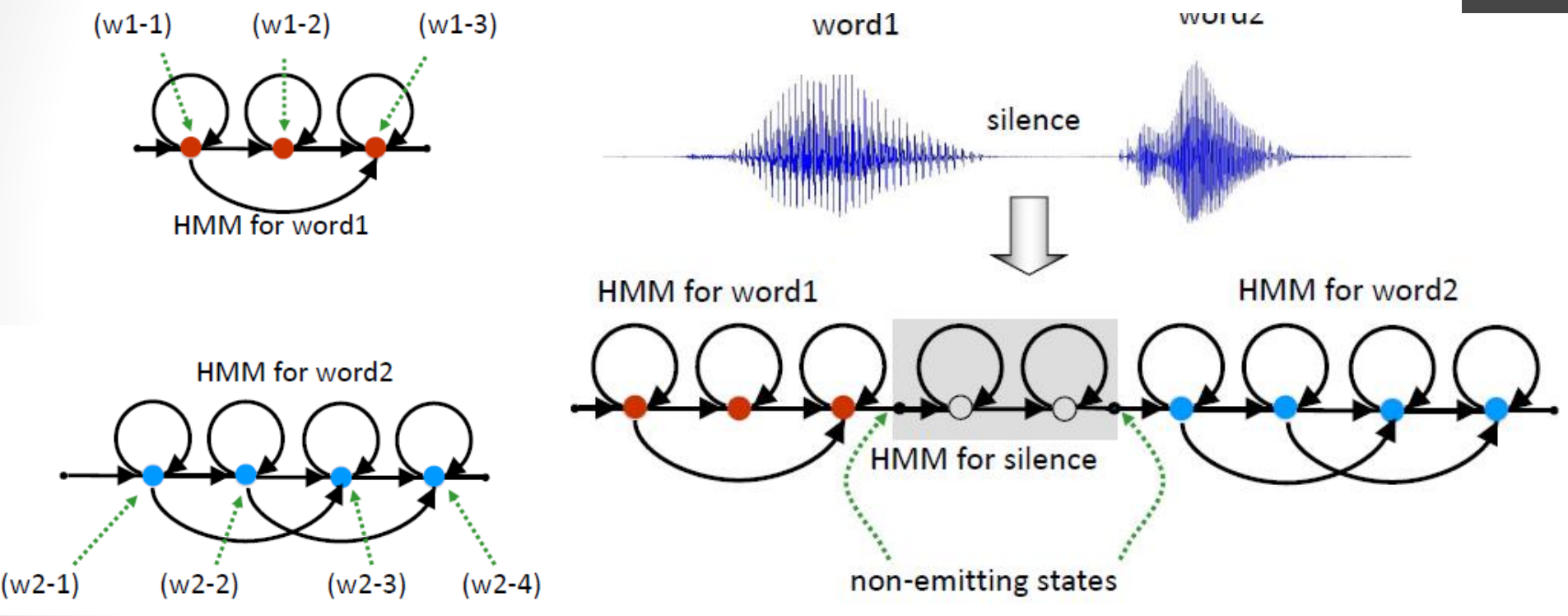


Representasi Probabilitas Output (2)

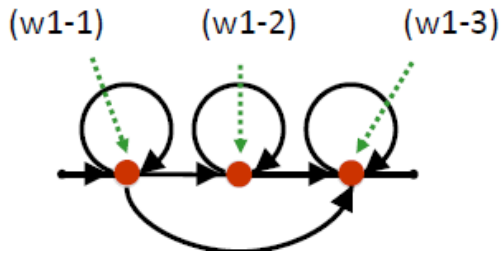
- Deep Neural-Network (6-7 layer, ratusan hingga ribuan nodes per layer)



