Hidden Markov Models summary

D.Makrygiannis, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr

Version 3.1

Artificial Intelligence & Information Analysis Lab

Hidden Markov Models

- **Markov Chains**
- Hidden Markov Chains
- HMMs applications
  - Speech recognition
  - Name-entity recognition
  - Human Action Recognition
  - Gesture recognition

Markov chain

A **Markov chain** is a stochastic model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous one.

Markov chains are distinguished from the fact that no matter how the process arrived at its present state, the possible future states are fixed.

Markov chain

The distribution over states can be written as $\boldsymbol{x}^{(n+1)} = \boldsymbol{x}^{(n)}\boldsymbol{P}$ *where:*

- ***x*** *is the state as* stochastic row vector
- ***P*** *is the probabilities as transition matrix*
- *n is the current time period*

Markov chain

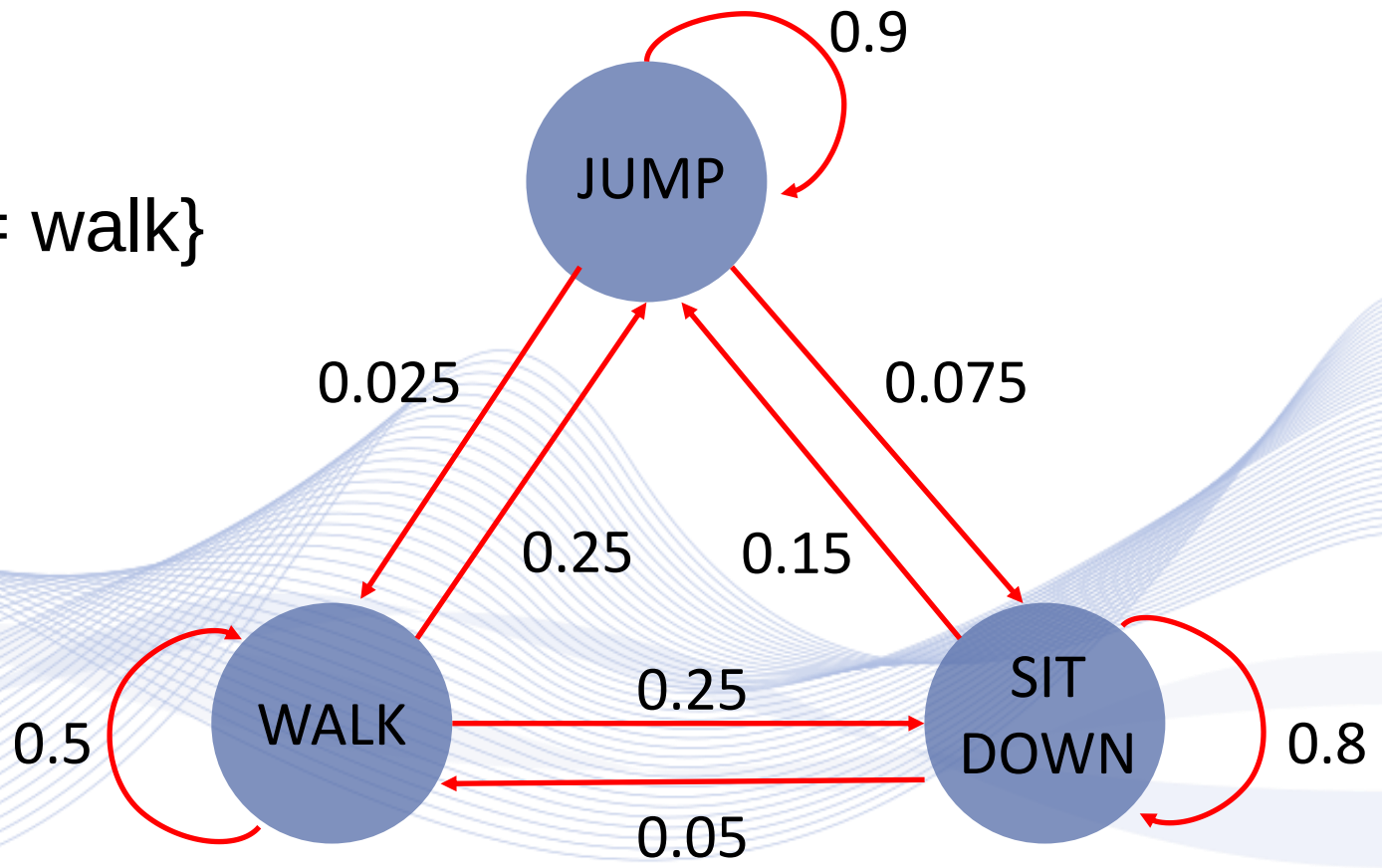
Example:

