Spoken language understanding: Speech recognition

Milica Gašić

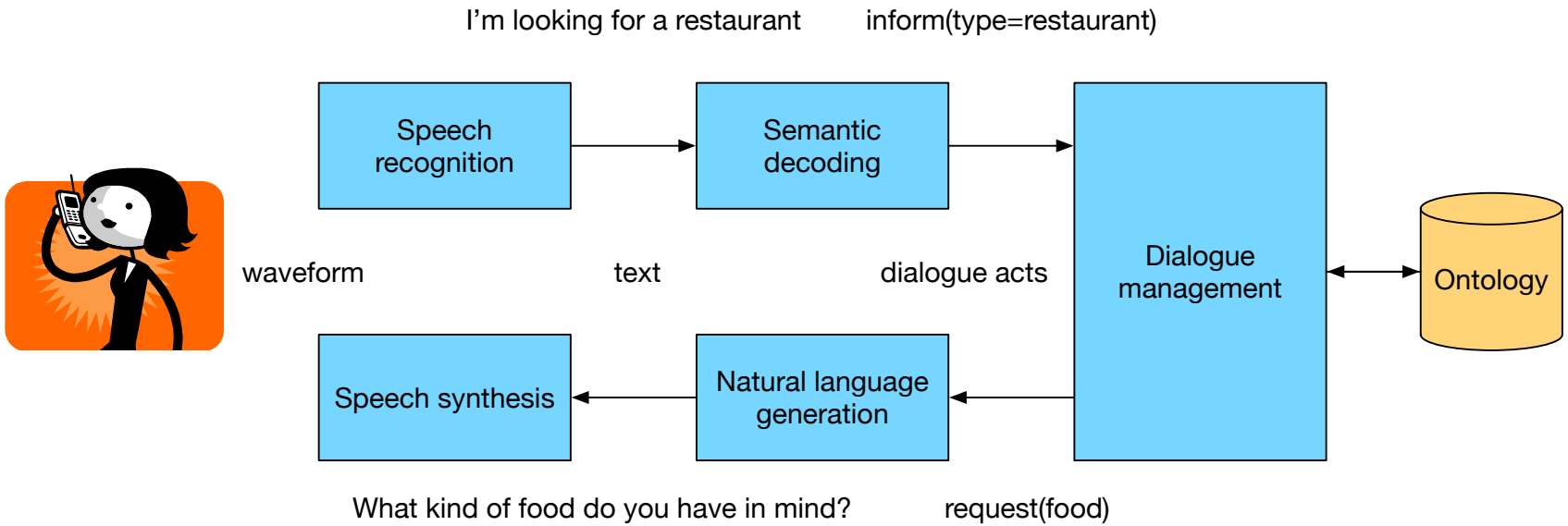
Dialogue Systems and Machine Learning Group,
Heinrich Heine University Düsseldorf

In this lecture...

Modular approach from probabilistic perspective

Speech recognition in dialogue

Architecture of a spoken dialogue system



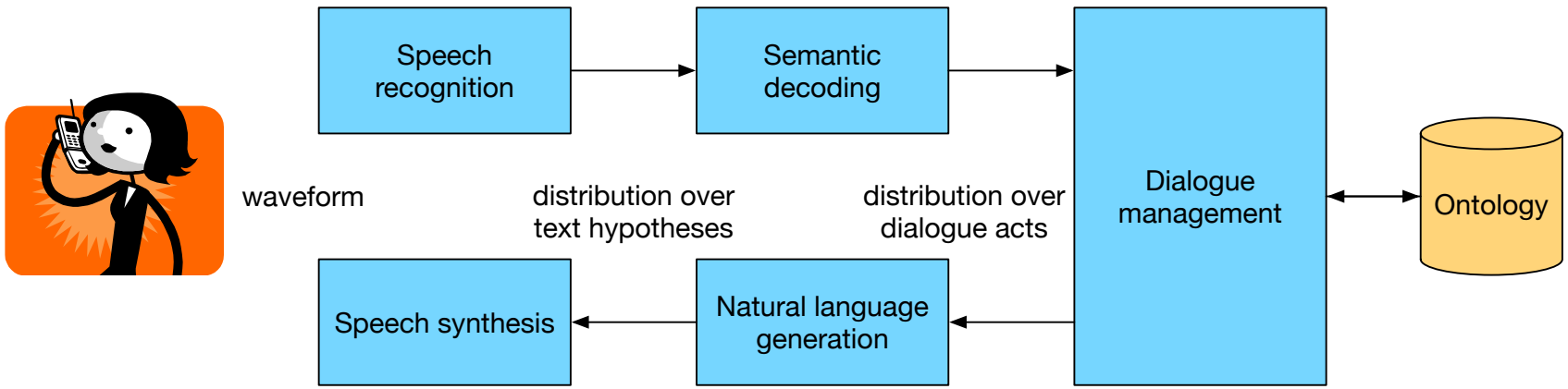
Downside of modular approach

- Information loss between the modules

Modular approach from probabilistic perspective

- Machine learning allows us not only to find the best output but a probability distribution over possible outputs.
- Similarly, machine learning allows us to consider a list of possible inputs scored by their probability.
- In this way, uncertainty is propagated through the pipeline.
- This is particularly important for spoken dialogue systems.

