Multimodal Interaction

Lesson 6
Speech Interaction

Maria De Marsico
demarsico@di.uniroma1.it

Foundations

- Vocal (speech) interaction requires speech recognition
  - «Semaphoric» triggerwords
  - More articulated composition of words

  - Speaker dependent
  - Speaker independent
- Problems are related to variations of the context, speakers, and environment

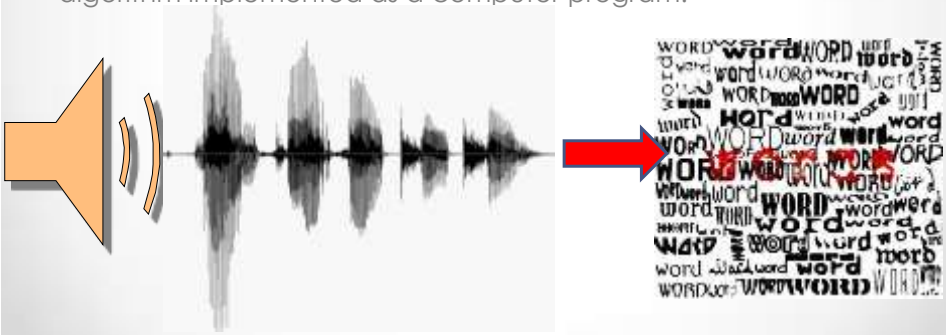
Credits

- This lesson is partly derived by the folowing review:
- M.A.Anusuya, S.K.Katti. **Speech Recognition by Machine: A Review.** *(IJCSIS) International Journal of Computer Science and Information Security,*
- *Vol. 6, No. 3, 2009, pp. 181-205*

Definition

Speech recognition:

Speech Recognition (or Automatic Speech Recognition (ASR), or computer speech recognition) is the process of converting a speech signal to a sequence of words, by means of an algorithm implemented as a computer program.



Terminology: Utterance

- When the user says something, this is known as an **utterance.** An utterance is any stream of speech between two periods of **silence**. Utterances are sent to the speech engine to be processed.
- Silence, in speech recognition, is almost as important as what is spoken, because silence delineates the start and end of an utterance.
- An utterance can be a single word, or it can contain multiple words (a phrase or a sentence). For example,

  "checking"
  "checking account"
  "I'd like to know the balance of my checking account please"

  are all examples of possible utterances - things that a user might say to a banking application.
- Whether these words and phrases are valid at a particular point in a dialog is determined by which **grammars** are active .
- There are small snippets of silence between the words spoken within a phrase. If the users pauses too long between the words of a phrase, the end of an utterance can be detected too soon, and only a partial phrase will be processed by the engine.

Terminology: Pronunciations

- The speech recognition engine uses all sorts of data, statistical models, and algorithms to convert spoken input into text.
- One piece of information that the speech recognition engine uses to process a word is its **pronunciation**, which represents what the speech engine thinks a word should sound like.
- Words **can have multiple pronunciations** associated with them. For example, the word "the" has at least two pronunciations in the U.S. English language: "thee" and "thuh." One may want to provide multiple pronunciations for certain words and phrases to allow for variations in the ways users may speak them.

Terminology: Grammars

- A developer must specify the words and phrases that users can say to an application. These words and phrases are defined to the speech recognition engine and are used in the recognition process.
- One can specify the valid words and phrases in a number of different ways, for instance by specifying a **grammar (as for example in VoiceXML)**. A grammar uses a particular syntax, or set of rules, to define the words and phrases that can be recognized by the engine.
- A grammar can be as simple as a list of words, or it can be flexible enough to allow such variability in what can be said that it approaches natural language capability.
- Grammars define the domain, or context, within which the recognition engine works. The engine compares the current utterance against the words and phrases in the active grammars. If the user says something that is not in the grammar, the speech engine will not be able to decipher it correctly.

Terminology: Accuracy

- Accuracy is typically a quantitative measurement of the performance of a speech recognition system and can be calculated in several ways.
- Arguably the most important measurement of accuracy is whether the desired end result occurred.
- For example, if the user said "yes," the engine returned "yes," and the "YES" action was executed, it is clear that the desired end result was achieved.
- What happens if the engine returns text that does not exactly match the utterance? For example, what if the user said "nope," the engine returned "no," yet the "NO" action was executed? Should that be considered a successful dialog? The answer to that question is yes because the desired end result was achieved.

Terminology: Accuracy

- Another measurement of recognition accuracy is whether the engine recognized the utterance exactly as spoken.
- This measure of recognition accuracy is expressed as a percentage and represents the number of utterances recognized correctly out of the total number of utterances spoken.
- Based on the accuracy measurement, one may want to analyze a grammar to determine if there is anything to improve accuracy. One may also want to check a grammar to see if it allows words that are acoustically similar (for example, "repeat/delete," "Austin/Boston," and "Addison/Madison"), and determine if there is any way to make the allowable words more distinctive to the engine.
- Recognition accuracy is an important measure for all speech recognition applications. It is tied to grammar design and to the acoustic environment of the user.

Conditions

- The conditions of evaluation - and hence the accuracy of any system - can vary along a number of dimensions …

Conditions: Vocabulary size and confusability.

