Speech Recognition and Processing

Lecture 5

Yossi Keshet
Agenda

- ASR (Automatic Speech Recognition) Structure
- Phonemes
- Acoustic Model, Language Model, Pronunciation Model
- Language Model
- Acoustic Model
Recall

- Up until now we use the following decoding process:

\[ c^* = \arg\min_{c \in Vocab} \text{distance}(A_{test}, A_c) \]

- Where we set \text{distance} to be the DTW alignment score
Limitations

- The above approach has many drawbacks:
  - Classifying whole words
  - Limited dictionary size
  - Lazy method
  - Non-parametric, Non-probabilistic
  - Etc.
Can we do better?
Limitations

• The above approach has many drawbacks:
  
  • Classifying whole words
  
  • Limited dictionary size
  
  • Lazy method
  
  • Non-parametric, Non-probabilistic
  
  • Etc.
Classifying whole words

- Cons:
  - Words usually span over many frames
  - Need to split audio files to words, a hard task
  - Classifying words will force us to use multi-class classification
    - Extreme classification (~200K+ words)
    - Limited dictionary
  - Spoken language tend to change — language is evolving
Can we do better?

- Can we break words into smaller sub-units?
- Option 1: Characters
  - Often not represent the acoustic signal (know, lough, musically, etc.)
  - Different pronunciations for the same word (accents, slang, etc.)
- Option 2: Phonemes
Can we do better?

- Can we break words into smaller sub-units?
- Option 1: Characters
  - Often not represent the acoustic signal (know, musically, etc.)
  - Different pronunciations for the same word (accents, slang, etc.)
- Option 2: Phonemes
Phonemes
Phonetics

• **Phonetics:** Study of the physical properties of speech-sounds

• **Articulatory Phonetics:** how speech-sounds are made

• **Auditory Phonetics:** how speech-sounds are heard

• **Acoustic Phonetics:** how speech-sounds are transmitted
Phonetics

- **Phones**: distinct speech sound or gesture, regardless of whether the exact sound is critical to the meanings of words

- **Phoneme**: a sound or a group of different sounds perceived to have the same function by speakers of the language

  - For example: phoneme /k/ -> cat, kit, scat, skit

- **Different standards**: IPA, SAMBA

- **Allophone**: The different phonetic realizations of a phoneme
Phonetics

Partial reproduction of the International Phonetic Alphabet ©2005

Visual chart by Jonathan Coombs (sils.org); keyboard shortcuts are for the MSKLC IPA keyboard at scripts.sils.org/UniIPAKeyboard
Phonetics

Vowels at right & left of bullets are rounded & unrounded.
# Phonetics

<table>
<thead>
<tr>
<th>PHONEME</th>
<th>EXAMPLE</th>
<th>PHONEME</th>
<th>EXAMPLE</th>
<th>PHONEME</th>
<th>EXAMPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>/æ/</td>
<td>bat</td>
<td>/e/</td>
<td>bet</td>
<td>/i/</td>
<td>bit</td>
</tr>
<tr>
<td>/ɛ/</td>
<td>bait</td>
<td>/ɛ/</td>
<td>bat</td>
<td>/ɛ/</td>
<td>Bob</td>
</tr>
<tr>
<td>/ɑ/</td>
<td>/ɑ/</td>
<td>/ɑ/</td>
<td>/ɑ/</td>
<td>/ɑ/</td>
<td>/ɑ/</td>
</tr>
<tr>
<td>/ɔ/</td>
<td>boat</td>
<td>/ʌ/</td>
<td>but</td>
<td>/ɔ/</td>
<td>book</td>
</tr>
<tr>
<td>/ʌ/</td>
<td>bought</td>
<td>/ɒ/</td>
<td>boat</td>
<td>/o/</td>
<td>book</td>
</tr>
<tr>
<td>/ɜ/</td>
<td>Burt</td>
<td>/ʊ/</td>
<td>boot</td>
<td>/ɛ/</td>
<td>boot</td>
</tr>
<tr>
<td>/ɔ/</td>
<td>Boyd</td>
<td>/ɔ/</td>
<td>bout</td>
<td>/ɔ/</td>
<td>about</td>
</tr>
<tr>
<td>/ə/</td>
<td>/ə/</td>
<td>/ə/</td>
<td>/ə/</td>
<td>/ə/</td>
<td>/ə/</td>
</tr>
</tbody>
</table>
Limitations

- The above approach has many drawbacks:
  - Classifying whole words
  - Limited dictionary size
  - Lazy method
  - Non-parametric, Non-probabilistic
  - Etc.