Labeling the states

{1 = jump, 2 = sit down, 3 = walk}

the transition matrix is:

$$P = \begin{bmatrix} 0.9 & 0.075 & 0.025 \\ 0.15 & 0.8 & 0.05 \\ 0.25 & 0.25 & 0.5 \end{bmatrix}$$



Markov Chain representation

Hidden Markov Models

- Markov Chains
- **Hidden Markov Chains**
- HMMs applications
  - Speech recognition
  - Name-entity recognition
  - Human Action Recognition
  - Gesture recognition

Hidden Markov Models

Markov chain introduces a non-deterministic process that generates output observation symbols in any given state.

This model is known as a **Hidden Markov Model** and can be defined as a double-stochastic process with an underlying stochastic process.

Hidden Markov Models

More specifically:

Hidden Markov Models (HMMs) are a class of probabilistic graphical **model** that allows the prediction of a sequence of unknown (**hidden**) variables from a set of observed variables

Hidden Markov Models

A Hidden Markov model is defined by:

1. A finite set of states $\Omega = \{s_1, \ldots, s_k\}$
2. A finite signal alphabet $\Sigma = \{\sigma_1, \ldots, \sigma_m\}$.
3. Initial probabilities $P(s)$ (for every $s \in \Omega$) defining the probability of starting in state s
4. Transition probabilities $P(s_i \mid s_j)$ (for every $(s_i, s_j) \in \Omega^2$) defining the probability of going from state s_j to state s_i
5. Emission probabilities $P(\sigma \mid s)$ (for every $(\sigma, s) \in \Sigma \times \Omega$) defining the probability of emitting symbol σ in state s

Hidden Markov Models- Viterbi

Given a model Φ and a sequence of observations, what is the most likely state sequence in the model that produces the observations?

Hidden Markov Models- Viterbi

This problem is solved through the Viterbi algorithm.

Optimal state sequence:

$$viterbi(i,s) \Leftrightarrow max_{s'} P(\sigma_1, \dots \sigma_i, s_{i-1} = s', s_i = s)$$

$$viterbi(1,s) = P(s)P(\sigma_i|s)$$

$$viterbi(i,s) = max_{s'} viterbi(i-1,s')P(s|s')P(\sigma_i|s)$$

$$max_s viterbi(n,s) = max_{s_1,\dots s_n} P(\sigma_1, \dots, \sigma_n, \dots, s_1, \dots s_n)$$

Hidden Markov Models- Forward-backward algorithm

Given a model Φ and a sequence of observations, what is the probability of the model that generates the observations?

Hidden Markov Models- The training problem

Training problem answers the question: Given a model structure and a set of sequences, find the model that best fits the data.

- For this problem the following 3 algorithms can be used:
  - MLE (maximum likelihood estimation)
  - Viterbi training(DO NOT confuse with Viterbi decoding)
  - Baum Welch = forward-backward algorithm

Hidden Markov Models- Baum-Welch algorithm

The Baum-Welch algorithm works in the following way:

→For each sequence in the training set of sequences:

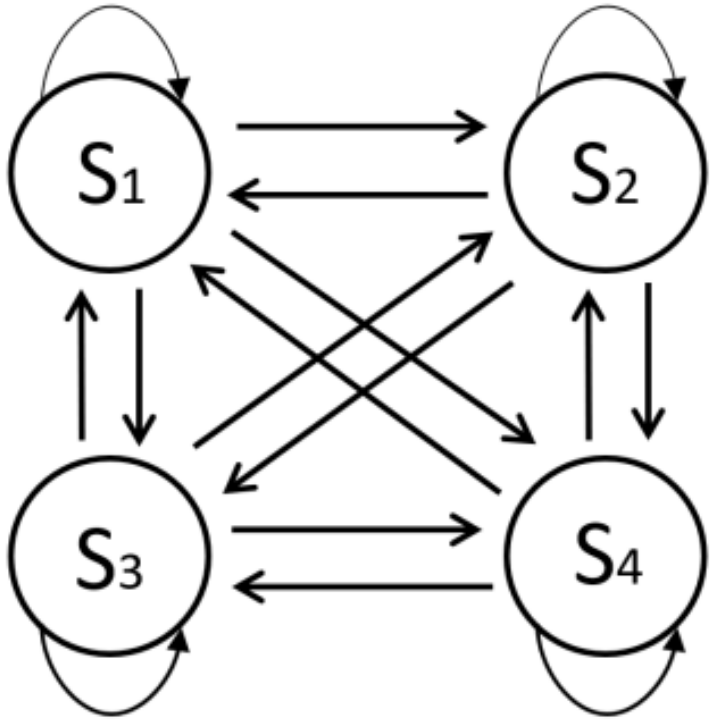
- Calculate forward probabilities with the forward algorithm
- Calculate backward probabilities with the backward algorithm
- Calculate the contributions of the current sequence to the transitions of the model & the contributions of the current sequence to the emission probabilities of the model.

