9. Automatic Speech Recognition

(some slides taken from Glass and Zue course)
What is the task?

Getting a computer to understand spoken language
By “understand” we might mean
  React appropriately
  Convert the input speech into another medium, e.g. text
How do humans do it?

Articulation produces sound waves which the ear conveys to the brain for processing.
How might computers do it?

Digitization
Acoustic analysis of the speech signal
Linguistic interpretation

Acoustic waveform
Acoustic signal

Speech recognition
Challenges in ASR processing

- Inter-speaker variability
  - Vocal tract, gender, dialects
- Language variability
  - From isolated words to continuous speech
  - Out-of-vocabulary words
- Vocabulary size and domain
  - From just a few words (e.g. Isolated numbers) to large vocabulary speech recognition
  - Domain that is being recognized (medical, social, engineering, ...)
- Noise
  - Convolutive: recording/transmission conditions, reverberation
  - Additive: recording environment, transmission SNR
  - Intra-speaker variability: stress, age, humor, changes of articulation due to environment influence, ...
Approaches to ASR

- The acoustic-phonetic approach
- The pattern recognition approach
- Statistics-based approach

General block diagram of a task-oriented dialog (speech input-output) system.
Typology of ASR systems

Several ASR systems can be developed, depending on:

- Speaker-dependent vs. independent
- Language constraints:
  - isolated word recognition
  - connected word recognition
  - Keyword spotting
  - continuous speech recognition
- Robustness constraints:
  - laboratory (office) conditions: imposed microphone, no ambient noise
  - (quiet) telephone system
  - real-life (human-like) ASR …
Acoustic-phonetic approach to ASR

Also called rule-based approach

acoustic-phonetic speech-recognition system.
Acoustic phonetic approach

Use knowledge of phonetics and linguistics to guide search process

Usually some rules are defined expressing everything (anything) that might help to decode:
   Phonetics, phonology, phonotactics
   Syntax
   Pragmatics

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
Identify individual phonemes
Identify words
Identify sentence structure and/or meaning
Interpret prosodic features (pitch, loudness, length)
Acoustic-phonetic example: vowel classifier
Acoustic-phonetic example 2: speech sound classifier

1DP: Speech Recognition
Pattern-recognition speech recognition

- **Feature measurement**: Filter Bnk, LPC, DFT, ...
- **Pattern training**: Creation of a reference pattern derived from an averaging technique
- **Pattern classification**: Compare speech patterns with a local distance measure and a global time alignment procedure (DTW)
- **Decision logic**: Similarity scores are used to decide which is the best reference pattern.
Template Matching Mechanism

- Test pattern, $T$, and reference patterns, $\{R_1, \ldots, R_V\}$, are represented by sequences of feature measurements.

- Pattern similarity is determined by aligning test pattern, $T$, with reference pattern, $R_V$, with distortion $D(T, R_V)$.

- Decision rule chooses reference pattern, $R^*$, with smallest alignment distortion $D(T, R^*)$

$$R^* = \arg\min_V D(T, R_V)$$

- Dynamic time warping (DTW) is used to compute the best possible alignment warp, $\phi_V$, between $T$ and $R_V$, and the associated distortion $D(T, R_V)$. 

TDP: Speech Recognition 14
Alignment Example
Dynamic Time Warping (DTW)

- **Objective:** an optimal alignment between variable length sequences $T = \{t_1, \ldots, t_N\}$ and $R = \{r_1, \ldots, r_M\}$

- The overall distortion $\mathcal{D}(T, R)$ is based on a sum of local distances between elements $d(t_i, r_j)$

- A particular alignment warp, $\phi$, aligns $T$ and $R$ via a point-to-point mapping, $\phi = (\phi_t, \phi_r)$, of length $K_\phi$

  $$t_{\phi_t(k)} \Leftrightarrow r_{\phi_r(k)} \quad 1 \leq k \leq K_\phi$$

- The optimal alignment minimizes overall distortion

  $$\mathcal{D}(T, R) = \min_{\phi} \mathcal{D}_\phi(T, R)$$

  $$\mathcal{D}_\phi(T, R) = \frac{1}{M_\phi} \sum_{k=1}^{K_\phi} d(t_{\phi_t(k)}, r_{\phi_r(k)}) m_k$$
DTW Issues

- Endpoint constraints:
  \[
  \phi_t(1) = \phi_r(1) = 1 \quad \phi_t(K) = N \quad \phi_r(K) = M
  \]

- Monotonicity:
  \[
  \phi_t(k + 1) \geq \phi_t(k) \quad \phi_r(k + 1) \geq \phi_r(k)
  \]

- Path weights, \( m_k \), can influence shape of optimal path

- Path normalization factor, \( M_\phi \), allows comparison between different warps (e.g., with different lengths)
  \[
  M_\phi = \sum_{k=1}^{K_\phi} m_k
  \]
Statistics-based approach

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

Sometimes seen as “anti-linguistic” approach

Fred Jelinek (IBM, 1988): “Every time I fire a linguist my system improves”

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
Machine learning

Acoustic and Lexical Models

- Analyse training data in terms of relevant features
- Learn from large amount of data different possibilities
  - different phone sequences for a given word
  - different combinations of elements of the speech signal for a given phone/phoneme
- Combine these into a Hidden Markov Model expressing the probabilities
HMMs for some words

- Word model for "the"
- Word model for "need"
- Word model for "on"
- Word model for "I"
The usage of language models

To make speech recognition a bit more robust, some information on the probability of certain words occurring next to each other is used. This is what a language model does.

Language models can be statistically trained from lots of data or hand-made for particular tasks.

LM Model the likelihood of each word given previous word(s).

Usually we use n-gram models:

Build the model by calculating bigram (groups of 2 words) or trigram (groups of 3 words) probabilities from a text training corpus.

$$\arg\max_{\text{wordsequence}} P(\text{wordsequence} | \text{acoustics}) =$$

$$\arg\max_{\text{wordsequence}} \frac{P(\text{acoustics} | \text{wordsequence}) \times P(\text{wordsequence})}{P(\text{acoustics})}$$
Knowledge integration for speech recognition: Bottom-up
Example of Speech Recognition Architecture