- - - Speech Processing - - -

- Basic questions involving HMMs are:
  - Given an observation sequence X and an HMM λ, how could we efficiently compute the probability of observing X considering the model, i.e., $P(O|\lambda)$?
    - Solution: backward-forward algorithm.
  - Given an observation sequence X and an HMM λ, how could we find the best sequence of states and transitions to explain the observation?
    - Solution: Viterbi's Algorithm.
  - Given an observation sequence X and an HMM λ, how could we adjust the parameters to maximize the probability of observing a certain sequence, i.e., how to train the model for its maximum efficiency?
    - Solution: Baum-Welch's (expectation - maximization (EM)) algorithm.

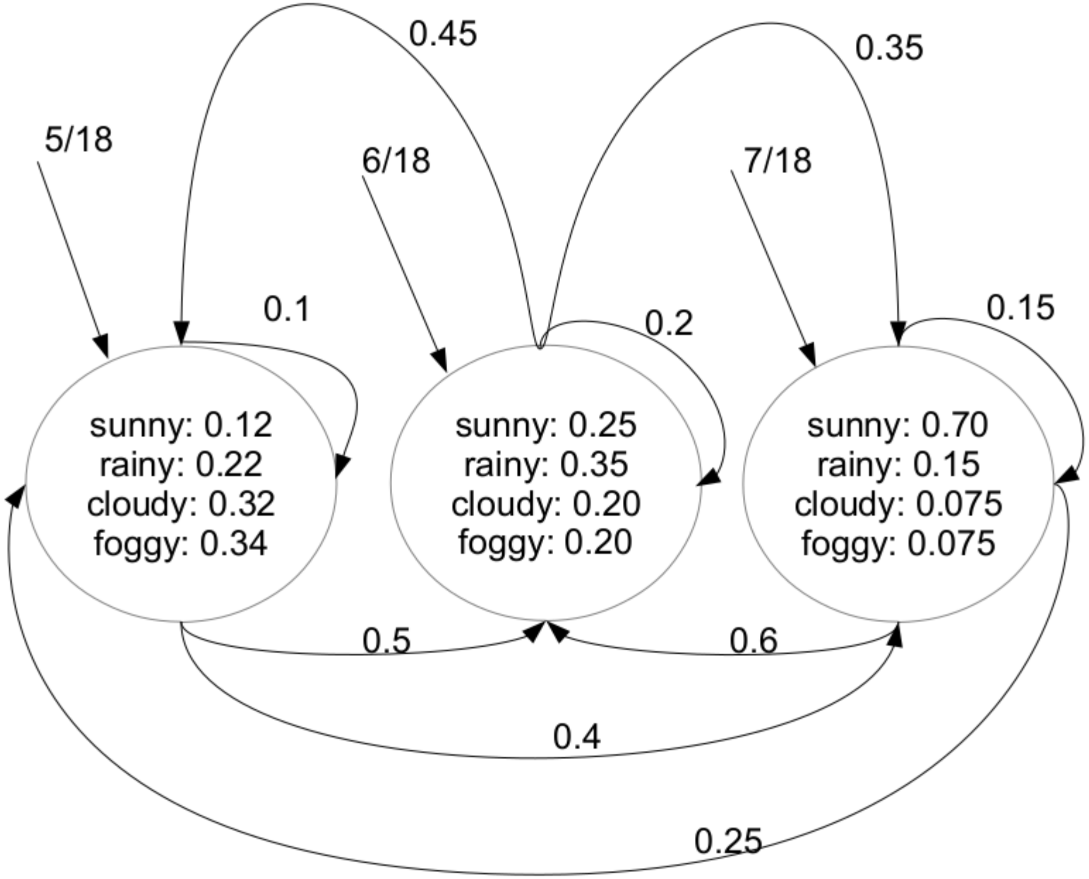
$$\pi_i \leftarrow \text{expected number of times in state } i \text{ at the initial time}$$

$$a_{i,j} \leftarrow \frac{\text{expected transitions from state } i \text{ to } j}{\text{expected transitions from state } i}$$

$$b_{j,k} \leftarrow \frac{\text{expected number of times in state } j \text{ and observing the symbol } k}{\text{expected number of times in state } j}$$