Architecture of a statistical spoken dialogue system



Speech recognition

- A speech recogniser converts speech into text
- It performs no understanding of what has been said
- It does however need to deal with acoustic ambiguity
  `"Recognise speech"` vs `"Wreck a nice beach"`

Example taken from [Murphy, 2012]

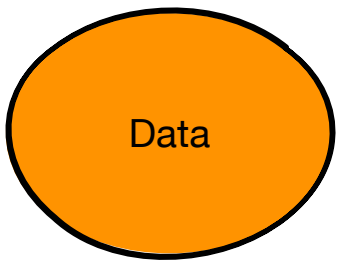
Speech recognition is hard

From a linguistic perspective

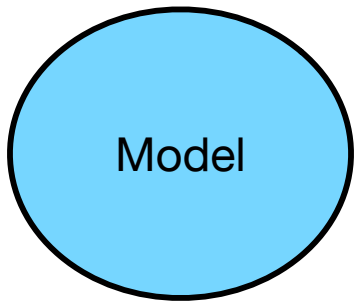
Speaker	may change
Environment	maybe noisy, there maybe other speakers, different channels (microphone)
Vocabulary	maybe diverse
Accent/Dialect	may vary for different speakers
Languages spoken	estimated to 7000, some involve code-switching
Paralinguistics	such as the emotional state influences the accuracy

Slide taken from Peter Bell's lecture

Speech recognition as a machine learning task



- Transcribed speech



- Classification
- Sequence-to-sequence models

- User utterance

Speech recognition is hard

From a machine learning perspective

- As a classification problem: high-dimensional output
- As a sequence-to-sequence problem: long sequences
- Data is often noisy, with many nuisance factors of variation in the data
- Very limited quantities of training data available (in terms of words) compared to text-based NLP
- Manual speech transcription is very expensive (10x real time)
- Hierachical and compositional nature of speech production and comprehension makes it difficult to handle with a single model

---

Slide taken from Peter Bell's lecture

Representing speech

- Represent a recorded utterance as a sequence of feature vectors.
- Feature vector is constructed by extracting features from a frame of a waveform.
- The frames are roughly 10-20ms in length, over which the waveforms are assumed to be stationary.
- Spectral properties of each segment are then calculated in order to yield a low-dimensional representation of the speech segment
- Spectral properties are usually calculated based on some model of human hearing.
- Recent neural network approaches directly use raw acoustic features.

Slide taken from Peter Bell's lecture

Labeling speech

- Labels could be phones, characters, words etc.
- Labels may be time-aligned i.e. the start and end times of an acoustic segment corresponding to a label are known

Two key challenges

During training having aligned labels

During recognition finding the most likely sequence out of all possible sequences

Hidden Markov model provides a good solution to both problems.

Slide taken from Peter Bell's lecture

Theory: Hidden Markov model (HMM)

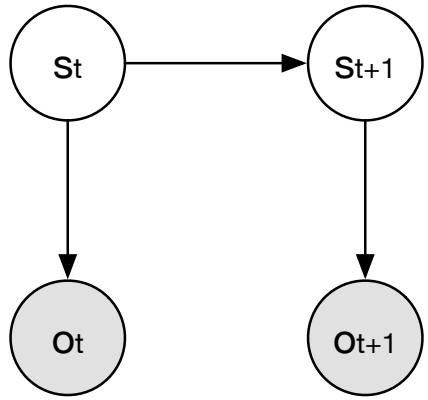
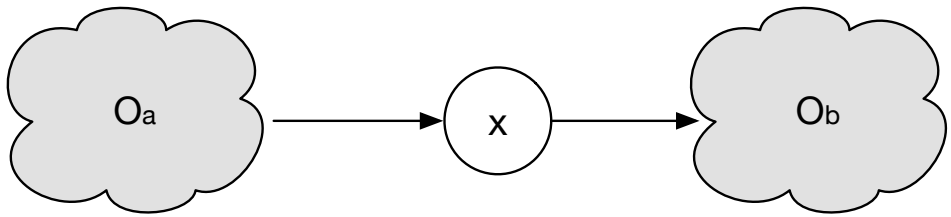


Figure 1: HMM as a dynamic Bayesian network

- In each time step we have an observation
- This observation is generated by an (unknown/hidden) state
- For each pair of state and observation we have the probability that an observation is generated by the state - *observation probability* $p(o_t|s_t, \Theta)$
- For each pair of state we have the probability that one state follows another *transition probability* $p(s_{t+1}|s_t, \Theta)$

Theory: Belief propagation

Probabilities conditional on the observations

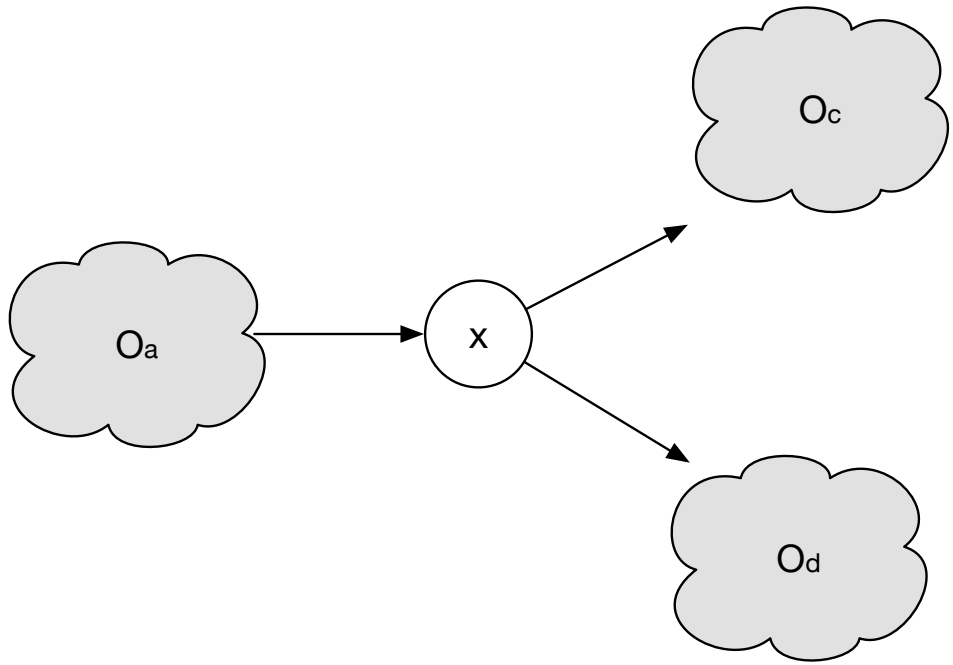


Interested in marginal probabilities $p(x|O), O = O_a \cup O_b$

$$p(x|O_b, O_a) \propto p(x, O_b|O_a) = p(O_b|x, O_a)p(x|O_a) = p(O_b|x)p(x|O_a)$$

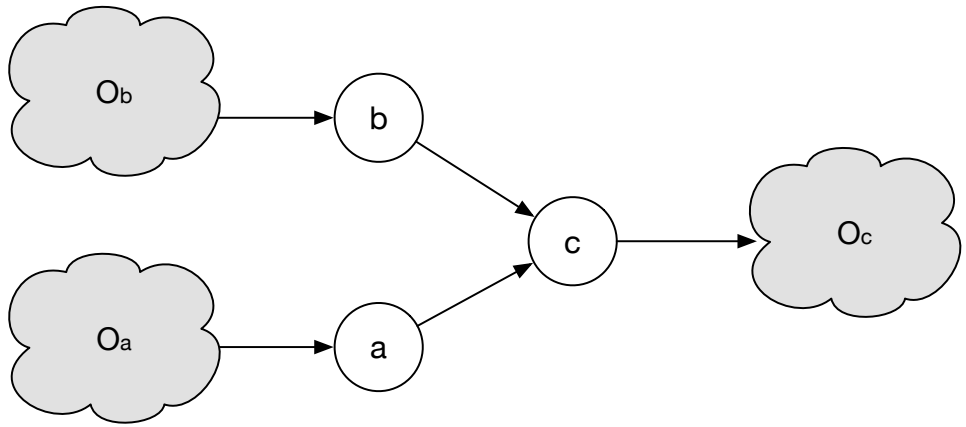
Theory: Belief propagation

Split O_b further into O_c and O_d



$$p(x|O_a, O_c, O_d) \propto p(O_c, O_d|x)p(x|O_a) = p(O_c|x)p(O_d|x)p(x|O_a)$$

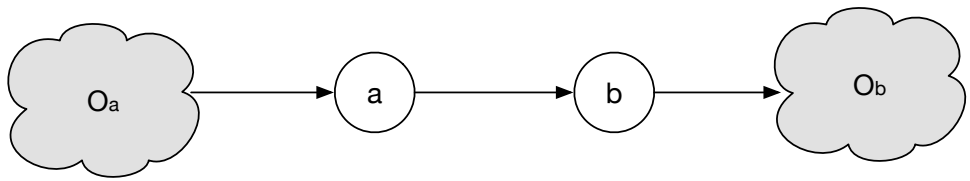
Theory: Belief propagation



$$p(c|O_a, O_b) = \sum_{a,b} p(a|O_a)p(b|O_b)p(c|a, b)$$

$$p(O_c, O_b|a) \propto \sum_{b,c} p(O_c|c)p(b|O_b)p(c|a, b)$$

Theory: Belief propagation



$$p(b|O_a) = \sum_a p(a|O_a)p(b|a)$$

$$p(O_b|a) = \sum_b p(O_b|b)p(b|a)$$

Theory: Belief propagation in HMM

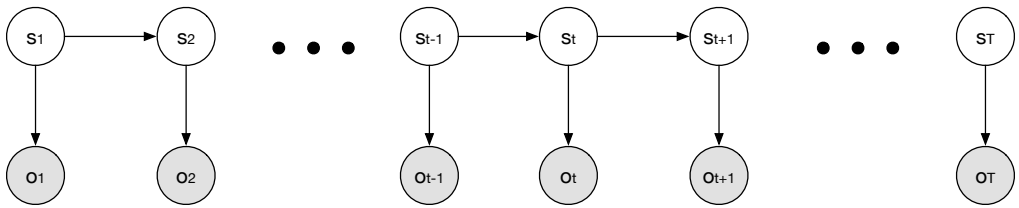


Figure 2: HMM as a dynamic Bayesian network

$$p(\mathbf{o}, \mathbf{s}|\Theta) = p(s_1|\Theta)\Pi_{t=2}^{T} p(s_t|s_{t-1}, \Theta)\Pi_{t=1}^{T} p(o_t|s_t, \Theta) \qquad (1)$$

sequence of observations $\mathbf{o} = (o_1, \cdots, o_T)$

sequence of states $\mathbf{s} = (s_1, \cdots, s_T)$

Theory: Belief propagation in HMM

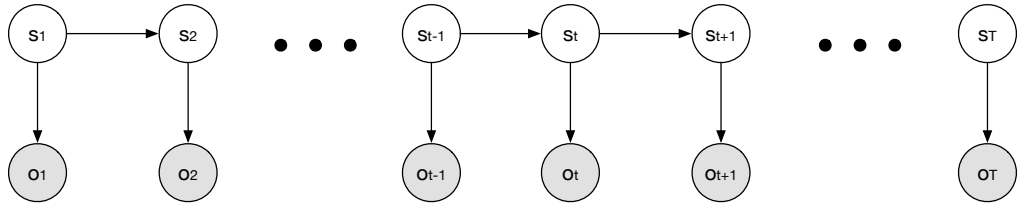


Figure 3: HMM as a dynamic Bayesian network

forward probability $\alpha_t = p(o_1, \cdots, o_t, s_t|\Theta)$

$$\alpha_t = \Sigma_{s_{t-1}} \alpha_{t-1} p(s_t|s_{t-1}, \Theta) p(o_t|s_t, \Theta)$$

backward probability $\beta_t = p(o_{t+1}, \cdots, o_T|s_t, \Theta)$

$$\beta_t = \Sigma_{s_{t+1}} p(s_{t+1}|s_t, \Theta) p(o_{t+1}|s_{t+1}, \Theta) \beta_{t+1}$$

$$p(s_t|o_1, \cdots, o_t, \Theta) \propto \alpha_t \tag{2}$$

$$p(s_t|\mathbf{o}, \Theta) \propto \alpha_t \beta_t \tag{3}$$

$$p(s_t, s_{t+1}|\mathbf{o}, \Theta) \propto \alpha_t p(o_{t+1}|s_{t+1}, \Theta) \beta_{t+1} p(s_{t+1}|s_t, \Theta) \tag{4}$$

Theory: Expectation Maximisation for HMMs

We start with some initial parameters Θ'

E step We would like to calculate the posterior distribution of the hidden states given the parameters and observations $p(\mathbf{s}|\mathbf{o}, \Theta')$. (You can think of this as soft alignment.) We could then use this posterior distribution to evaluate the expectation of the complete data log likelihood $L(\Theta) = \Sigma_{\mathbf{s}} p(\mathbf{s}|\mathbf{o}, \Theta') \log p(\mathbf{o}, \mathbf{s}|\Theta)$, which is a function of the parameters Θ. In order to evaluate L(Θ) Eq.1,3&4 are sufficient [Bishop, 2006].