- As a general rule, it is easy to discriminate among a small set of words, but error rates naturally increase as the vocabulary size grows.
- For example, the 10 digits "zero" to "nine" can be recognized essentially perfectly , but vocabulary sizes of 200, 5000, or 100000 may have error rates of 3%, 7%, or 45%.
- Even a small vocabulary can be hard to recognize if it contains confusable words. For example, the 26 letters of the English alphabet (treated as 26 "words") are very difficult to discriminate because they contain so many confusable words; an 8% error rate is considered good for this vocabulary

Conditions: Speaker dependence vs. independence.

- Some SR systems use "training" where an individual speaker reads sections of text into the SR system. These systems analyze the person's specific voice and use it to fine tune the recognition of that person's speech, resulting in more accurate transcription.
- Systems that **use** training are called "**Speaker Dependent**" systems.
- Systems that **do not use** training are called "**Speaker Independent**" systems

Conditions: Speaker dependence vs. independence.

- By definition, a speaker dependent system is intended for use by a single speaker, but a speaker independent system is intended for use by any speaker.
- Speaker independence is difficult to achieve because a system's parameters become tuned to the speaker(s) that it was trained on, and these parameters tend to be highly speaker-specific.

Conditions: Isolated, discontinuous, or continuous speech.

- Isolated speech means single words; discontinuous speech means full sentences in which words are artificially separated by silence; and continuous speech means naturally spoken sentences.
- Isolated and discontinuous speech recognition is relatively easy because word boundaries are detectable and the words tend to be cleanly pronounced.

Conditions: Task and language constraints.

- Even with a fixed vocabulary, performance will vary with the nature of constraints on the word sequences that are allowed during recognition.
- Some constraints may be task-dependent (for example, an airline querying application may dismiss the hypothesis "The apple is red"); other constraints may be semantic (rejecting "The apple is angry"), or syntactic (rejecting "Red is apple the").
- Constraints are often represented by a grammar, which ideally filters out unreasonable sentences so that the speech recognizer evaluates only plausible sentences.
- Grammars are usually rated by their perplexity, a number that indicates the grammar's average branching factor (i.e., the number of words that can follow any given word). The difficulty of a task is more reliably measured by its perplexity than by its vocabulary size.

Conditions: Read vs. spontaneous speech.

- Systems can be evaluated on speech that is either read from prepared scripts, or speech that is uttered spontaneously.
- Spontaneous speech is vastly more difficult, because it tends to be peppered with disfluencies like "uh" and "um", false starts, incomplete sentences, stuttering, coughing, and laughter; and moreover, the vocabulary is essentially unlimited, so the system must be able to deal intelligently with unknown words (e.g., detecting and flagging their presence, and adding them to the vocabulary, which may require some interaction with the user).

Conditions: Adverse conditions.

- A system's performance can also be degraded by a range of adverse conditions. These include:
  - environmental noise (e.g., noise in a car or a factory);
  - acoustical distortions (e.g, echoes, room acoustics);
  - different microphones (e.g., close-speaking, omnidirectional, or telephone);
  - limited frequency bandwidth (in telephone transmission);
  - altered speaking manner (shouting, whining, speaking quickly, etc.).

Multilevel

- Speech recognition is a multileveled pattern recognition task
- Acoustical signals are examined and structured into a hierarchy of
  - subword units (e.g., phonemes),
  - words,
  - sentences.
- Each level may provide additional temporal constraints, e.g., known word pronunciations or legal word sequences, which can compensate for errors or uncertainties at lower levels.
- Hierarchy of constraints can best be exploited by combining decisions probabilistically at all lower levels, and making discrete decisions only at the highest level.

LVR systems

- Large Vocabulary Recognition (LVR) are mostly based on the principles of statistical pattern recognition.
- Basic methods to apply those principles to speech recognition were proposed by Baker, Jalinek and other researchers from IBM in '70

Basic Model

The standard approach to large vocabulary continuous speech recognition assumes a simple **probabilistic model** of speech production

A specified word sequence, W, produces an acoustic observation sequence Y, with probability **P(W,Y)**. The goal is to decode the word string, based on the acoustic observation sequence, so that the decoded string has the maximum a posteriori (**MAP**) probability.
Given the observation sequence A, this is equivalent to find

$$\widehat{W} = argmax_W \ P(W/A)$$

Using Bayes rule, we can also write

$$P(W/A) = \frac{P(A/W)}{P(A)} P(W)$$

Since P(A) is independent of W, we must find that W such that

$$\widehat{W} = argmax_W \ P(A/W)P(W)$$

P(A/W), is generally called the **acoustic model**, as it estimates the probability of a sequence of acoustic observations, conditioned on the word string.