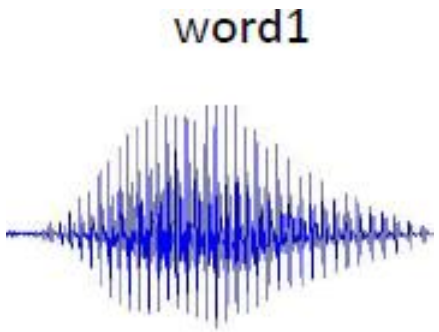
Word-based Model

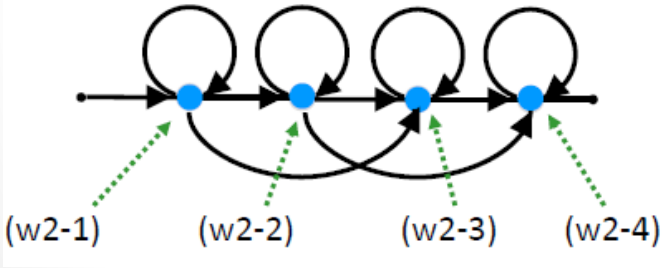


Phone-based Model

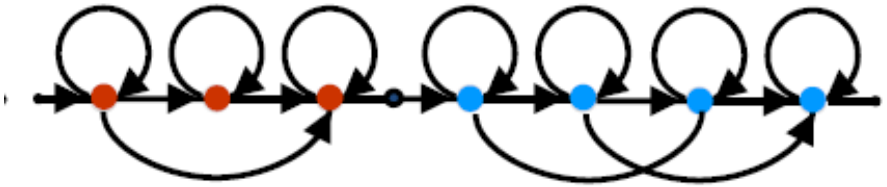


HMM for phone 1



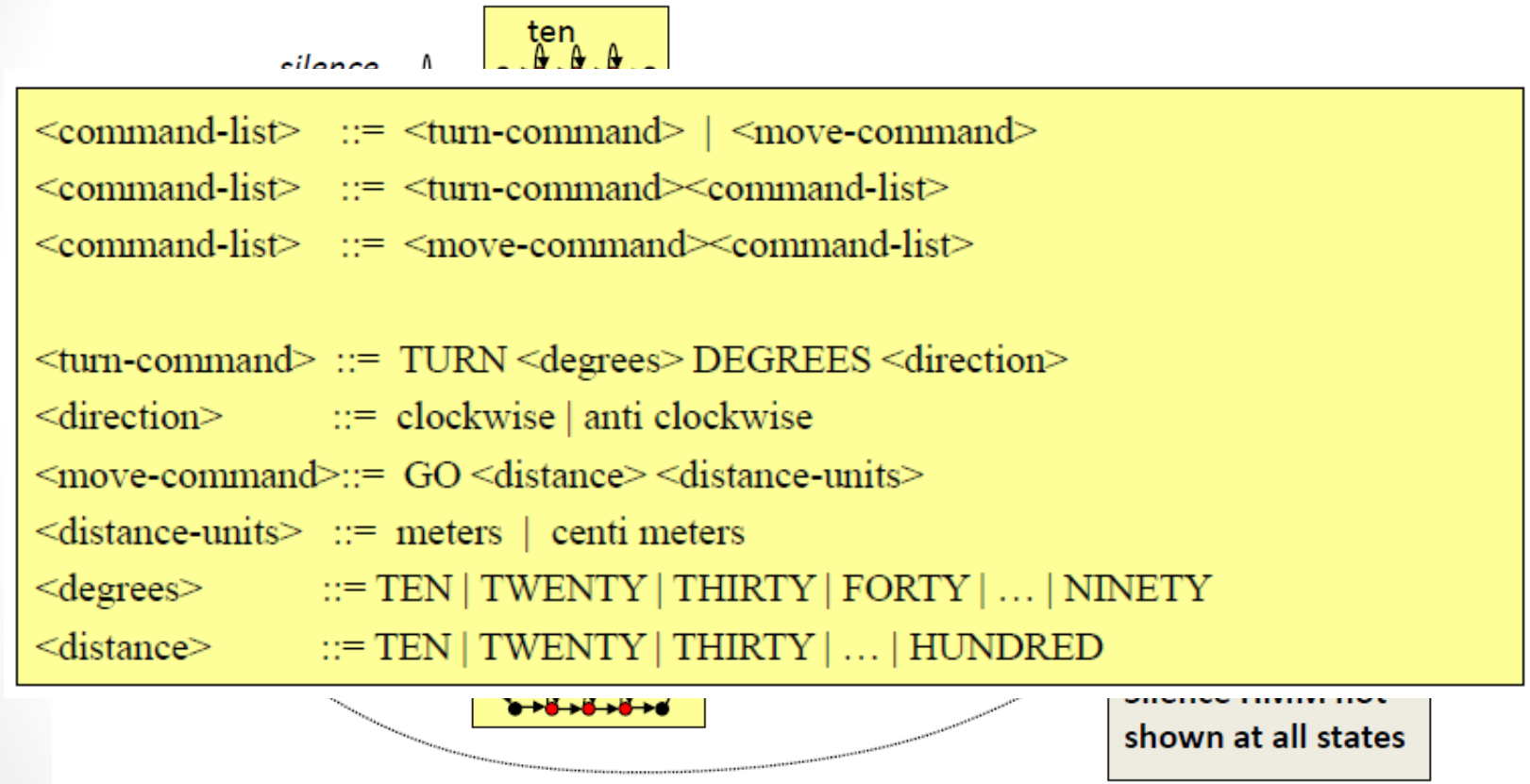


HMM for phone 2



HMM for word 1

Model Bahasa : Aturan



<command-list> ::= <turn-command> | <move-command>
<command-list> ::= <turn-command><command-list>
<command-list> ::= <move-command><command-list>

<turn-command> ::= TURN <degrees> DEGREES <direction>
<direction> ::= clockwise | anti clockwise
<move-command>::= GO <distance> <distance-units>
<distance-units> ::= meters | centi meters
<degrees> ::= TEN | TWENTY | THIRTY | FORTY | … | NINETY
<distance> ::= TEN | TWENTY | THIRTY | … | HUNDRED

Model Bahasa: Statistikal

- *P*(*W*) merupakan model bahasa yang merepresentasikan probabilitas sebuah rangkaian kata pada sebuah bahasa.
- Biasanya digunakan model n-gram.

$$P(W) = P(w_1, w_2, \ldots, w_q) = \prod_{i=1}^{q} P(w_i \mid w_{i-n+1}, \ldots, w_{i-1})$$

- *n* : order dari proses markov.
- *n* = 2 (bigrams) dan *n* =3 (trigrams) biasa digunakan.
- Untuk trigram:

$$P(w_i \mid w_{i-2}, w_{i-1}) = \frac{N(w_{i-2}, w_{i-1}, w_i)}{N(w_{i-2}, w_{i-1})}$$

Pengaruh LM terhadap Performansi ASR

Contoh ASR dengan 20K kata di dalam leksikon:

- Tanpa LM (“any word is equally likely” model):
  - AS COME ADD TAE ASIAN IN THE ME AGE OLE FUND IS MS. GROWS INCREASING ME IN TENTS MAR PLAYERS AND INDUSTRY A PAIR WILLING TO SACRIFICE IN TAE GRITTY IN THAN ANA IF PERFORMANCE

- Dengan LM yang baik (“knows” what word sequences make sense):
  - AS COMPETITION IN THE MUTUAL FUND BUSINESS GROWS INCREASINGLY INTENSE MORE PLAYERS IN THE INDUSTRY APPEAR WILLING TO SACRIFICE INTEGRITY IN THE NAME OF PERFORMANCE

Syntak dan Semantik

- Kalimat yang memiliki arti, 5K kata dalam leksikon:
  - ASR : 4.5% word error rate (WER)
  - Manusia : 0.9% WER

- Kalimat sintetis tanpa arti :
  - Contoh : *BECAUSE OF COURSE AND IT IS IN LIFE AND …*
  - ASR : 4.4% WER
  - Manusia : 7.6% WER
  - Tanpa konteks, kemampuan manusia lebih buruk