M step We find new parameters $\Theta' = \arg\max_{\Theta} L(\Theta)$

We repeat these two steps until Θ' stops changing. EM algorithm is guaranteed to increase $\log p(O|\Theta)$ in every step. For derivation of parameters for HMMs used in speech recognition refer to [Gales and Young, 2007].

Theory: Viterbi algorithm

- For fixed parameters Θ, Viterbi algorithm allows us to find the most likely sequence of states for a given sequence of observations.
- Note that this is different to finding the set of states that are individually most probable as is done in E step of the EM algorithms.

Theory: Viterbi algorithm

- Consider a time step t and a particular state s^k.
- We want to find the sequence that has the highest probability and takes state s^k at time t.
  $\phi_t(s^k) = max_{s_1,\cdots,s_{t-1}} p(o_1, \cdots, o_t, s_1, \cdots, s_{t-1}, s_t = s^k)$
- Because there are K possible states at time step t, we need to keep track of K such sequences.
- At time step $t+1$, there will be K^2 possible sequences to consider, comprising K possible sequences leading out of each of the K current states, but we only need to retain K of these corresponding to the best sequence for each state at time $t+1$.
  $\phi_{t+1}(s^l) = max_k \phi_t(s^k) p(s_{t+1} = s^l | s_t = s^k) p(s_{t+1} = s^l | o_{t+1})$
- When we reach the final time step T we will discover which state corresponds to the overall most probable sequence. Because there is a unique path coming into that state we can trace the sequence back to the beginning.

Hierarchical modelling of speech

- Utterance consists of words
- Words consist of subwords (ie. phones, triphones) - this is provided by a pronunciation dictionary
- Three senones: beginning, middle and end comprise a subword unit. If we take into account contextual subword units - there may be as many as $20,000$ senones.
- The sequence of senones for an utterance is modelled as a sequence of states in HMM.
- Each state of an HMM generates a feature vector of a frame.

Hierarchical modelling of speech

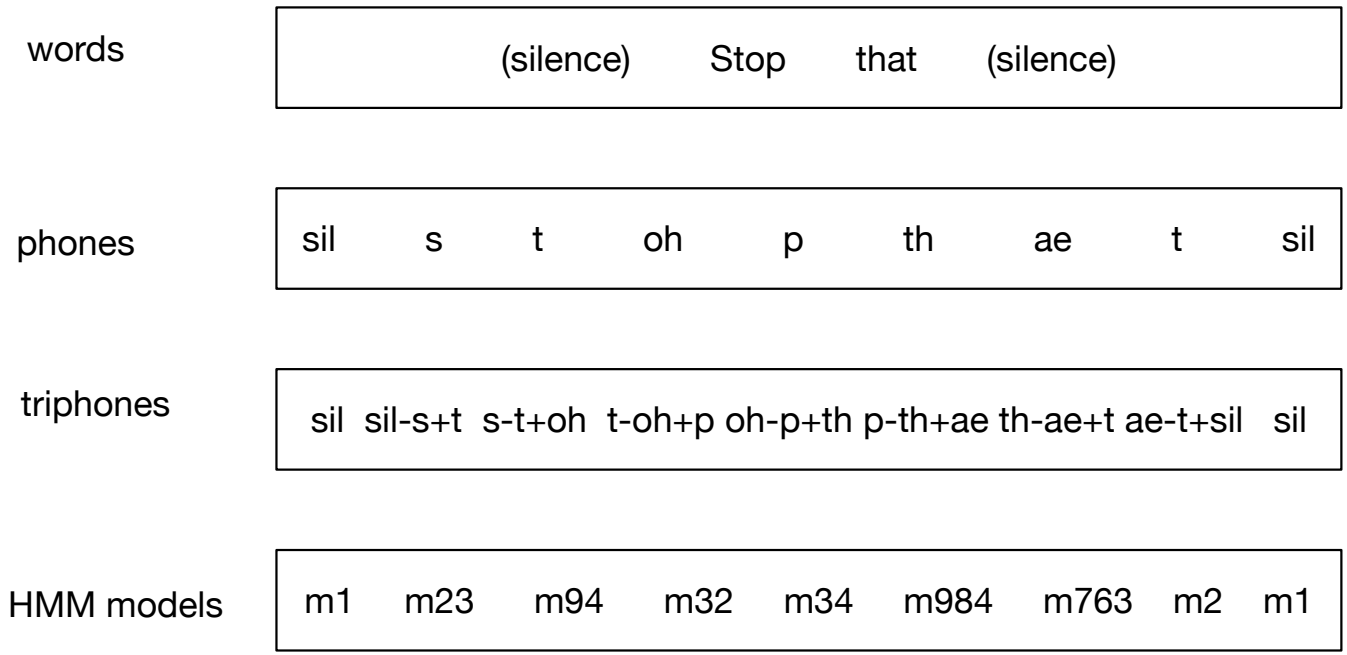


Figure 4: Context dependent phone modeling [Gales and Young, 2007]

Hidden Markov model in speech recognition

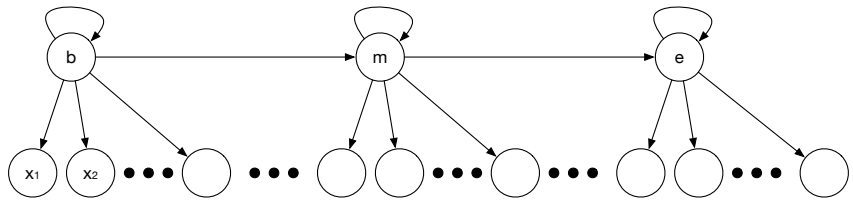


Figure 5: HMM as a probabilistic finite state machine

- The transition probabilities are partly known: the next state is either the same senon, or middle if previous was beginning , or end if previous was middle.
- The observation probability: how a state generates the feature vector is unknown
- This can be modelled as a Gaussian Mixture Model.

Hidden Markov model in speech recognition

- Traditionally each phone was modelled with three states.
- This enforces a minimal duration of three frames per phone.
- The phone HMMs can be concatenated to form an HMM for the whole word, using the **pronunciation dictionary**.

Fundamental Equation of Statistical Speech Recognition

- If $\mathbf{x}$ is the sequence of acoustic feature vectors (observations) and $\mathbf{w}$ denotes a word sequence, the most likely word sequence $\mathbf{w}^*$ is given by

$$\mathbf{w} = \arg\max_{\mathbf{w}} P(\mathbf{w}|\mathbf{x}) \tag{5}$$

- Applying Bayes Theorem:

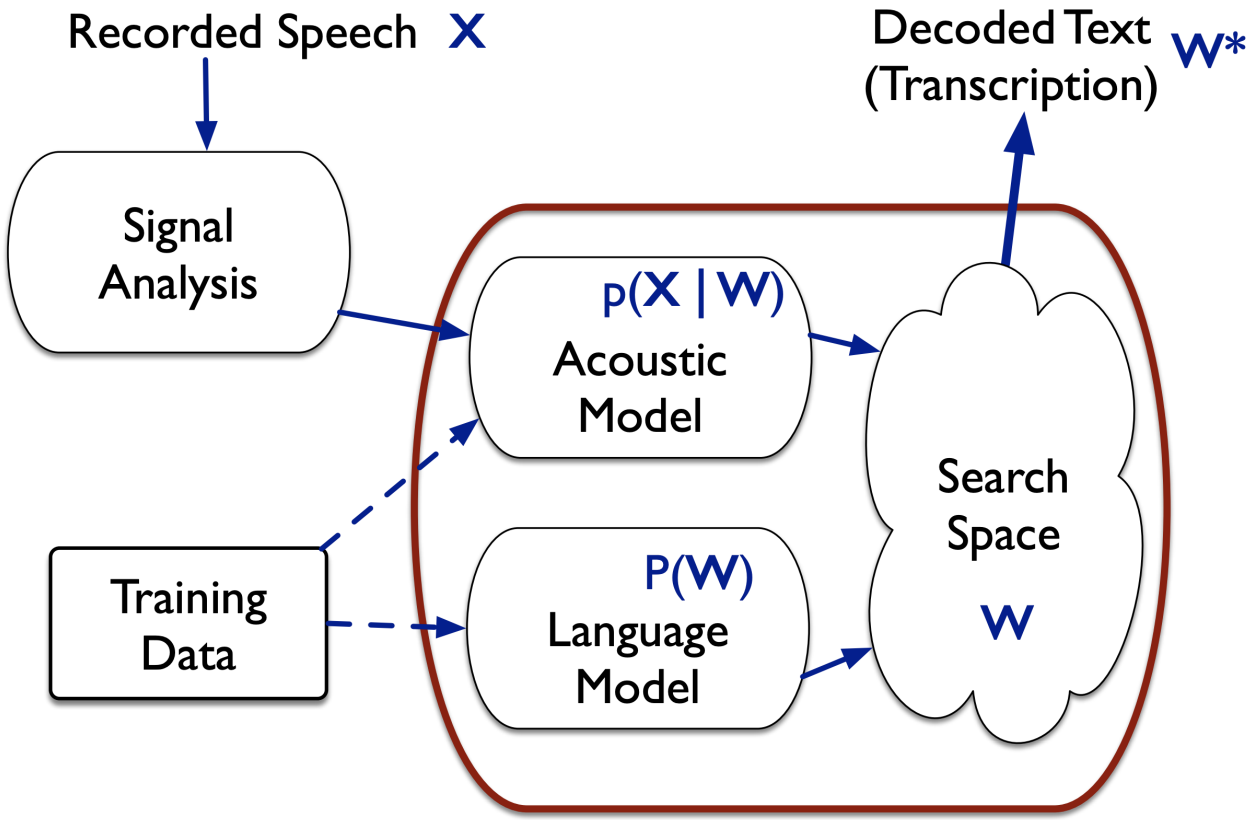
$$P(\mathbf{w}|\mathbf{x}) = \frac{P(\mathbf{x}|\mathbf{w})P(\mathbf{w})}{P(\mathbf{x})} \tag{6}$$