P(W), is called the **language model**. It describes the probability associated with a postulated sequence of words

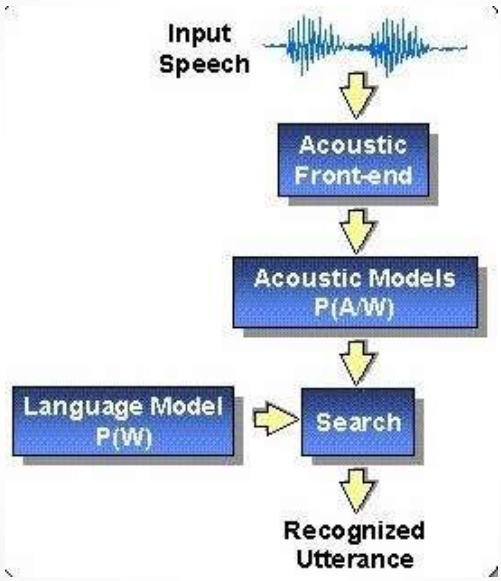
Acoustic model

- **P(A/W)** is computed. For large vocabulary speech recognition systems, it is necessary to build statistical models for sub word speech units, build up word models from these sub word speech unit models (using a lexicon to describe the composition of words), and then postulate word sequences and evaluate the acoustic model probabilities via standard concatenation methods.

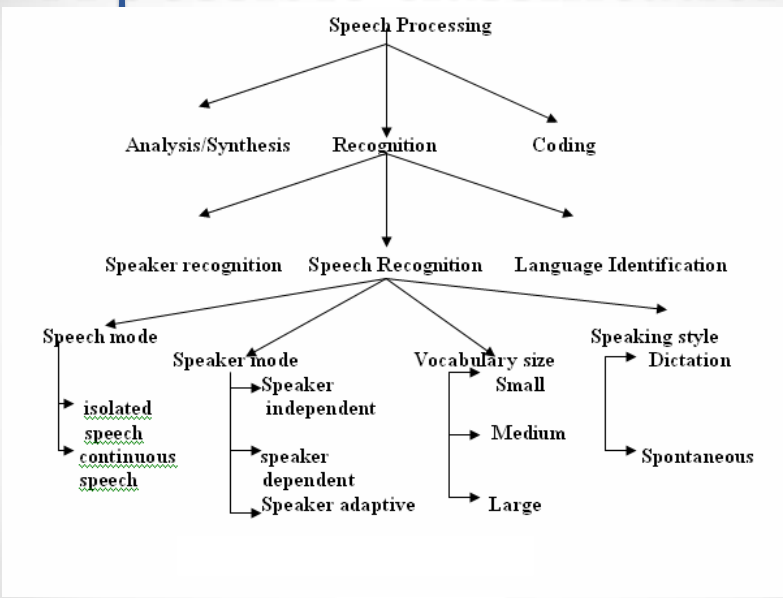
Language Model

- **P(W**) model can incorporate both syntactic and semantic constraints of the language and the recognition task.

Basic Model



A possible classification



Issues for accuracy

- A number of factors may affect recognition accuracy

Environment	**Type of noise; signal/noise ratio; working conditions**
Transducer	Microphone; telephone
Channel	Band amplitude; distortion; echo
Speakers	Speaker dependence/independence Sex, Age; physical and psychical state
Speech styles	Voice tone(quiet, normal, shouted); Production(isolated words or continuous speech read or spontaneous speech) Speed(slow, normal, fast)
Vocabulary	Characteristics of available training data; specific or generic vocabulary

Possible approaches

We may roughly identify three possible approaches to speech recognition:

- **Acoustic Phonetic** Approach
- **Pattern Recognition** Approach
- **Artificial Intelligence** Approach

Acoustic Phonetic Approach

- The acoustic phonetic approach, postulates that there exist finite, distinctive phonetic units (phonemes) in spoken language and that these units can be labeled and broadly characterized by a set of acoustics properties that are manifested in the speech signal over time.

Acoustic Phonetic Approach

- Acoustic properties of phonetic units are highly variable, both with speakers and with neighboring sounds (the so-called **co articulation** effect)
- It is assumed in the acoustic-phonetic approach that the rules governing the variability are straightforward and can be readily learned by a machine.

Acoustic Phonetic Approach

- First step: spectral analysys + feature detection to convert spectral mesures to a set of features broadly describing acoustic properties of the of the different phonetic units
  - Spectral analysis in signal processing identifies a frequency domain representation of a time domain signal, typically by means of Fourier transform
- Second step: a segmentation and labeling phase in which the speech signal is segmented into stable acoustic regions, followed by attaching one or more phonetic labels to each segmented region (phoneme lattice)
- Third step: to **determine a valid word** (or string of words) from the phonetic label sequences produced by the segmentation to labeling:l inguistic **constraints** on the task (i.e., vocabulary, syntax, and other semantic rules) are invoked in order to access the lexicon for word decoding based on the phoneme lattice.

Speech features

- In ASR systems, classification is not performed using the speech signal directly.
- The speech signal is tansformed into a sequence of ***feature vectors***, or ***parameter vectors***, and classification is performed using these feature vectors.
- The feature vectors most widely used used are **cepstral** coefficients, derived from power spectra of short windowed segments, or ***frames*** of speech.
- Each feature vector thus corresponds to a frame of speech.

Pattern Recognition Approach

- Two essential steps:

pattern training

pattern comparison.

- This approach establishes consistent **speech pattern representations**, for reliable **pattern comparison**, from a set of labeled **training samples** via a formal training algorithm.
- A speech pattern representation can be in the form of

a speech template

a statistical model (e.g., a HIDDEN MARKOV MODEL or HMM)

Dynamic Time Warping (DTW)

Vector Quantization(VQ)

- This approach can be applied to a sound (smaller than a word), a word, or a phrase.