Hidden Markov Models- Baum-Welch algorithm

- Calculate the new model parameters (start probabilities, transition probabilities, emission probabilities)
- Calculate the new log likelihood of the model
- Stop when the change in log likelihood is smaller than a given threshold or when a maximum number of iterations is passed.

The HMMs topologies-ergodic

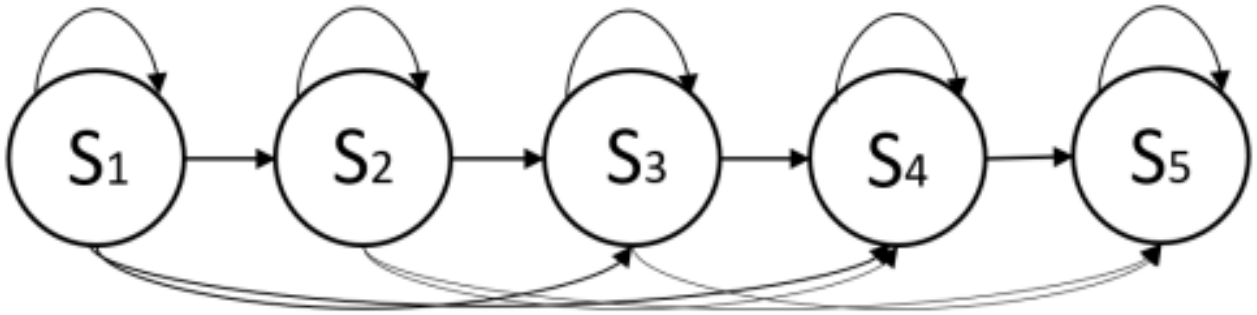
The ergodic model has no restrictions on the transition matrix. Each state is accessible from any other state with a simple transition



Ergodic topology

The HMMs topologies-left-to-right

The left-to-right models are used to depict processes characterized by variable properties over time, such as human voice signals.