$$\propto P(\mathbf{x}|\mathbf{w})P(\mathbf{w}) \tag{7}$$

- $P(\mathbf{x}|\mathbf{w})$ is the **acoustic model**
- $P(\mathbf{w})$ is the **language model**

Slide taken from Peter Bell's lecture

ASR overview



Slide taken from Peter Bell's lecture

Acoustic modelling

- The role of the acoustic model is to model the probability that a sequence of feature vectors is generated by a given word sequence $p(\mathbf{x}|\mathbf{w})$
- This can be done using an HMM where the transition probabilities are parametrised as a Gaussian Mixture Model and estimated using expectation maximisation.

Hybrid HMM-DNN systems

- By modelling the observation probabilities as a GMM we place very strong assumptions
- A deep neural network (DNN) as a function estimator places fewer assumptions on the unknown function that it is approximating
- Once we have a GMM/HMM we know which state corresponds to which observation and we can model the observations probability as a DNN
- This is *bootstrapping*: the process of self improvement
- In this case we still use the HMM to find the most likely sequence

End-to-end acoustic models

- If we have sequence of phones or characters and the corresponding sequence of frames
- We can directly model acoustics in an end-to-end fashion, as a sequence to sequence learning task
- What is particularly useful is that one directly consider raw audio features
- One good way to achieve so is via a transformer [Pham et al., 2019].

How do we measure how good is speech recognition?

- We are interested in the quantity called **word error rate**
- It is the Levenshtein distance between the hypothesised sequence of words and the target sequence of words

Theory: Levenshtein distance

$\mathbf{h}$ hypothesis sequence

$\mathbf{t}$ target sequence

$$m = |\mathbf{h}|, n = |\mathbf{t}|$$

$$D_{0,0} = 0, D_{i,0} = i, D_{0,j} = j, 0 \leq i \leq n, 0 \leq j \leq m$$

$$D_{i,j} = \min \begin{cases} D_{i-1,j-1} & h_i = t_j \\ D_{i-1,j-1} + 1 & \text{substitution} \\ D_{i,j-1} + 1 & \text{insertion} \\ D_{i-1,j} + 1 & \text{deletion} \end{cases} \qquad 1 \leq i \leq n, 1 \leq j \leq m$$

Word error rate

- Use Levenshtein distance to calculate the distance between the ASR output h and the reference transcription t, by calculating substitutions, insertions and deletions.
- If there are n words in the reference transcript, the word error rate and the accuracy is given as

$$\text{WER} = 100 * \frac{D(h,t)}{n} \tag{8}$$

$$\text{Accuracy} = 100 - \text{WER} \tag{9}$$

Acoustic modelling for dialogue systems

- Spoken dialogue systems are meant to be used everywhere: busy street, noisy car
- Advantage: the conversation spans over several turns so it is possible to perform adaptation in the first turn to improve future interactions
- Advantage: the same speaker through-out the dialogue

Language modelling

- The role of a language model is to model the probability of a word sequence $p(\mathbf{w})$
- Some sequences of words are more likely than others
- $p(\mathbf{w}) = p(w_1)p(w_2|w_1)\cdots p(w_t|w_{t-1}, ...)$, where w_i is the ith word in the sequence
- $p(w_t|w_{t-1}, ...) \approx p(w_t|w_{t-1}, ..., w_{t-n})$ **n-gram** language model

Perplexity

- Entropy

$$H(\mathbf{w}) = -\frac{1}{N} \log p(w_1, \cdots, w_N)$$

- Perplexity

$$PP(\mathbf{w}) = 2^{H(\mathbf{w})}$$

Language modelling for dialogue systems

- The vocabulary in limited domain dialogue systems is small so the language model can be trained with in-domain data
- A general purpose language model can be combined with in-domain language model to provide better recognition results and also deal with out-of-domain requests.