Pattern Recognition Approach

- In the pattern-comparison stage of the approach, a direct **comparison** is made between the unknown speeches (the speech to be recognized) **with each possible pattern learned** in the training stage in order to determine the identity of the unknown according to the goodness of match of the patterns.

- It is necessary to exploit both suitable **distance measures** and **decision thresholds** (when similarity is sufficient?)

Pattern Recognition Template Based Approach

- A collection of **prototypical** speech patterns are stored as reference patterns representing the dictionary of candidate words.

- Recognition is carried out by matching an unknown spoken utterance with each of the reference templates and selecting the category of the **best matching** pattern. Usually templates for entire words are constructed.
  - Pros: errors due to segmentation or classification of smaller acoustically more variable units such as phonemes can be avoided.
  - Cons: each word must have its own full reference template; template preparation and matching become prohibitively expensive or impractical as vocabulary size increases beyond a few hundred words.

Pattern Recognition Stochastic Approach

- In general, entails the use of probabilistic models to deal with uncertain or incomplete information.
- In speech recognition, uncertainty and incompleteness arise from many sources; for example, confusable sounds, speaker variability , contextual effects, and homophones words.

- One of the most popular stochastic approaches today is based on **Hidden Markov Modeli (HMM)**
- A Hidden Markov Model is characterized by a **finite state** Markov model and a set of output distributions.
- The transition parameters in the Markov chain model temporal variabilities, while the parameters in the output distribution model spectral variabilities. These two types of variabilites are the essence of speech recognition.

Pattern Recognition Stochastic Approach

Basic definitions from Wikipedia

- In probability theory and statistics, a **Markov process**, named for the Russian mathematician Andrey Markov, is a stochastic process satisfying a certain property, called the Markov property.
- A Markov process can be though of as **'memoryless'**: loosely speaking, a process satisfies the Markov property if one can make predictions for the future of the process based solely on its present state just as well as one could knowing the process's full history. I.e., *conditional on the present state of the system, its future and past are independent*.
- A **hidden Markov model** (**HMM**) is a statistical Markov model in which the system being modeled is assumed to be a Markov process with unobserved (*hidden*) states.
- In a regular Markov model, the state is directly visible to the observer, and therefore the state transition probabilities are the only parameters. In a ***hidden* Markov model**, **the state is not directly visible**, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states.

Maria De Marsico - demarsico@di.uniroma1.it

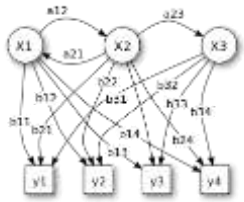
Pattern Recognition Stochastic Approach

- In its discrete form, a hidden Markov process can be visualized as a generalization of the **Urn problem**: A genie is in a room that is not visible to an observer. In this hidden room there are urns X1, X2, X3, ... each of which contains a known mix of balls, each ball labeled y1, y2, y3, ...
- The genie chooses an urn in that room and randomly draws a ball from that urn. It then puts the ball onto a conveyor belt, where the observer can observe the sequence of the balls but not the sequence of urns from which they were drawn.
- The genie has some procedure to choose urns; the choice of the urn for the *n*-th ball depends only upon a random number and the choice of the urn for the (*n* − 1)-th ball. The choice of urn does not directly depend on the urns chosen before this single previous urn; therefore, this is called a **Markov process**. It can be described by the upper part of figure.

Maria De Marsico - demarsico@di.uniroma1.it

Pattern Recognition Stochastic Approach

- The Markov process itself cannot be observed, and only the sequence of labeled balls can be observed, thus this arrangement is called a "hidden Markov process". This is illustrated by the lower part of the diagram shown in the figure , where one can see that balls y1, y2, y3, y4 can be drawn at each state.
- Even if the observer knows the composition of the urns and has just observed a sequence of three balls, *e.g.* y1, y2 and y3 on the conveyor belt, the observer still cannot be sure which urn (*i.e.*, at which state) the genie has drawn the third ball from. However, the observer can work out other details, such as the identity of the urn the genie is most likely to have drawn the third ball from.

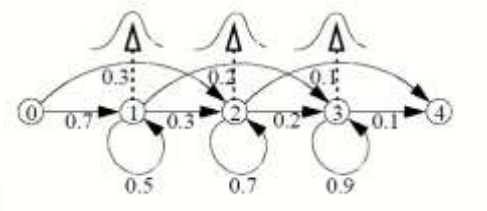


Pattern Recognition Stochastic Approach

- In speech recognition, the words a person speaks are not known but must be determined by how closely they match a model of the measurements of how they should sound.
- HMM's can be used to build such a model to provide the likelihood of a sequence of states. In speech recognition, the states comprise feature vectors; thus the HMM's yield the likehood of a particular sequence of acoustic vectors.
- In HMM-based recognition systems the mechanism that generates the sequence of feature vectors **representing any word** is modeled by an HMM. When generating the sequence, the generator is assumed to be in one of a finite set of states at any instant of time. Each state has a probability distribution function, referred to as the ***state distribution*** of that state, associated with it.
- The hidden Markov modeling paradigm assumes that to generate the feature vector at any instant, the generator draws a vector from the state distribution of the state it is in at that instant.

