Automatic Speech Recognition [1]

Spoken language understanding is a difficult task, and it is remarkable that humans doas well at it as we do. The goal of automatic speech recognition (ASR) research is to address this problem computationally by building systems that map from an acoustic signal to a string of words. Automatic speech understanding (ASU) extends this goal to producing some sort of understanding of the sentence, rather than just the words

1. application

- human-computer interaction (hands-busy or eyes-busy applications)
- telephony (in call-routing)
- dictation

Speech recognition is easier if the number of distinct words we need to recognize is smaller

2. The difficulty of Automatic Speech Recognition

- The vocabulary size.
- how fluent, natural, or conversational the speech is
- Isolated word
- channel and noise
- accent or speaker-class characteristics

Large-Vocabulary Continuous Speech Recognition (LVCSR) Large vocabulary generally means that the systems have a vocabulary of roughly 20,000 to 60,000 words

3. Speech Recognition Architecture

The Hidden Markov Model (HMM) based speech recognition systems view the task of taking an acoustic waveform and produce a string words as output as denoising the input version (acoustic waveform) of the words. The first step of this procedure is to model the noisy channel. In this case the speech recognition will be consider as special case of bayesian inference. We need to search in the search space efficiently to fine the sentences that have good chance of maching the input.

3.1 formulation

- **O**: the acoustic input is a sequence of individual symbols or observation (ex. sample evry 10 ms and represent the samples with energy or frequencies values).

$$O = o_1, o_2, o_3, ..., o_t$$

- **W**: the sentence composed of string of words.

$$W = w_1, w_2, w_3, ..., w_n$$

- The estimation of the sentce is given by:

$$\hat{W} = \underset{w \in \mathbf{L}}{\operatorname{argmax}} P(w/O)$$

- based on the Baye's rule we have:

$$\hat{W} = \underset{w \in \mathbf{L}}{\operatorname{argmax}} \frac{P(O/W)P(W)}{P(O)}$$

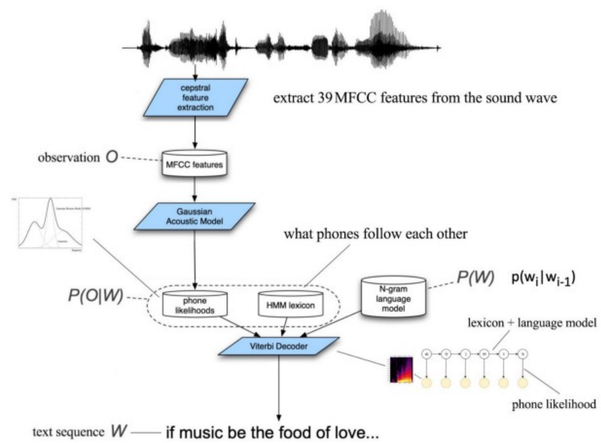
Since we are maximizing over all possible sentences P(0) dosen't change for each sentence.Thus.

$$\hat{W} = \underset{w \in \mathbf{L}}{\operatorname{argmax}} P(O/W)P(W)$$

with $P(O/W)$:the likelihood and $P(W)$ the prior.

- Based on th N-gram grammar

$$P\left(w_1^n\right) \approx \prod_{k=1}^{n} P\left(w_k \mid w_{k-N+1}^{k-1}\right)$$



Schematic architecture for a simplified speech recognition [1]

References

[1] *Speech and Language Processing*. URL: https://web.stanford.edu/~jurafsky/slp3/ (visited on 11/13/2020).