Speech recognition for dialogue systems

Provide alternative recognition result

- N-best list (extension of the Viterbi algorithm)
- Confusion network
- Lattice

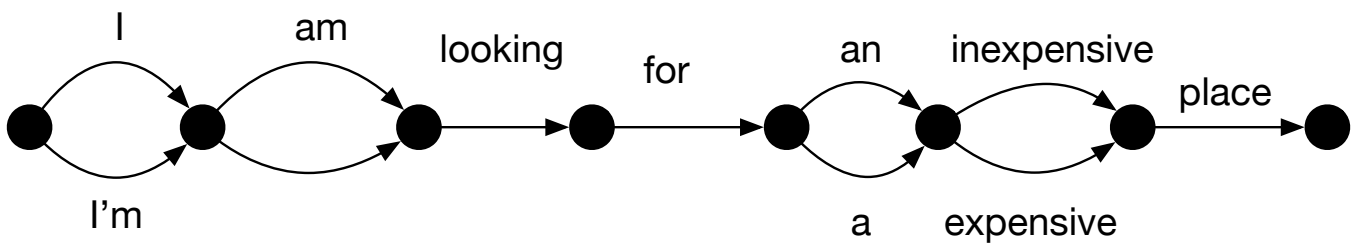


Figure 6: Confusion network

Speech recognition for dialogue systems

Recognise when the user has started speaking

- Key-word spotter running on a smartphone - always listening [Chen et al., 2015]
- Requirements: low memory footprint, low computational cost and high precision

Recognise when the user has stopped speaking

- This is studied in the broad context of voice activity detection

Theory: Precision and recall

Measure	Data	Model
TP true positive	+	+
TN true negative	-	-
FP false positive	-	+
FN false negative	+	-

$$P = \frac{TP}{TP + FP}$$
$$R = \frac{TP}{TP + FN}$$
$$F = 2\frac{P \cdot R}{P + R}$$

How much data do we need to train a speech recogniser?

ACHIEVING HUMAN PARITY IN CONVERSATIONAL SPEECH RECOGNITION

W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu and G. Zweig

Microsoft Research
Technical Report MSR-TR-2016-71
Revised February 2017

ABSTRACT

Conversational speech recognition has served as a flagship speech recognition task since the release of the Switchboard corpus in the 1990s. In this paper, we measure the human error rate on the widely used NIST 2000 test set, and find that our latest automated system has reached human parity. The

self-corrections, hesitations and other disfluencies that are pervasive. The Switchboard [10] and later Fisher [11] data collections of the 1990s and early 2000s provide what is to date the largest and best studied of the conversational corpora. The history of work in this area includes key contributions by institutions such as IBM [12], BBN [13], SRI [14], AT&T [15], LIMSI [16], Cambridge University [17], Microsoft [18]

- Thousands hours of transcribed speech to train the acoustic model
- Close to billion words to train the language model
- Deep neural network models have achieved impressive results, though human parity remains arguable.

Summary

- Speech recognition converts speech into text.
- This is a hard AI problem.
- Hierarchical modelling of speech decomposes utterance into words, words in subword units.
- Each subword unit can be modelled as a parameterised HMM with acoustic feature vectors as observations.
- For an HMM-GMM there is a closed form solution for parameters, but more recently hybrid HMM-DNN models achieved state of the art results.
- From a point of view of dialogue what is important to consider are alternative recognition so that the uncertainty is propagated through the pipeline.

Next lecture

- Semantic decoding

References I


Bishop, C. M. (2006).
Pattern Recognition and Machine Learning (Information Science and Statistics).
Springer-Verlag, Berlin, Heidelberg.


Chen, G., Parada, C., and Sainath, T. (2015).
Query-by-example keyword spotting using long short-term memory networks.
In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 5236–5240.


Gales, M. and Young, S. (2007).
The application of hidden markov models in speech recognition.
Found. Trends Signal Process., 1(3):195304.

References II


Murphy, K. P. (2012).
Machine Learning: A Probabilistic Perspective.
The MIT Press.


Pham, Q., Nguyen, T.-S., Niehues, J., Muller, M., and Waibel, A. (2019).
Very deep self-attention networks for end-to-end speech recognition.
In *Proceedings of Interspeech*.

Credits

We thank Peter Bell for sharing his lecture notes Automatic Speech Recognition