Pattern Recognition Stochastic Approach

- The vectors that the generator draws from a state distribution are said to **belong** to that state.
- The HMM also has a set of **transition probabilities** associated with each state.
- A generator that is in state ***i*** at time ***t*** and moves to state ***j*** at time instant ***t+1*** is said to ***transit*** from state ***i*** to state ***j*** at time instant ***t***.
- The *transition probabilities* of a state refer to the probability distribution of the states that the generator can be in at the next instant, given that it is in that state at the current instant.
- The generator draws from this distribution in order to determine which state it will be in at the next instant of time.
- The transition probabilities and the state distributions are all specific to the **word** being modeled by the HMM.



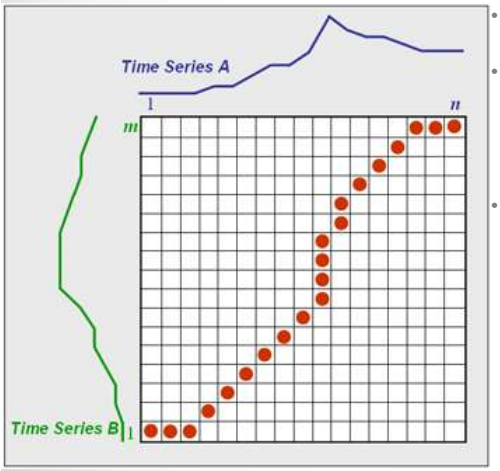
Maria De Marsico - demarsico@di.uniroma1.it

Pattern Recognition DTW Approach

- **Dynamic time warping is an algorithm for measuring**similarity between two sequences which may vary in time orspeed. For instance, similarities in walking patterns would be detected, even if in one video, the person was walking slowly and if in another, he or she were walking more quickly, or even if there were accelerations and decelerations during the course of one observation.
- DTW has been applied to video, audio, and graphics
- The sequences are "warped" non-linearly in the time dimension to determine a measure of their similarity independent of certain non-linear variations in the time dimension. This sequence alignment method is often used in the context of hidden Markov models.
- Continuity is less important in DTW than in other pattern matching algorithms; DTW is an algorithm particularly suited to matching sequences with missing information, provided there are long enough segments for matching to occur.
- The optimization process is performed using dynamic programming, hence the name.

Maria De Marsico - demarsico@di.uniroma1.it

Pattern Recognition
DTW Approach



- The two sequences A and B to match are arranged on the sides of a grid.
- Both sequences start on the bottom left of the grid. Inside each cell a distance measure can be placed, comparing the corresponding elements of the two sequences.
- The **best match** or alignment between these two sequences is given by a **path** through the grid, which minimizes the total distance between A and B.

Pattern Recognition
DTW Approach

- The procedure for finding the best alignment between A and B involves finding all possible routes through the grid and for each one compute the overall distance = the sum of the distances between the individual elements on the warping path.
- The final DTW distance between A and B is the minimum overall distance over all possible warping paths.
- It is apparent that for any pair of considerably long sequences the number of possible paths through the grid will be very large. However, the power of the DTW algorithm resides in the fact that instead of finding all possible routes through the grid, the DTW algorithm makes use of **dynamic programming** and works by keeping track of the cost of the best path at each point in the grid.

From:**Dynamic Time Warping Algorithm**
http://cst.tu-plovdiv.bg/bi/DTWimpute/DTWalgorithm.html

Pattern Recognition VQ Approach

- **Vector Quantization(VQ)** is useful for speech coders, i.e., efficient data reduction.
- **Vector quantization** is a classical quantization technique from signal processing which allows the modeling of probability density functions by the distribution of prototype vectors. It was originally used for data compression. It works by dividing a large set of points (vectors) into groups having approximately the same number of points closest to them. Each group is represented by its centroid point, as in k-means and some other clustering algorithms.
- The utility of VQ here lies in the efficiency of using compact codebooks for reference models and codebook searcher in place of more costly evaluation methods.
- For **Interactive Web Response (IWR)**, each vocabulary word gets its own VQ codebook, based on training sequence of several repetitions of the word. The test speech is evaluated by all codebooks and ASR chooses the word whose codebook yields the lowest distance measure.
- In basic VQ, codebooks have no explicit time information (e.g., the temporal order of phonetic segments in each word and their relative durations are ignored) , since codebook entries are not ordered and can come from any part of the training words.

Artificial Intelligence Approach

- The **Artificial Intelligence** approach is a hybrid exploiting both the ideas and concepts of Acoustic phonetic and pattern recognition methods. It attempts to mechanize the recognition procedure according to the way a person **applies intelligence** in analyzing, and characterizing speech based on a set of measured acoustic features.

Artificial Intelligence Knowledge based Approach

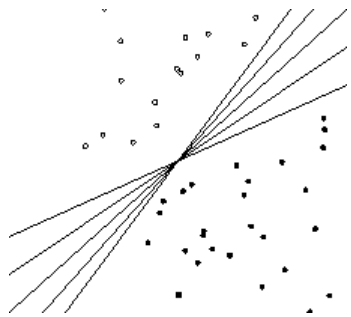
- **Knowledge based** approach uses the information regarding linguistic, phonetic and spectrogram.
- Knowledge engineering design involves the direct and explicit incorporation of **expert s speech knowledge** into a recognition system. This knowledge is usually derived from careful study of spectrograms and is incorporated using **rules** or procedures.
- Knowledge has also been used to guide the design of **the models and algorithms** of other techniques such as template matching and stochastic modeling. This form of knowledge application makes an important distinction between knowledge and algorithms. **Algorithms** enable us to solve problems. . **Knowledge** enable the algorithms to work better.