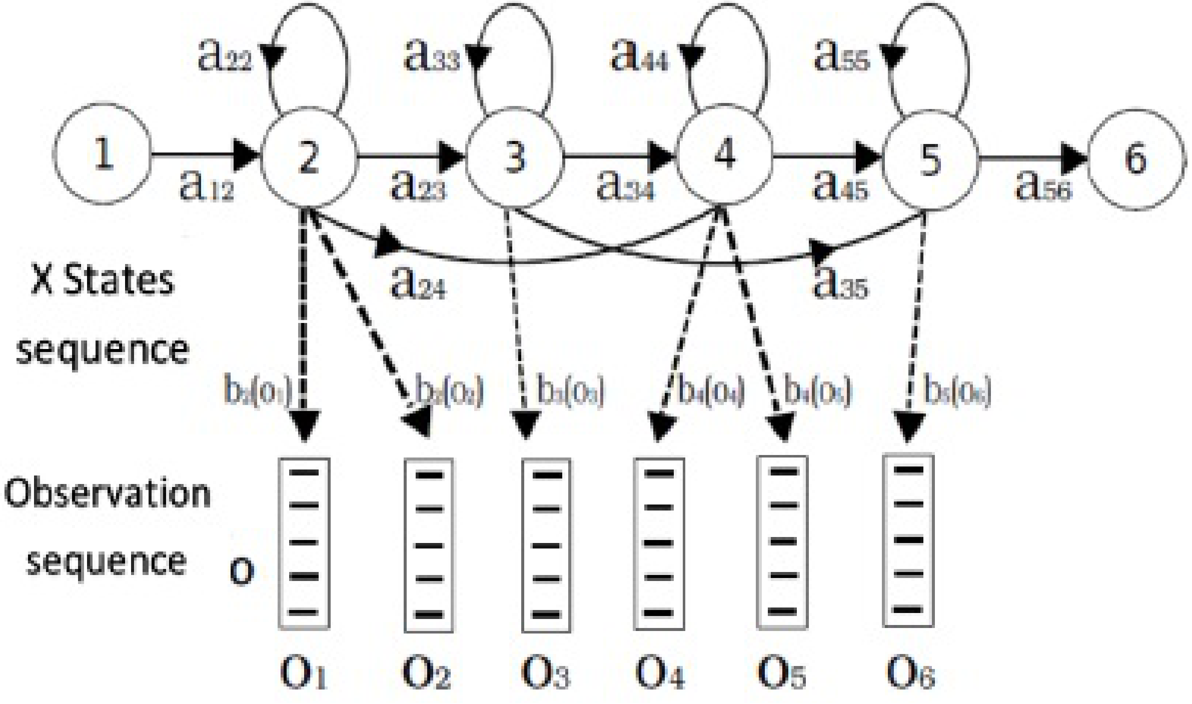
- - - Speech Processing - - -

- Example: Answer the two first questions above for the example weather-related HMM shown below, considering the observation sequence $O = \{\text{rainy}, \text{foggy}\}$.



- - - Speech Processing - - -

- Basic idea for HMM-based speech recognition: a left-to-right HMM is used for each phoneme, where state transitions represent time-shifting and each state represents a phonetic unit.