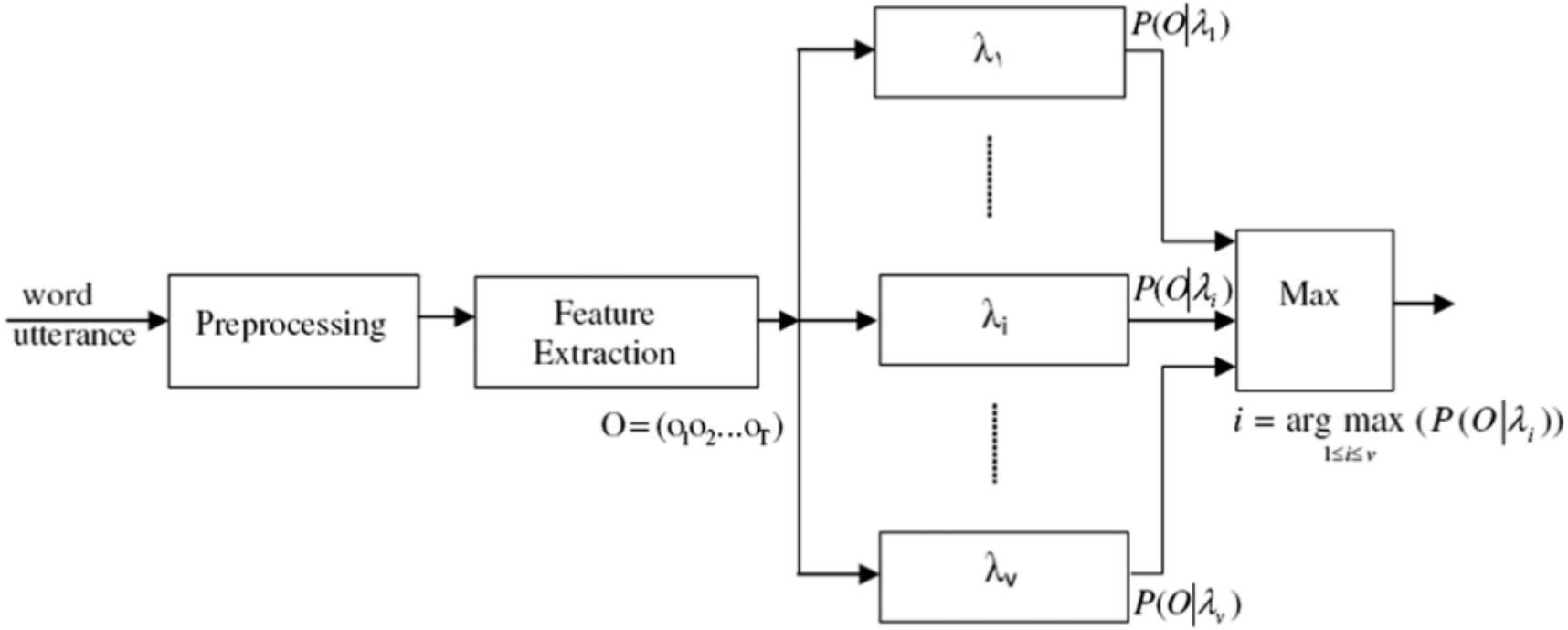


Left-to-right topology

Hidden Markov Models

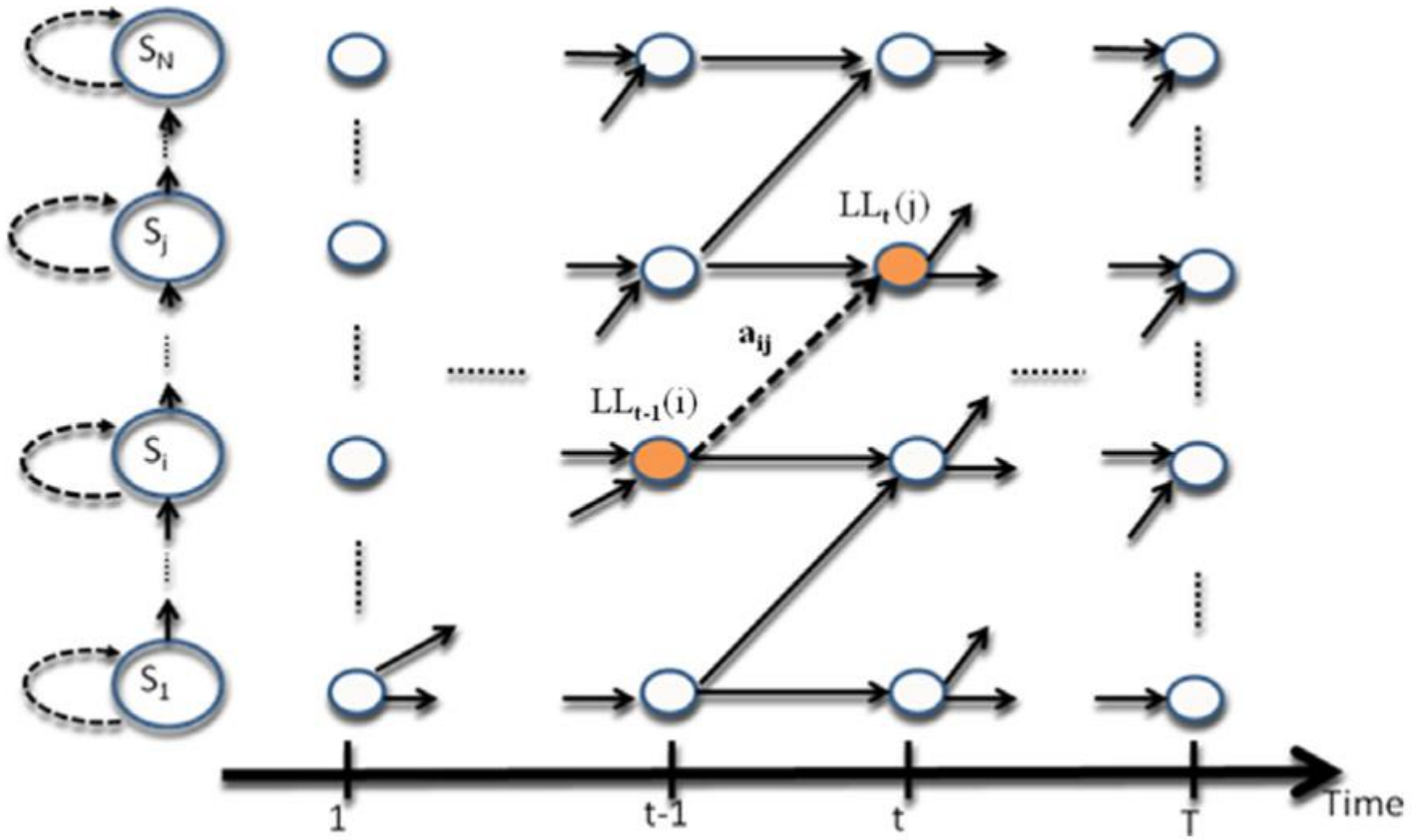
- Markov Chains
- Hidden Markov Chains
- **HMMs applications**
  - Speech recognition
  - Name-entity recognition
  - Human Action Recognition
  - Gesture recognition