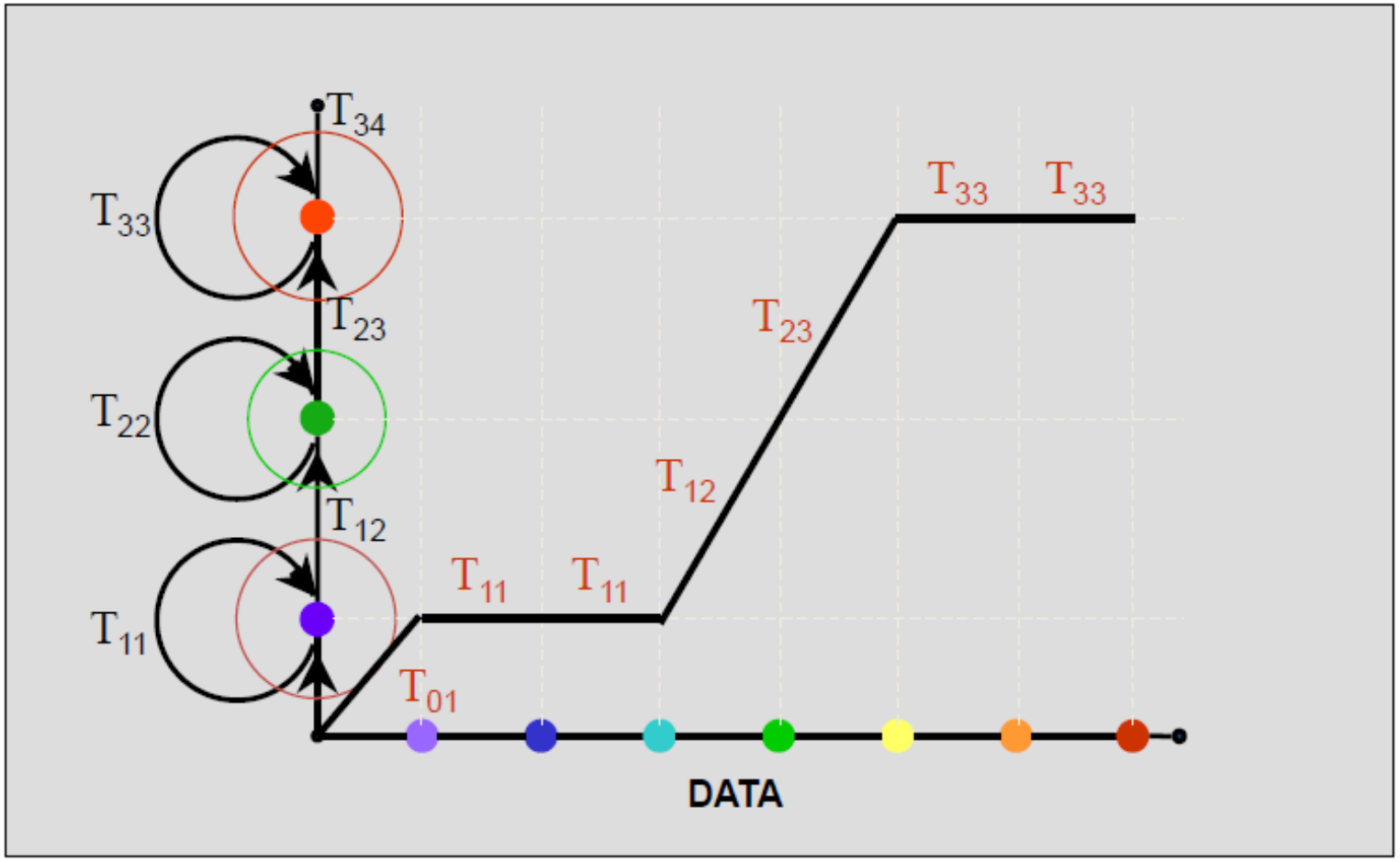
*(*dari *Spoken Language Processing*, by Huang, Acero and Hon)

Leksikon

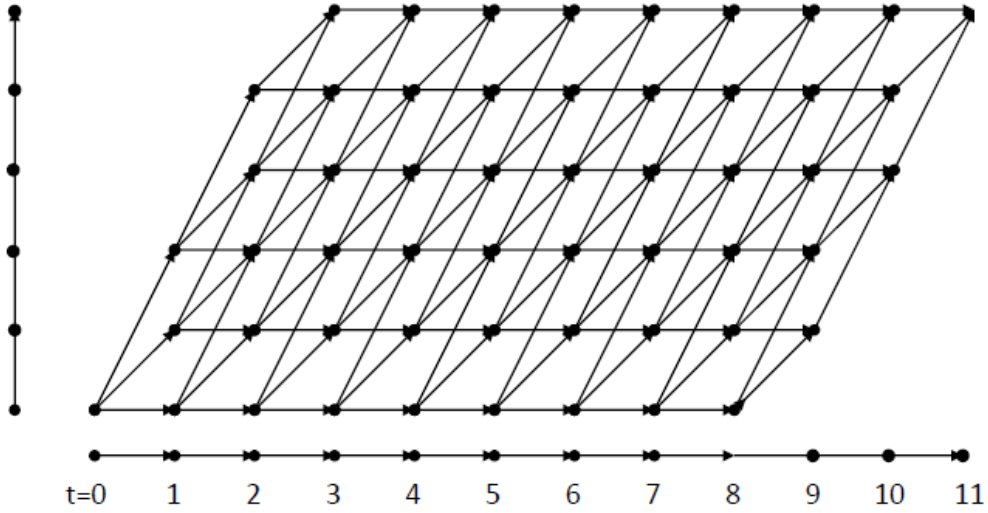
- Kamus kata yang berisi kata-kata yang dapat dikenali oleh ASR
- Format umum: <kata> < cara pelafalan : rangkaian bunyi>
- Contoh:
  - aku /a /k /u
  - suka /s /u /k /a
  - speech /s p /i /c /h

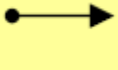
- Biasanya diperoleh dari ekstraksi kata-kata yang terdapat di dalam teks korpus berskala besar
  - Memerlukan pemrosesan teks (segmen per kalimat, normalisasi simbol-simbol, pembersihan salah ketik, dll)
  - Disesuaikan dengan domain atau sebanyak mungkin
- Pemberian pelafalan : kata dalam KBBI kanonikal, G2P

Pencarian Jawaban dengan Viterbi



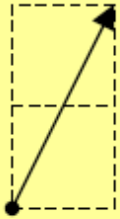
Menentukan Transisi : Trellis





The next input frame aligns to the same template frame as the previous one. (Allows a template segment to be arbitrarily stretched to match some input segment)

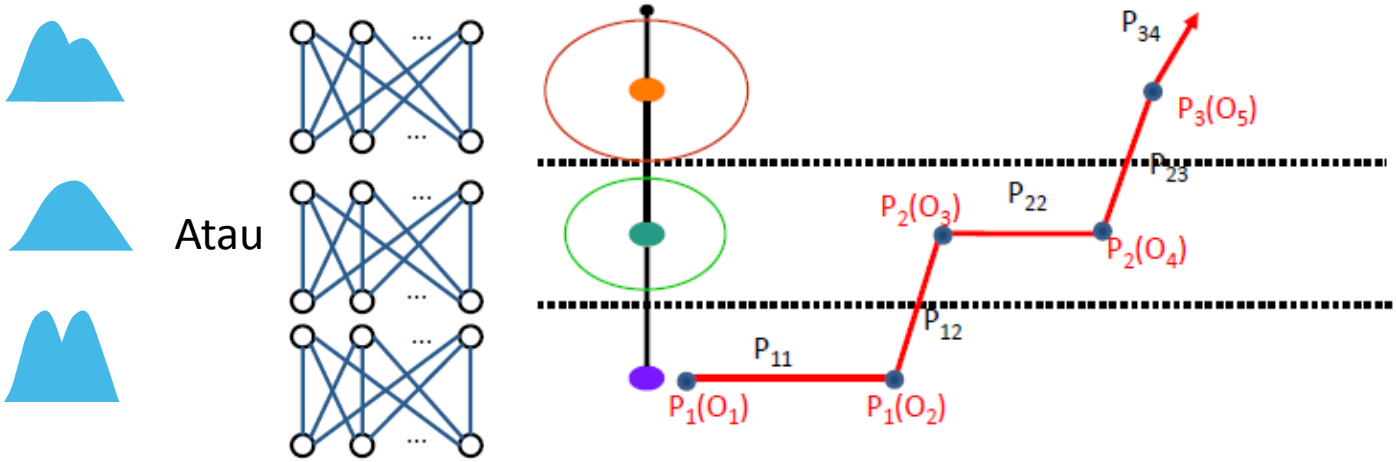
The next input frame aligns to the next template frame. No stretching or shrinking occurs in this region