Classify Phonemes
<table>
<thead>
<tr>
<th>Word</th>
<th>Transcription</th>
</tr>
</thead>
<tbody>
<tr>
<td>odd</td>
<td>AA D</td>
</tr>
<tr>
<td>at</td>
<td>AE T</td>
</tr>
<tr>
<td>cow</td>
<td>K AW</td>
</tr>
<tr>
<td>be</td>
<td>B IY</td>
</tr>
<tr>
<td>cheese</td>
<td>CH IY Z</td>
</tr>
<tr>
<td>eat</td>
<td>IY T</td>
</tr>
<tr>
<td>read</td>
<td>R IY D</td>
</tr>
</tbody>
</table>
How can we handle wrong phonetic sequences?

Wrong prediction

Wrong pronunciation
ASR Structure
ASR Structure

• Automatic Speech Recognition

• **Acoustic Model:** Represent the relationship between audio signal (features) to phonemes

• **Language Model:** Compute the probability to observe a given sequence of words

• **Pronunciation Model:** Builds a lexicon for how people pronounce words
ASR Structure

- How can we combine all these components?

- Let’s be more formal:
  - Given a sequence of acoustic features: $\tilde{d}$
  - We would like to set:

$$\tilde{w}^* = \arg\max_{\tilde{w}} P(\tilde{w} | \tilde{d})$$
ASR Structure

\[ \tilde{w}^* = \arg \max_{\tilde{w}} P(\tilde{w} | \tilde{o}) \]

\[ = \arg \max_{\tilde{w}} \frac{P(\tilde{o} | \tilde{w}) \cdot P(\tilde{w})}{P(\tilde{o})} \]

\[ = \arg \max_{\tilde{w}} P(\tilde{o} | \tilde{w}) \cdot P(\tilde{w}) \]
ASR Structure

\[
P(\tilde{o} \mid \tilde{w}) = \sum_{\tilde{p}} P(\tilde{o} \mid \tilde{p}, \tilde{w}) \cdot P(\tilde{p} \mid \tilde{w})
\]

\[
\approx \max_{\tilde{p}} P(\tilde{o} \mid \tilde{p}, \tilde{w}) \cdot P(\tilde{p} \mid \tilde{w})
\]

\[
\approx \max_{\tilde{p}} P(\tilde{o} \mid \tilde{p}) \cdot P(\tilde{p} \mid \tilde{w})
\]
ASR Structure

\[ \tilde{w}^* = \arg \max_{\tilde{w}} \max_{\tilde{p}} P(\tilde{o} | \tilde{p}) \cdot P(\tilde{p} | \tilde{w}) \cdot P(\tilde{w}) \]
\[
\tilde{w}^* = \arg \max_{\tilde{w}} \max_{\tilde{p}} P(\tilde{o} | \tilde{p}) \cdot P(\tilde{p} | \tilde{w}) \cdot P(\tilde{w})
\]
In language modeling we would like to predict the probability of a given sequence of tokens $P(\vec{w})$.