HMMs for speech recognition



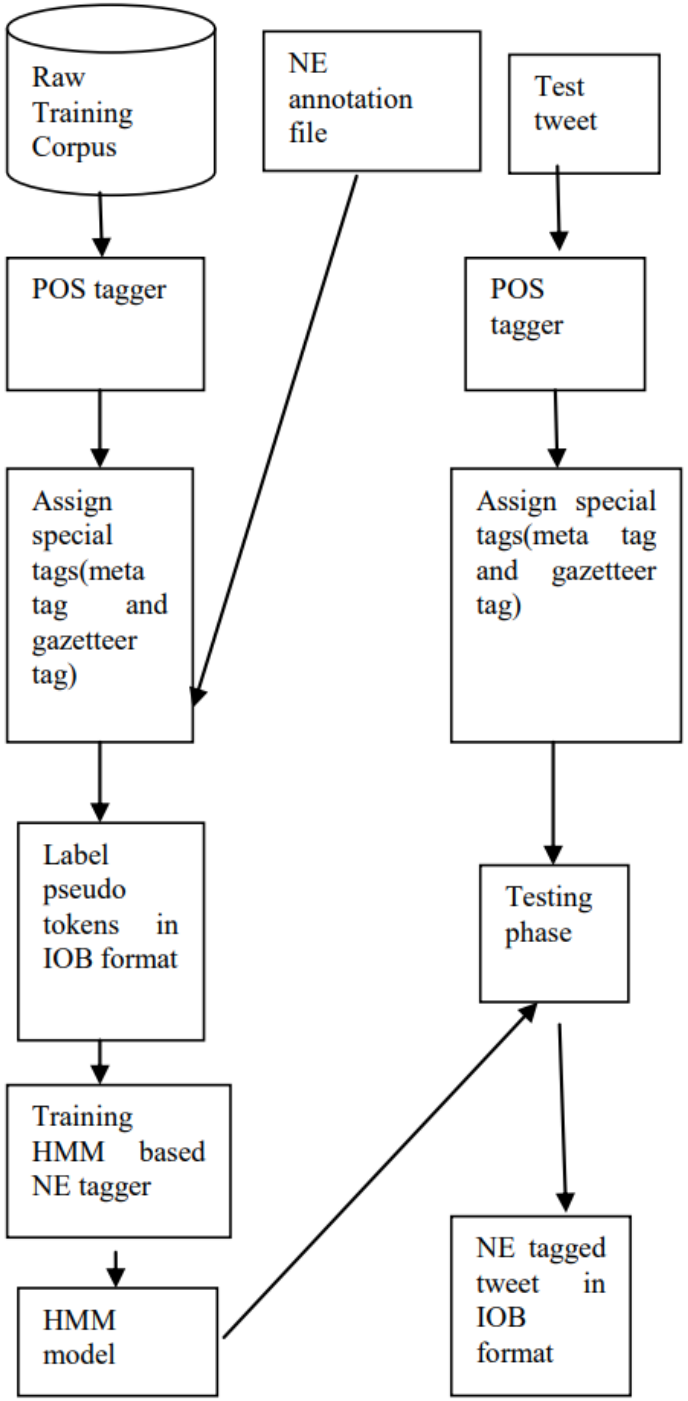
The overall block diagram of an automatic speech recognition system [NAJ2010]

HMMs for speech recognition



A Viterbi trellis diagram [NAJ2010]