Artificial Intelligence NN Approach

- Connectionist approach or Artificial Neural Network (NN) approach aims at speech recognition by learning the relationships among phonetic events.
- Knowledge or constraints are not encoded in individual units, rules, or procedures, but distributed across many simple computing units.
- Uncertainty is modeled not as likelihoods or probability density functions of a single unit, but by the pattern of activity in many units.
- The computing units are simple in nature, and knowledge is not programmed into any individual unit function; rather, it lies in the connections and interactions between linked processing elements (resembling the style of computation in the nervous system).
- Like stochastic models, NN models rely critically on the availability of good **training** or **learning** strategies.
- **Multilayer** neural networks can be trained to generate rather complex **nonlinear** classifiers or mapping function.

Interrupt: linear classifier

- A **linear combination** is an expression constructed from a set of terms by multiplying each term by a constant and adding the results (e.g. a linear combination of *x* and *y* would be any expression of the form *ax* + *by*, where *a* and *b* are constants). In two dimensions, a linear classifier is a line.
- In two dimensions, a linear classifier is a line. Five examples are shown in figure



Maria De Marsico - demarsico@di.uniroma1.it

Interrupt: linear classifier

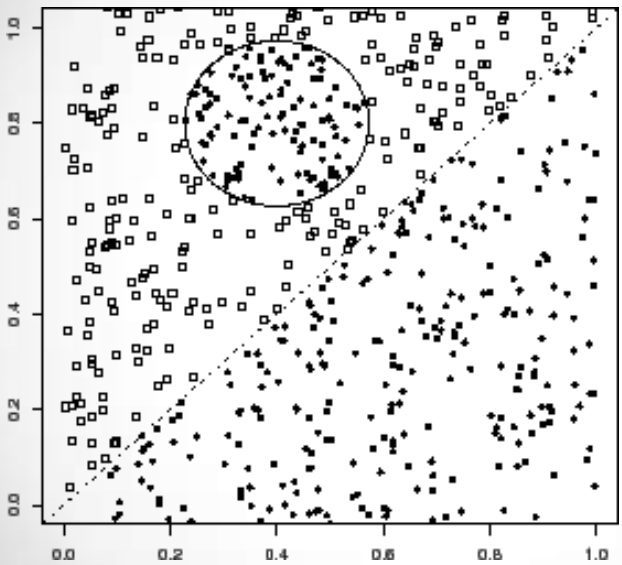
- Classification lines have the functional form $w_1\mathbf{x}_1+w_2\mathbf{x}_2=b$.
- The classification rule of a linear classifier is to assign a sample element to class **c** if $w_1\mathbf{x}_1+w_2\mathbf{x}_2>b$ and to class $\bar{c}$ if $w_1\mathbf{x}_1+w_2\mathbf{x}_2<b$.
- Here, $(x_1, x_2)^T$ is the two-dimensional vector representation of the sample to be classified and $(w_1, w_2)^T$ is the parameter vector that defines (together with *b*) the decision boundary.
- We can generalize the 2D linear classifier to higher dimensions by defining a hyperplane as:

$$\vec{w}^T\vec{x} = b$$

- The assignment criterion then is: assign to c if $\vec{w}^T\vec{x} > b$ and to $\bar{c}$ if $\vec{w}^T\vec{x} \leq b$
- We call a hyperplane that we use as a linear classifier a decision hyperplane .

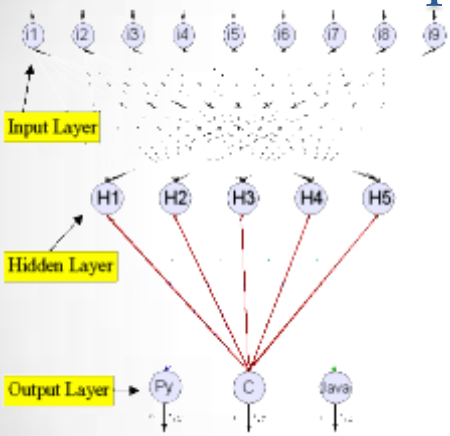
Maria De Marsico - demarsico@di.uniroma1.it

Interrupt: linear classifier



A non-linear problem

Artificial Intelligence
NN Approach



- **Pros**: The simplicity and uniformity of the underlying processing element makes connectionist models attractive for hardware implementation, which enables the operation of a net to be simulated efficiently.
- **Cons**: training often requires much iteration over large amounts of training data, and can, in some cases, be prohibitively expensive.

Image from: http://www.ibm.com/developerworks/library/l-neural/

Artificial Intelligence
SVM Approach

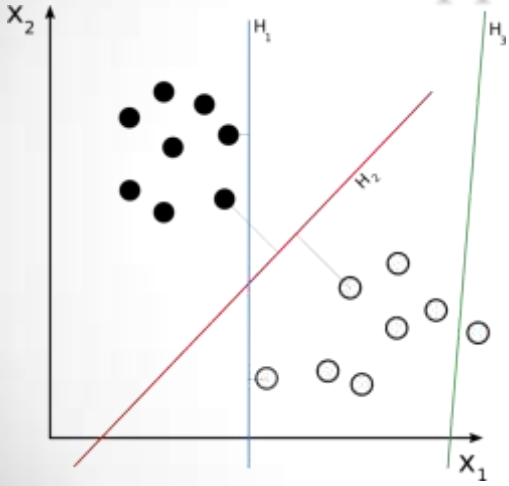
- **Support Vector Machines (SVMs)** use linear and nonlinear separating hyper-planes for data classification.
- Since SVMs can only classify fixed length data vectors, this method cannot be readily applied to task involving variable length data classification.
- The standard SVM takes a set of input data and predicts, for each given input, which of two possible classes forms the input.
- Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other.
- An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.
- SVM is a generalized linear classifier with maximum-margin fitting functions.

Artificial Intelligence
SVM Approach

- A data point is viewed as a *p*-dimensional vector (a list of *p* numbers), and we want to know whether we can separate such points with a $(p - 1)$-dimensional hyperplane. This is called a linear classifier. There are many hyperplanes that might classify the data.

- One reasonable choice as the best hyperplane is the one that represents the **largest separation**, or margin, between the two classes. So we choose the hyperplane so that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is known as the ***maximum-margin hyperplane*** and the linear classifier it defines is known as a ***maximum margin classifier.***

Artificial Intelligence
SVM Approach

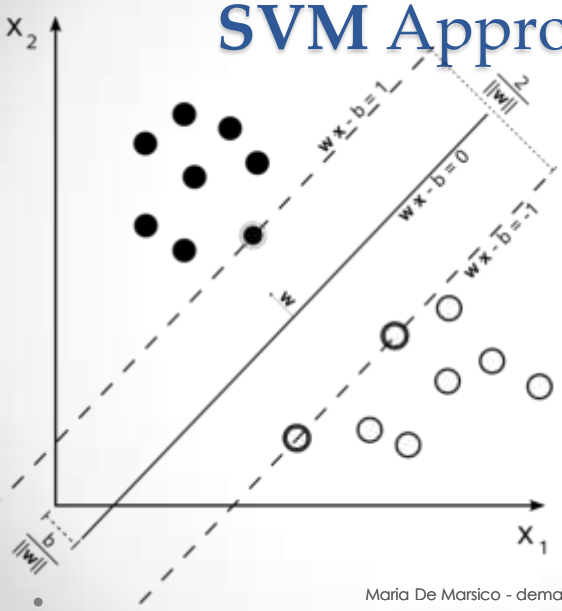


From Wikipedia:

H3 (green) doesn't separate the two classes. H1 (blue) does, with a small margin and H2 (red) with the maximum margin.

Maria De Marsico - demarsico@di.uniroma1.it

Artificial Intelligence
SVM Approach



From Wikipedia:

Maximum-margin hyperplane and margins for an SVM trained with samples from two classes. Samples on the margin are called the **support vectors**.

Maria De Marsico - demarsico@di.uniroma1.it

Artificial Intelligence
SVM Approach

- Methods for handling non-linear classification
- Methods for handling multiclass classification

Feature extraction

- The main goal of the feature extraction step is to compute a sequence of feature vectors providing a **compact** representation of the given input signal. The feature extraction is usually performed in **three** stages.
- The first stage is called the **speech analysis** or the acoustic front end. It performs some kind of spectro temporal analysis of the signal and generates raw features describing the envelope of the power spectrum of **short speech intervals**.
- The second stage compiles an **extended feature vector** composed of static and dynamic features.
- The last stage(not always present) transforms these extended feature vectors into **more compact and robust vectors** that are then supplied to the recognizer.
- There is no real consensus as to what the optimal feature sets should look like, one usually would like them to have the following properties:
  - they should allow an automatic system to **discriminate** between different through similar sounding speech sounds,
  - they should allow for the automatic creation of **acoustic models** for these sounds without the need for an excessive amount of training data,
  - they should exhibit statistics which are largely **invariant across speakers** and **speaking environment**.