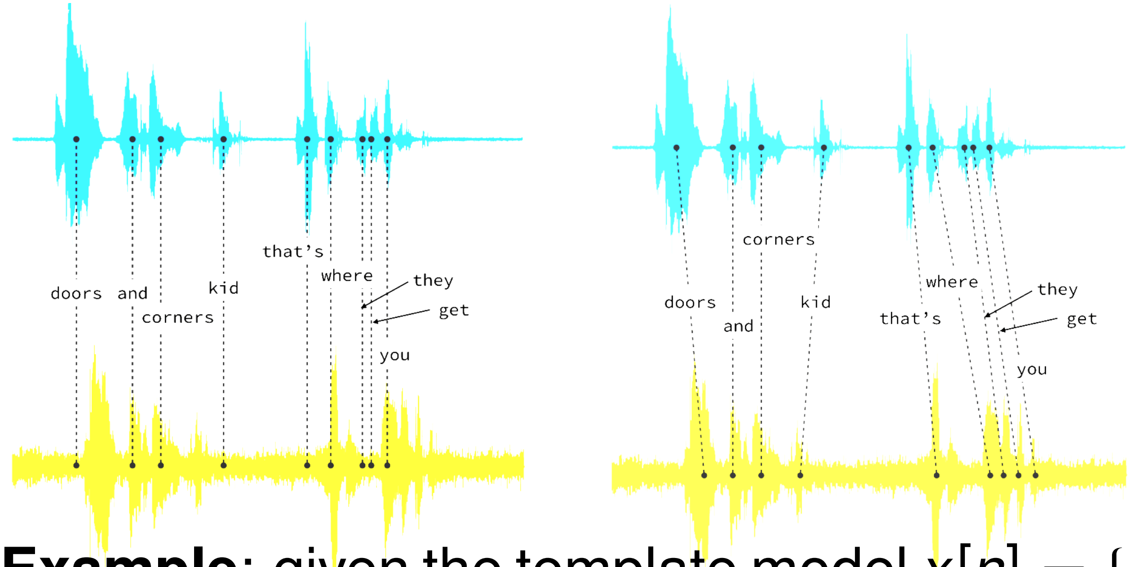


CCO50 - Digital Speech Processing

- - - Speech Processing - - -

- **Dynamic Time Warping (DTW) Classifier**: it is a strategy used to directly quantify the similarity between two signals of different lengths. It can be implemented based on a simple matrices-based strategy, as explained during the class.

Figure: [left]: ordinary comparison; [right]: DTW-based comparison



- **Numerical Example**: given the template model $x[n] = \{1, 1, 2, 3, 2, 0\}$, choose its best match based on DTW: either $y_1[n] = \{0, 1, 1, 2, 3, 2, 1\}$ or $y_2[n] = \{2, 1, 4, 2, 0, 2, 1\}$.

- - - Speech Processing - - -

- **Pattern Matching** vs **Knowledge-based Classifiers**: strategies based purely on template models and on statistical or numerical knowledge acquisition, respectively.
  - distance metrics, DTW, K-nearest neighbors (KNNs);
  - neural networks, support vector machines (SVMs), HMMs, and so on.
- **Shallow vs Deep Knowledge-based Classifiers**:
  - running requirements,
  - complexity,
  - accuracy.
- **A Final Overview on Speech Analysis Applications**: speech recognition; speaker identification/recognition (voice recognition); speaker verification and spoof detection; speech pathology detection; speech emotion recognition; idiom recognition. Popular datasets for speech analysis are TIMIT, YOHO, ASVSpoof, SVD, and so on.

- - - Speech Processing - - -

- **Additional Speech Filtering Approaches**: Wiener and Kalman Filters are particular strategies for noise removal from speech signals. The former is an FIR filter based on the least-squares method and requires samples of both noisy and noiseless frames to be defined, whereas the latter works as an IIR filter which just requires the noisy input because it assumes a Gaussian pattern for the noise. Particularly for Kalman filtering, the iterative required procedure is:
    - Beginning: for $i = 0$, we just set $y_0 = 0$ and $p_0 = 1$.
    - For $i = 1, 2, ..., M - 1$, where M is the length of $x[n]$ and $y[n]$, i.e., the noisy input and the filtered output signals, respectively, we repeat:

$$k_i = \frac{p_{i-1}}{p_{i-1} + \sigma_x}, \quad \text{where } \sigma_x \text{ is the standard deviation of } x[n]$$

$$y_i = k_i \cdot x_i + (1 - k_i) \cdot y_{i-1}$$

$$p_i = (1 - k_i) \cdot p_{i-1}$$

    - End.

Notably, as i increases, y_i approaches the desired filtered signal.

- - - Speech Processing - - -

Example: Assuming that $x[n] = \{1, 2, 4, 10, 20\}$ is the input noisy signal and $y[n] = \{\frac{3}{2}, \frac{3}{2}, 3, \frac{9}{2}, 15\}$ is its corresponding noiseless version, define a 2nd order Wiener filter to remove the noise from $x[n]$.

Example: Assuming that $x[n] = \{0.39, 0.50, 0.48, 0.29, 0.25, 0.32, 0.34, 0.48, 0.41, 0.45\}$ is the input signal contaminated with Gaussian noise, define the Kalman filter to remove the noise from $x[n]$.

Today's Short-Test (ST12): Comment on the applications of Wiener filtering involving speech data.

- **Speech Synthesis**: specific strategies adopted for producing speech signals artificially. Decades ago, they were based on the Lilgencrants & Fant (LF) model, used to produce the glotal source signal, followed by an IIR filtering approach that simulates the vocal and nasal tract. Currently, deep learning-based models have been adopted successfully: raw text is mapped to numbers, in a certain range representing phonemes, and then used as input to an artificial neural network which synthesizes speech directly.

- - - Speech Processing - - -

- **Voice Conversion**: also known as voice morphing, modifies a certain speaker's voice to sound as if it were spoken by a target speaker. Analysis, for pitch and formant tracking, followed by re-synthesis procedures were quite common decades ago. Nowadays, deep learning-based strategies have been intensively used.
- **Blind Source Separation**: strategies to separate individual voices from a set of voices occurring in parallel. Currently, they have also been based on deep-learning models.

- -

The course is over! Many thanks for attending! Hopefully, the material, examples, and discussions will help you in moving your own projects forward, no matter if it focuses on the field of speech processing or on any related subject!!!!

- -

- - - Final-Term Exam - - -

- **(1)**: Design an FIR filter ($q[n]$), subtype I, of order $M = 8$ to cut-off frequencies within the range 2000 Hz $\sim$ 4000 Hz, allowing for all the others to pass through. Assume that the input signal to be filtered ($x[n]$) was sampled at 22050 samples per second. Normalize the filters' coefficients in such a way that the filter presents a gain of 0dB in the pass-band. Lastly, write down the difference equation to filter an input signal $x[n]$ by using $q[n]$.
- **(2)**: Design an FIR filter with the following specifications:
    - $0.98 \leqslant |H(e^{j\omega})| \leqslant 1.02$, in the range $0 \leqslant \omega \leqslant 0.1\pi$
    - $|H(e^{j\omega})| \leqslant 0.06$, in the range $0.4\pi \leqslant \omega \leqslant \pi$

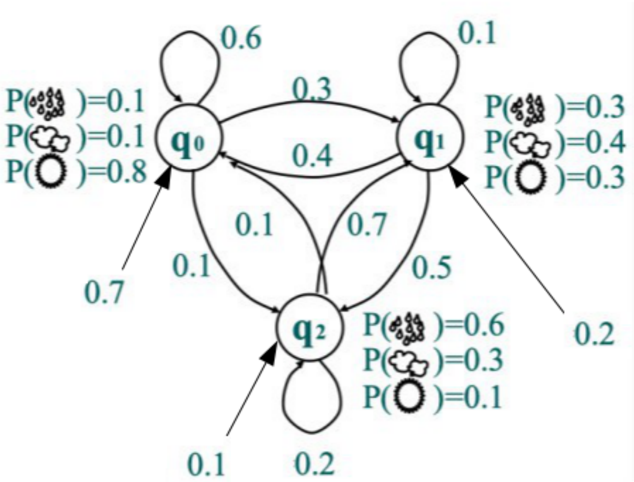
  Then, normalize the windowed filter, write down the difference equation to implement the filter you have just designed in a computer-based application, and depict the corresponding block diagram.
- **(3)**: Explain, in a short paragraph, the concept of window. What is it used for? Why is it important?

- - - Final-Term Exam - - -

- **(4)**: Find the transfer function $H[z]$ whose poles are at $z = -\frac{1}{4}$ and $z = \frac{1}{3}$, in addition to one zero at $z = \frac{1}{5}$. Write down the corresponding difference equation. Is the transfer function stable and causal?
- **(5)**: Consider the hypothetical speech signal segment $s[n] = \{-1, 2, -3, 3, 2, 1, -1, -1, -4, 5, 5, 4\}$, sampled at 16000 samples per second. Assume that a sliding rectangular window $w[n]$ traverses it in order to extract features for inclusion in the feature vector $f[n]$, covering 0.125ms at each placement, with 50% overlap between consecutive windows. What is the length of $f[n]$? What are the values in $f[n]$, considering the ordinary entropy, calculated with the log basis $\beta = 10$, as being the feature used?
- **(6)**: Explain, in a short paragraph, the concept of Bark scale. Why is it important?
- **(7)**: Estimate the 4$^{\text{rd}}$ order LPC coefficients $\{a_1, a_2, a_3, a_4\}$ for the signal $y[n] = \{2, 5, 2, 3, 5, 8, 4, 8, 10, 6\}$.

- - - Final-Term Exam - - -

- **(8)**: Explain the differences between, the advantages, and disadvantages of handcrafted features in comparison with those learned.
- **(9)**: Answer the first two HMM-related questions on page 99 for the weather-related model below, considering the observation sequence $O = \{rainy, sunny\}$.



- **(10)**: Given the template model for classes C_A and C_B, i.e., $\{0.2, 1, 0.7\}$ and $\{0.3, 0.9, 0.8\}$, respectively, find the best matching class for the testing signal $t[n] = \{0.2, 0.9, 0.9\}$ by using the ordinary Euclidian distance metric. What would you do to find that match in case the testing signal and the template models have different dimensions?