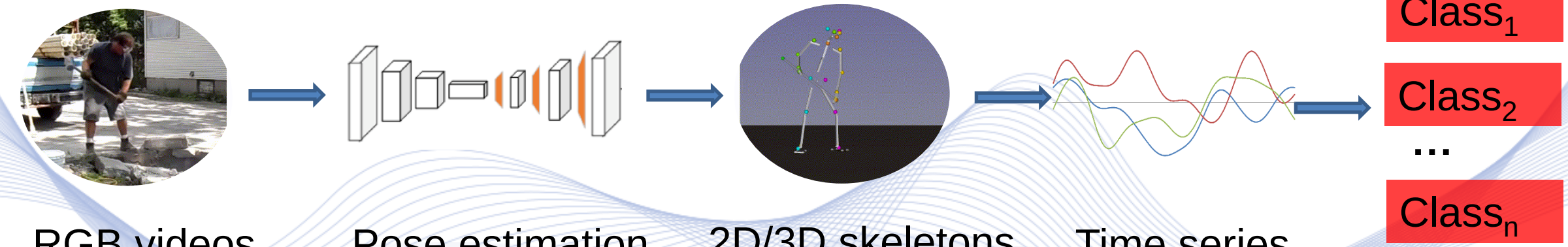
HMMs for Name-entity recognition



Architecture for an HMM based NE extraction system [SAR2015]