We can be factored this probability using the chain rule of probability and we get:

$$P(\vec{w}) = \prod_{i=1}^{M} P(w_i \mid w_1, \ldots, w_{i-1})$$

We usually assume that given the history of the previous $n$-1 tokens, each token is independent of the remaining history.

$$P(\vec{w}) = \prod_{i=1}^{M} P(w_i \mid w_{i-n+1}, \ldots, w_{i-1})$$
Language Model

• This can be implemented using different ML models:
  • N-grams
  • RNN
  • CNN
  • Etc.
ASR Structure
We would like to compute the probability: $P(\bar{o} | \bar{p})$

Consider the phone sequence $[p\ r\ aa\ b\ l\ iy]$

Corresponds to some spoken conversational speech segment of 320ms

There are 32 observations in such segment (each frame is 10ms long)

The phone sequence can be written as an expended sequence of 32 phone symbols $\bar{q} = [p, p, r, r, \ldots, r, aa, aa, \ldots, b, b, b, l, \ldots, iy, iy, iy]$
Markov Model

- Given $N$ states: $S_1, S_2, \ldots, S_N$

- In each time instant, $t = 1, 2, \ldots, T$ a system changes (makes a transition) to state $q_t$

- For a special case of a first order Markov chain

\[
P(q_t = S_j | q_{t-1} = S_i, q_{t-2} = S_k, \ldots) = P(q_t = S_j | q_{t-1} = S_i)
\]

- This usually refers as the state transition probability
Acoustic Model

- We can use this Markovian assumptions to reduce the complexity of our problem

\[
P(\tilde{\omega} | \tilde{q})P(\tilde{q}) = \prod_{i=1}^{T} P(o_t | q_t)P(q_t | q_{t-1})
\]

Probability of observing an acoustic features given a phoneme

Transition probability
Acoustic Model

- We can compute $P(o_t \mid q_t)$ using different models:
  - Gaussian Mixture Model (GMM)
  - Support Vector Machines
  - Deep Neural Networks (DNNs)
Deep Neural Networks

• We would like to find the parameters that minimize the following Negative Log Likelihood loss function:

\[ L(\theta) = \sum_t -\log(p(y = y_t | x_t)) \]

• In order to do so, we calculate \( \frac{\partial L}{\partial \theta} \) and use GD/SGD

• Notice, now we have several weight matrices and biases we need to find the gradients to all of them
Deep Neural Networks

\[ f(x) = g_n(g_{n-1}(\ldots g_2(g_1(x))))) \]

\[ g_1(x) = \sigma(w_1x + b_1) = h_1 \]

\[ g_2(x) = \sigma(w_2h_1 + b_2) = h_2 \]

\[ \ldots \]

\[ g_n(h_{n-1}) = \sigma(w_nh_{n-1} + b_n) = h_n \]

\[ p(y|x) = \text{softmax}(h_n) \]
Deep Neural Networks

\[ f(x) = g_n(g_{n-1}(\ldots g_2(g_1(x)))) \]

each layer learns different representation
Deep Neural Networks

- We use as input the Spectrogram / MFCC features

- **Problem:** However, in a single feature we do not have much information about the pronounced phoneme

- **Solution:** Concatenate for each frame k frames before and after (we usually set k=5,9,11)

- Note, we will still classify the phoneme in the center frame
Deep Neural Networks

• DNNs optimization is often hard

• Practically researchers often use unsupervised techniques as a pre-train step

  • Denoising Auto-Encoder

  • Restricted Boltzman-Machines

  • Deep Belief Networks

  • Etc.
Deep Neural Networks

- After training, the network will have outputs of the form $P(\bar{p}_t | \bar{o}_t)$
  For each frame

- Recall, we would like use the DNN to output $P(\bar{o}_t | \bar{q}_t)$

- We can use Bayes theorem to obtain: $P(\bar{o}_t | \bar{q}_t) = \frac{P(\bar{q}_t | \bar{o}_t) \cdot P(\bar{o}_t)}{P(\bar{q}_t)}$

- $P(\bar{o}_t)$ is unknown
  - All alignments will scale by the same factor so no effect

- We can assume a uniform prior over the phonemes or computer this probability
Acoustic Model

- Notice, we assume we have an alignment between the phonemes and the speech signal.

- Practically, to obtain $P(\tilde{o} | \tilde{p}) = \sum_{a \in A} \prod_{t=1}^{T} P(\tilde{o}_t | \tilde{q}_t^a) \cdot P(\tilde{q}_t^a | \tilde{q}_{t-1}^a)$

- To compute that we need to use DP algorithm called the Forward-Backward algorithm (next lesson)
Questions?