The next input frame skips the next template frame and aligns to the one after that. Allows a template segment to be shrunk (by at most ½) to match some input segment

Mengukur Kemiripan: Log-Likelihood

- Representasi Probabilitas Output
  - Laplacian, Gaussian, Gaussian Mixture Model (Mixture Model paling baik)
  - NN (pseudo-likelihood)

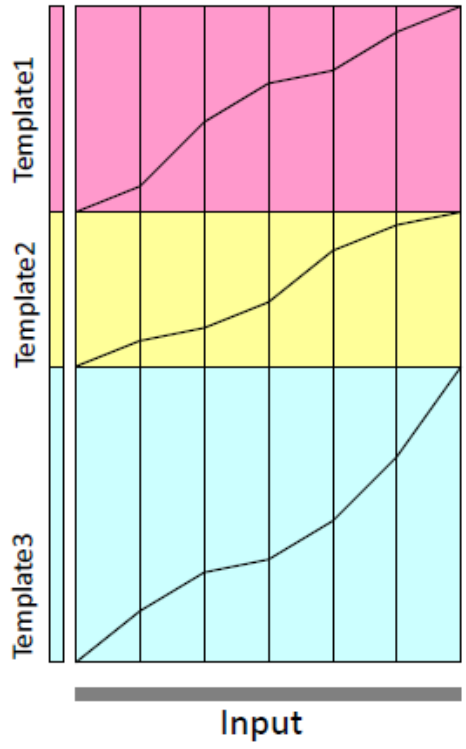


- Total likelihood dihitung dengan mengalikan probabilitas semua jalur yang dilalui:

  Product_over_nodes(cost of node) * Product_over_edges(cost of edge)
- Kebanyakan menggunakan log-likelihoods

Optimasi Pencarian

- Pencocokan semua model sekaligus Synchronously



- Beam Pruning
- Dikombinasikan dengan *Word Path-Graph* seperti WFST

Kemampuan ASR

Parameters	Range
Speaking Mode:	Isolated word to continuous speech
Speaking Style:	Read speech to spontaneous speech
Enrollment:	Speaker-dependent to speaker-independent
Vocabulary:	Small (<20 words) to large (>50,000 words)
Language Model:	Finite-state to context-sensitive
Perplexity:	Low (<10) to high (>200)
SNR:	High (>30dB) to low (<10dB)
Transducer:	Noise-canceling microphone to cell phone

Kakas Populer

- HTK
  - Ekstraksi Fitur, GMM-HMM, Adaptasi Model [MLLR-MAP], Decoder
- Julius
  - Decoder
- Sphinx
  - Ekstraksi Fitur, GMM-HMM, Decoder
- Kaldi
  - Ekstraksi Fitur, GMM-HMM++, DNN-HMM++, Decoder
- SRILM
  - Pemodelan Bahasa
- dll

Studi Kasus (dari HTK Book): Membangun **Phone Dialing** dengan HTK

- Contoh:
  - Dial three three two six five four
  - Phone Woodland
  - Call Steve Young

Steps for building ASR with Small Vocabulary

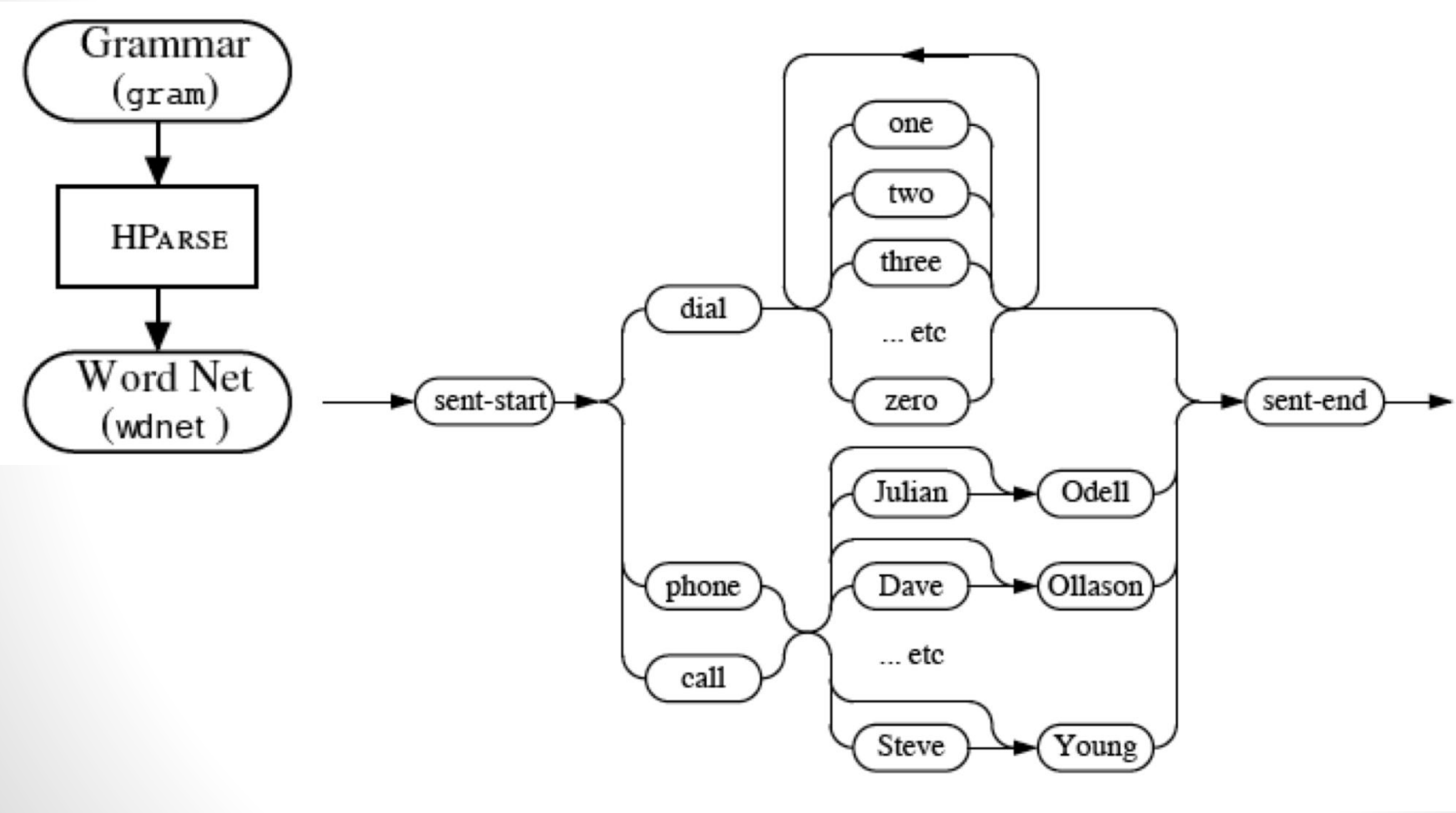
1. Develop Grammar

- Grammar:
  - $digit = ONE | TWO | THREE | FOUR | FIVE | SIX | SEVEN | EIGHT | NINE | OH | ZERO;
  - $name = [JOOP] JANSEN | [JULIAN] ODELL | [DAVE] OLLASON | [PHIL] WOODLAND | [STEVE] YOUNG;
  - (SENT-START (DIAL <$digit> | (PHONE|CALL) $name) SENT-END)

Untuk Large Vocabulary ASR gunakan LM yang dilatih dari teks korpus skala besar

Steps for building ASR with Small Vocabulary

2. Make Wordnet

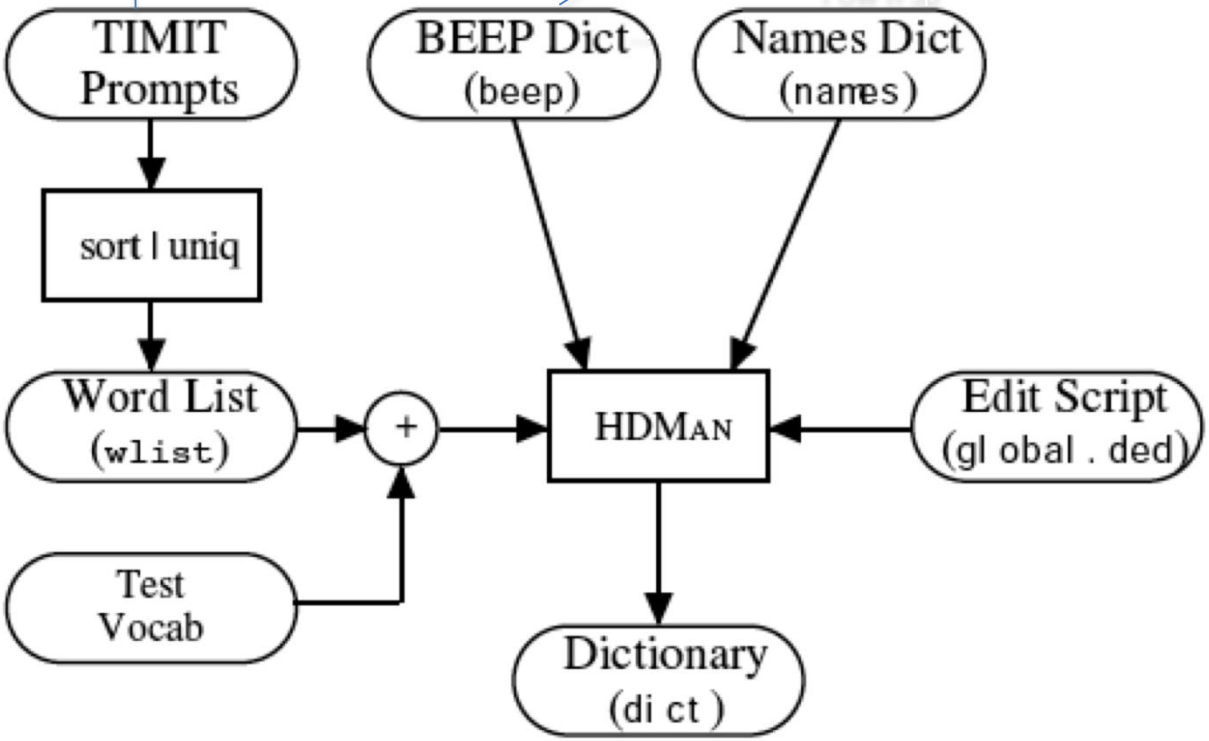


Steps for building ASR with Small Vocabulary

3. Make Lexicon/Dict

S0001 ONE VALIDATED ACTS OF SCHOOL DISTRICTS
S0002 TWO OTHER CASES ALSO WERE UNDER ADVISEMENT
S0003 BOTH FIGURES WOULD GO HIGHER IN LATER YEARS
S0004 THIS IS NOT A PROGRAM OF SOCIALIZED MEDICINE
etc

CALL ey sp
DIAL k ao l sp
d ay ax l sp



Steps for building ASR with Small Vocabulary

4. AM Modeling : Prepare Transcription File

```
#!MLF!#
"*/S0001.lab"
ONE
VALIDATED
ACTS
OF
SCHOOL
DISTRICTS
.
"*/S0002.lab"
TWO
OTHER
CASES
ALSO
WERE
UNDER
ADVISEMENT
.
"*/S0003.lab"
BOTH
FIGURES
(etc.)
```

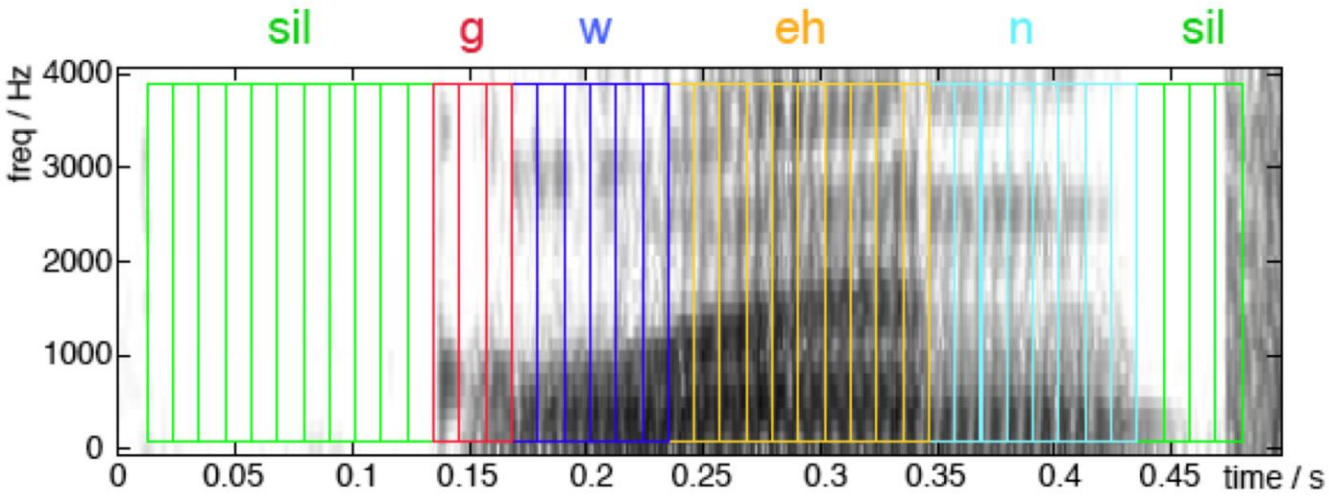
```
#!MLF!#
"*/S0001.lab"
sil
w
ah
n
v
ae
l
ih
d
.. etc
```

HTK scripting is used to generate Phonetic transcription for all training data

Steps for building ASR with Small Vocabulary

4. AM Modeling : Prepare Speech File

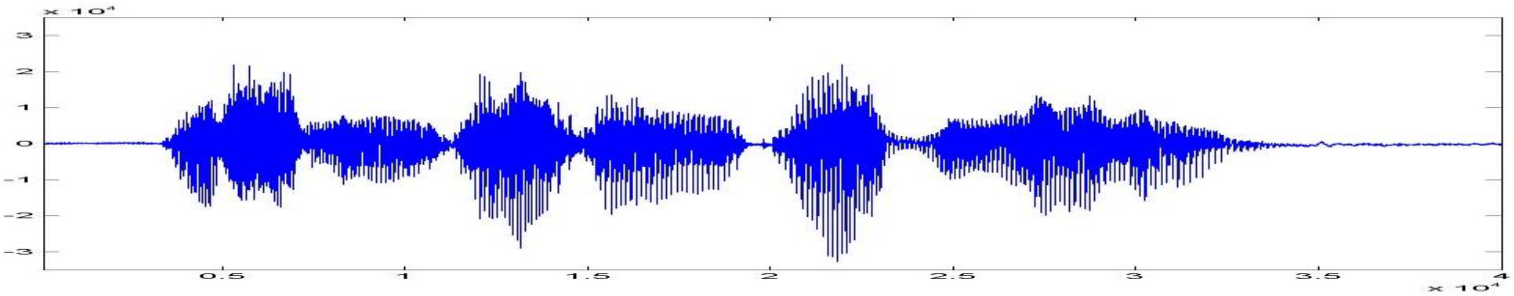
- For each transcription file, prepare speech (wave) file
  - Used already available speech corpus
  - Develop your own speech corpus

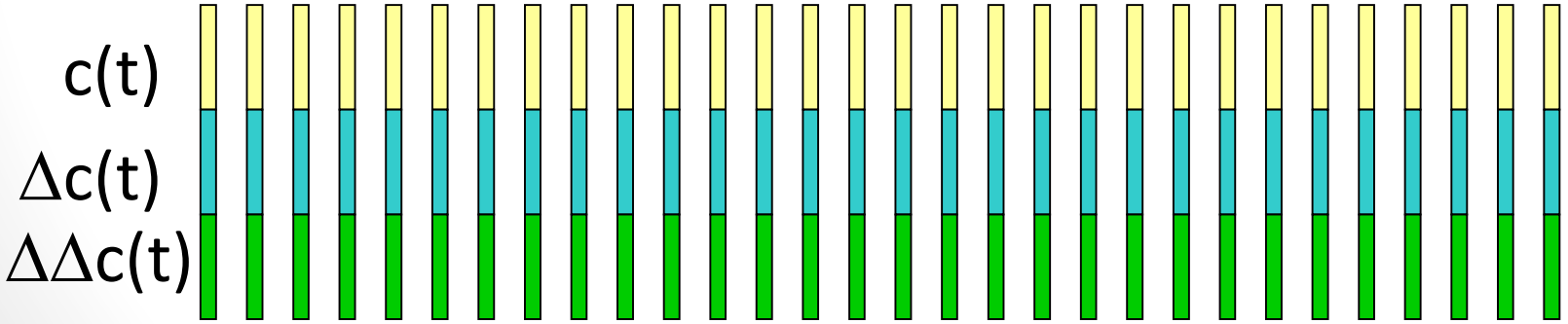


Steps for building ASR with Small Vocabulary

4. AM Modeling : Extracting Features (MFCC)

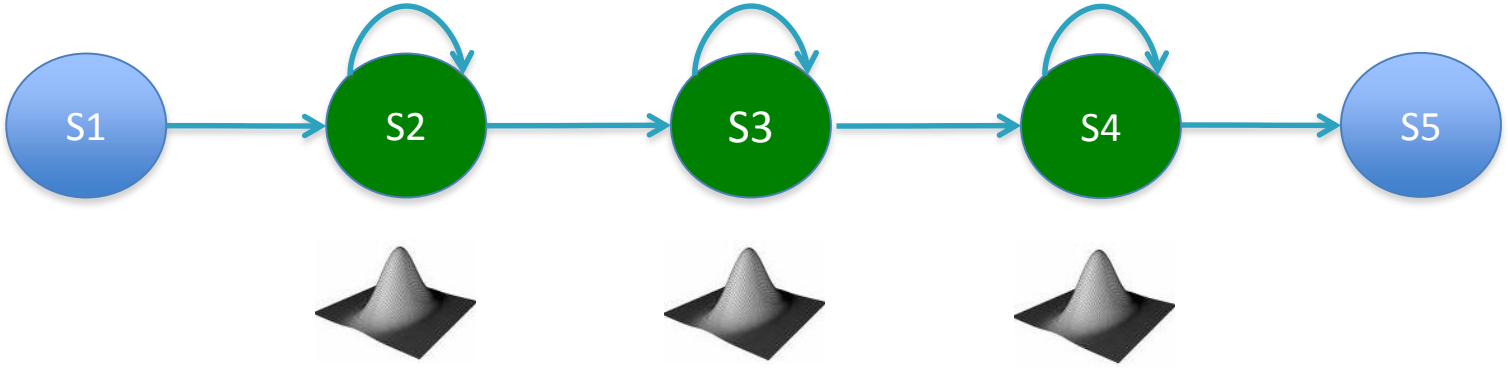
- For each wave file, extract MFCC features and save it as a .mfc file
- Konfigurasi Sampling dan Dimensi Fitur diset di file konfigurasi





4. AM Modeling :
Create Monophone HMM Topology

- 5 states: 3 emitting states



- Flat Start: Mean and Variance are initialized as the global mean and variance of all the data

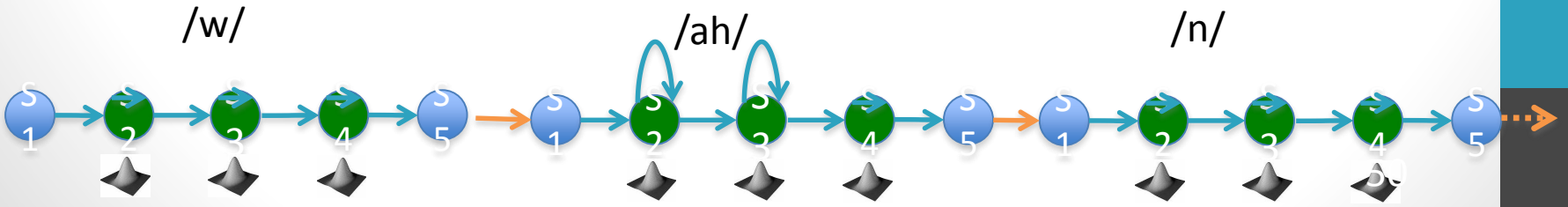
4. AM Modeling : Monophone Training

For each training pair of files (mfc+lab):

1. concatenate the corresponding monophone HMMS
2. Use the Baum-Welch Algorithm to train the HMMS given the MFC features.

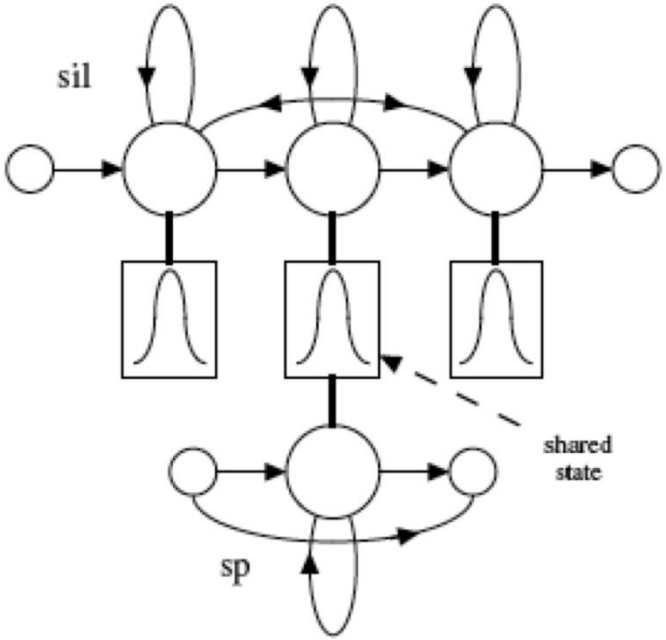
```
#!MLF!#
"*/S0001.lab"
sil
w
ah
n
v
ae
l
ih
d
.. etc
```

ONE VALIDATED ACTS OF SCHOOL DISTRICTS



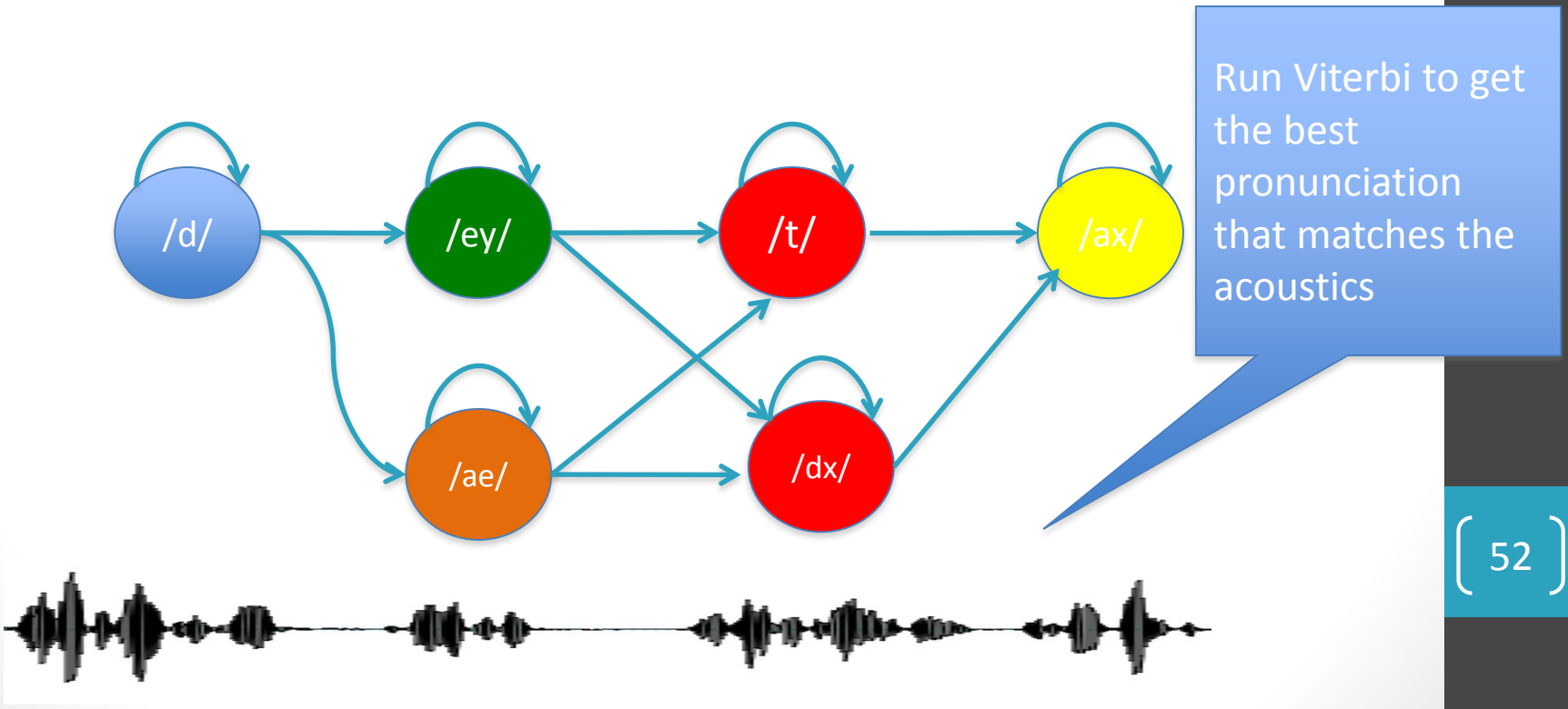
4. AM Modeling : Train Short Pause Model

- So far, we have all monophone models trained
- Next, we have to train the *sp* (short pause) model



4. AM Modeling : Forced alignment

- The dictionary may contains multiple pronunciations for some words.
- Realignment the training data



4. AM Modeling : Retrain Monophone

- After getting the best pronunciation

  => Train again using Baum-Welch algorithm using the “correct” pronunciation for 5 – n times to get model convergent.

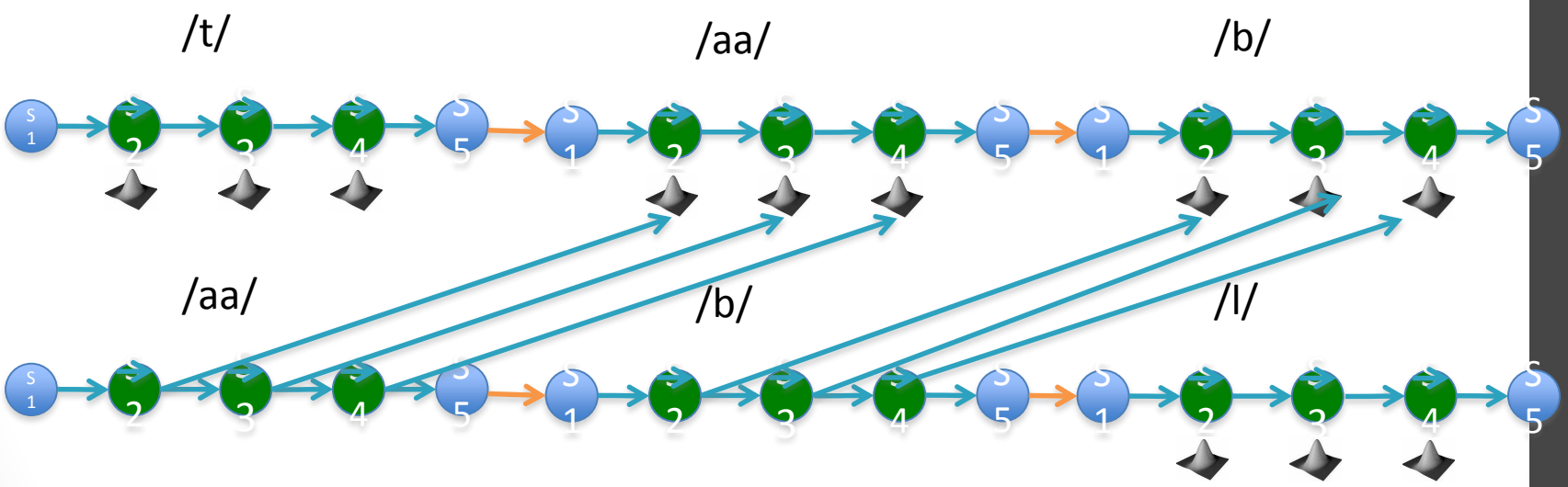
4. AM Modeling : Creating Triphone models

- Context dependent HMMs
- Make Tri-phones from monophones
  - Generate a list of all the triphones for which there is at least one example in the training data

    - jh-oy+s
    - oy-s
    - ax+z
    - f-iy+t
    - iy-t
    - s+l
    - s-l+ow

4. AM Modeling :
Creating Tied-Triphone models

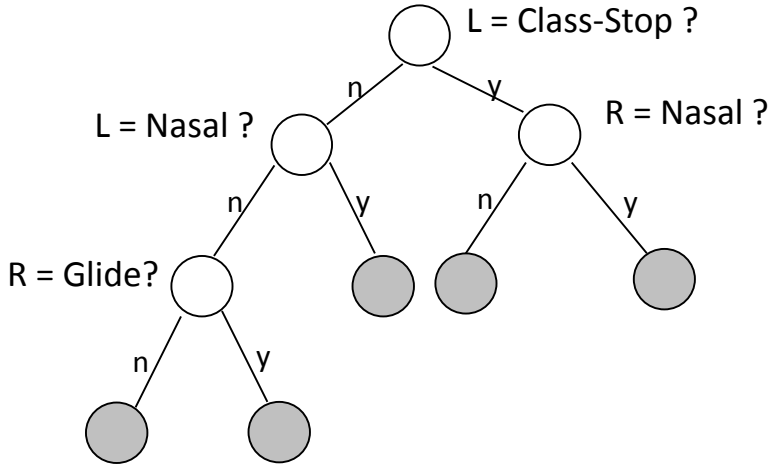
Data insufficiency => Tie states



4. AM Modeling : Phone Clustering

- Data Driven Clustering: Using similarity metric
- Clustering using Decision Tree.
  - All states in the same leaf will be tied

t+ih	
	t+ae
t+iy	
	t+ae
ao-r+ax	r
t+oh	t+ae
ao-r+iy	
t+uh	t+ae
t+uw	t+ae
sh-n+t	
sh-n+z	sh-n+t
ch-ih+l	
ay-oh+l	
ay-oh+r	ay-oh+l



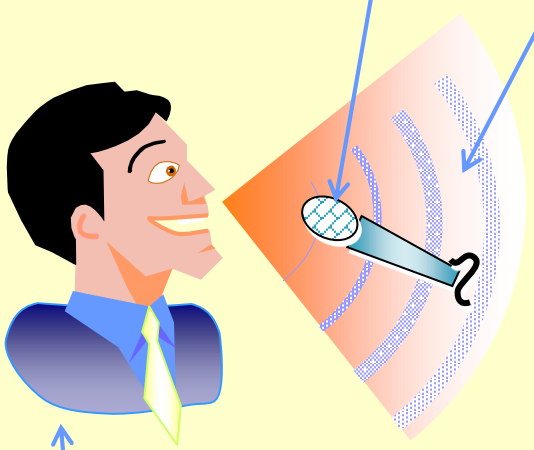
4. AM Modeling :
Retrain Tied-Triphone

- Train the Acoustic models again using Baum algorithm until convergent (usually 5 times)
- Using the grammar network for the phones, generate the triphone HMM grammar network WNET

Decoding

- Given a new Speech file, extract the mfcc features
  - Use the same configuration as training to get optimal result
- Run Viterbi on the WNET given the mfcc features to get the best word sequence.

Tantangan-tantangan (Furui, 2010)



Microphone
- Distortion
- Electrical noise
- Directional characteristics

Noise
- Other speakers
- Background noise
- Reverberations

Distortion
Noise
Echoes
Dropouts

Channel

Speech recognition system

Speaker
- Voice quality
- Pitch
- Gender
- Dialect

Speaking style
- Stress/Emotion
- Speaking rate
- Lombard effect

Task/Context
- Man-machine dialogue
- Dictation
- Free conversation
- Interview

Phonetic/Prosodic context

Variations in Speech

Terima Kasih

Referensi Utama

- Thomas, F. Quatieri, Discrete-Time Speech Signal Processing: Principles and Practice, Prentice Hall, 2001, Chapter 3
- L. R. Rabiner and R.W. Schafer, Theory and Applications of Digital Speech Processing, Prentice-Hall Inc., 2011
- Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, *Spoken Language Processing,* Prentice Hall PTR, 2001.
- Bhiksha Raj, Rita Singh, Design and Implementation of Speech Recognition Systems, Carnegie Mellon, School of Computer Science, 2011 (where many slides were taken from)
- An Introduction to Speech Recognition, B. Plannerer , 2005.
- Automatic Speech Recognition: Trials, Tribulations, Triumphs and, Sadaoki Furui, 2012.
- HTK Book