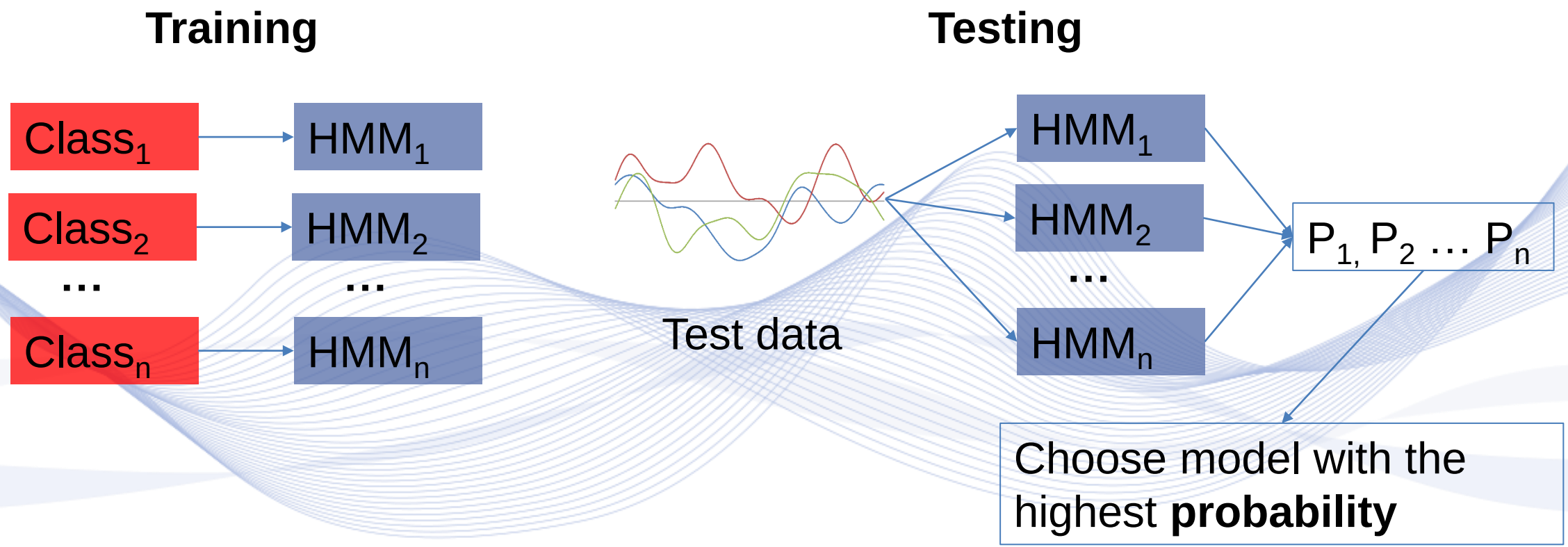
Human Action Recognition with HMMs

RGB videos to time series conversion

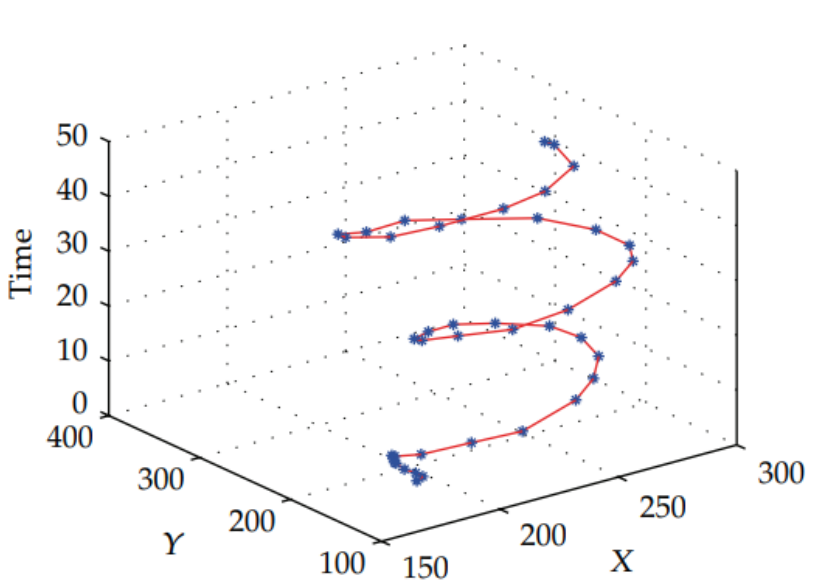


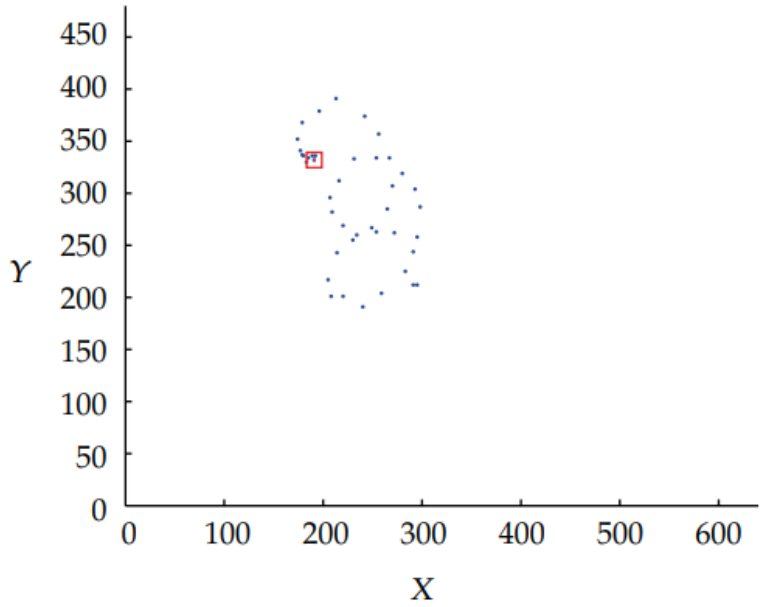
Human Action Recognition with HMMs

Supervised learning



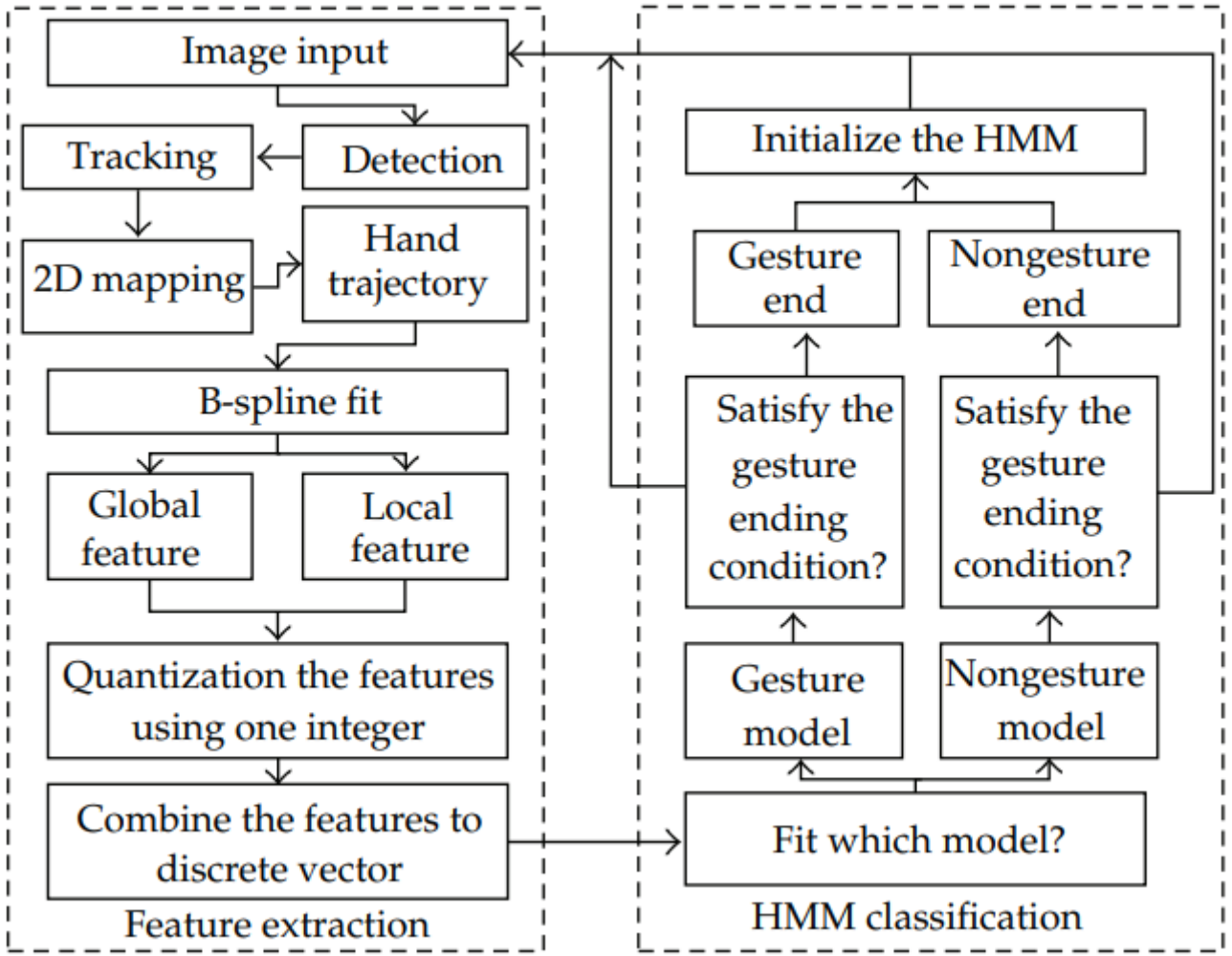