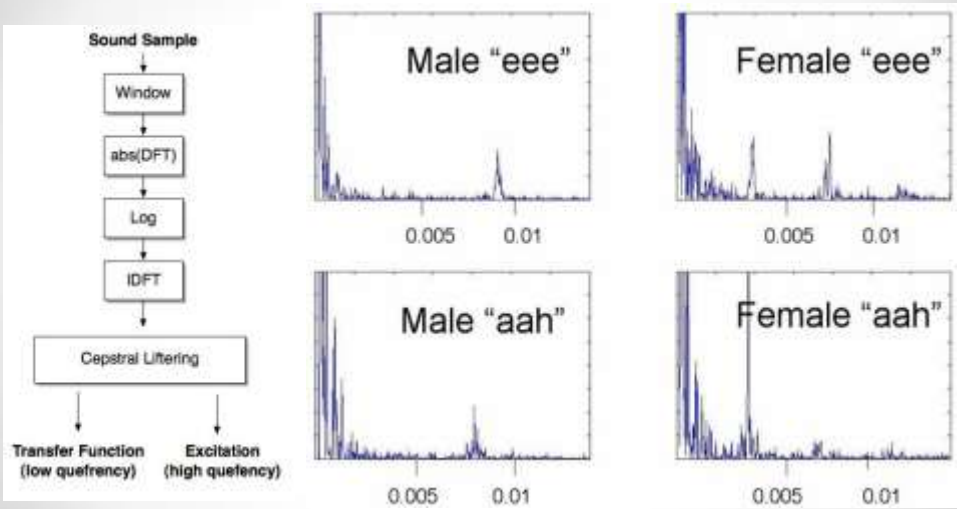
Feature extraction

- Fourier Transform provides a valuable method for analyzing the frequency spectrum of a digital signal …
- … but additional methods are needed to fully measure the features needed by a speech recognizer.
- The Mel-Frequency Cepstrum Coefficients (MFCC) representation is an example of a method that further analyzes the Fast Fourier Transform of the speech signal. The value of the method is attributed to its similarity to the functioning of the human auditory system.
- MFCC's use a mathematical transformation called the cepstrum which computes the inverse Fourier transform of the log-spectrum of the speech signal. The logarithmic nature of the technique is significant since the human auditory system perceives sound on a logarithmic scale above certain frequencies.

Cepstrum

- The **cepstrum** is a common transform used to gain information from a person's speech signal.
- It can be used to separate the **excitation** signal (which contains the **words** and the **pitch**) and the **transfer function** (which contains the voice quality).
- Cepstrum is "spectrum" with the first syllable flipped…
-

Cepstrum



Maria De Marsico - demarsico@di.uniroma1.it

Mel-frequency cepstral coefficients (MFCCs)

- The **mel (melody) scale** is a perceptual scale of pitches judged by listeners to be equal in distance from one another.
- **Mel-frequency cepstral coefficients** (**MFCCs**) are coefficients that collectively make up an MFC.

- The **difference** between the cepstrum and the mel-frequency cepstrum is that in the MFC, the frequency bands are equally spaced on the mel scale, which approximates the human auditory system's response more closely than the linearly-spaced frequency bands used in the normal cepstrum.
- MFCCs are commonly derived as follows:
- Take the Fourier transform of (a windowed excerpt of) a signal.
- Map the powers of the spectrum obtained above onto the mel scale, using triangular overlapping windows.
- Take the logs of the powers at each of the mel frequencies.
- Take the discrete cosine transform of the list of mel log powers, as if it were a signal.
- The MFCCs are the amplitudes of the resulting spectrum.

Maria De Marsico - demarsico@di.uniroma1.it

Readings

- B. Plannerer. **An Introduction to Speech Recognition** http://www.speech-recognition.de/pdf/introSR.pdf
- Kimberlee A. Kemble. **An Introduction to Speech Recognition.** ftp://service.boulder.ibm.com/software/partners/comarketing/na/ss/we/WS_Voice_Server_White_Paper.pdf
- Baker, J. ; Cohen, P. ; Cole, A. ; Jelinek, F. ; Lewis, B. ; Mercer, R. **Automatic recognition of continuously spoken sentences from a finite state grammar .** IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '78. pp.418-421
- Markov processes and HMM. http://www.google.it/url?sa=t&rct=j&q=&esrc=s&source=web&cd=6&sqi=2&ved=0CGYQFjAF&url=http%3A%2F%2Fwww.stat.columbia.edu%2F~liam%2Fteaching%2Fneurostat-spr11%2Fpapers%2Fhmm%2Frabiner.pdf&ei=sR18T9D6Fu3Z4QSMiZXWDA&usg=AFQjCNHE5XJ0bW9tZP5mPK8Tv_4Rq3d9JA
- Mixture Models. http://www.google.it/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=0CEcQFjAB&url=http%3A%2F%2Fstat.psu.edu%2F~jiali%2Fcourse%2Fstat597e%2Fnotes2%2Fmix.pdf&ei=AqR9T7CNJ8Go4gSLjvntDA&usg=AFQjCNGxvVnpYsZWq0OOIwAQ_iROi6_SPA

Readings

- **Hidden Markov Models**. http://www.google.it/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&sqi=2&ved=0CFUQFjAD&url=http%3A%2F%2Fstat.psu.edu%2F~jiali%2Fcourse%2Fstat597e%2Fnotes2%2Fhmm.pdf&ei=sR18T9D6Fu3Z4QSMiZXWDA&usg=AFQjCNGmG4HqZQB26BIfksRm5EX9BUluJQ
- **Dynamic Time Warping.** http://web.science.mq.edu.au/~cassidy/comp449/html/ch11s02.html
- **Vector Quantization**. http://www.google.it/url?sa=t&rct=j&q=&esrc=s&source=web&cd=7&sqi=2&ved=0CG4QFjAG&url=http%3A%2F%2Fwww.apl.jhu.edu%2FNotes%2FGeckle%2F525759%2Flecture8.pdf&ei=FzR8T7KvL6XP4QT7-pnpDA&usg=AFQjCNHZYPHZMbRvrpX3B5TBAVEWHnBFEg
- **Linear Classifiers**. http://webdocs.cs.ualberta.ca/~greiner/C-466/SLIDES/4-LinearClassifiers.pdf
- **Neural Networks**. http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html http://www.ibm.com/developerworks/library/l-neural/
- **Support Vector Machines**. http://ocw.metu.edu.tr/pluginfile.php/4885/mod_resource/content/0/svm.pdf