HMMs for gesture recognition



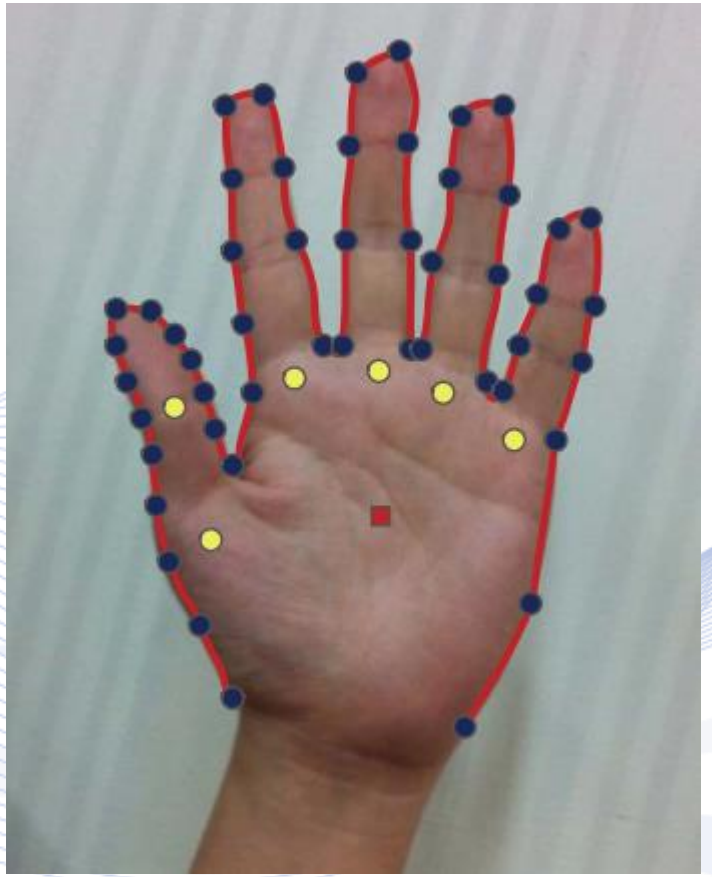


A dynamic gesture instance [WANG2012]

HMMs for gesture recognition



Overview of the hand gesture recognition process [WANG2012]



Hand contour [WANG2012]

Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr