Latent Variable Models for Structured Prediction and Content-Based Retrieval

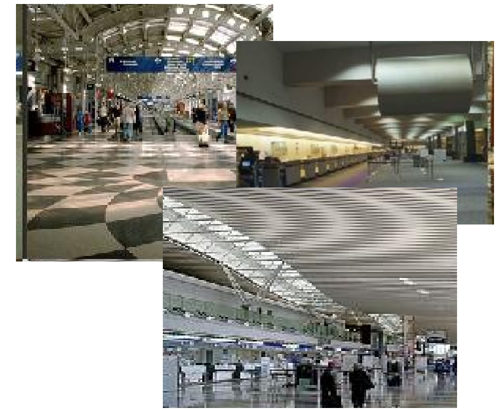
Ariadna Quattoni
Universitat Politècnica de Catalunya

Joint work with Borja Balle, Xavier Carreras, Adrià Recasens, Antonio Torralba

Scene Recognition

mountain

bar

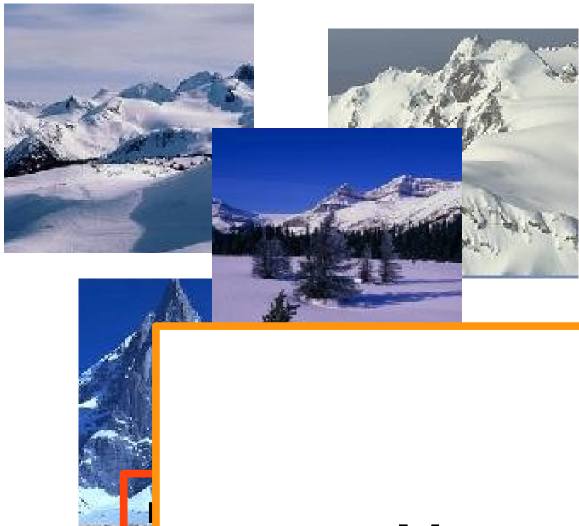


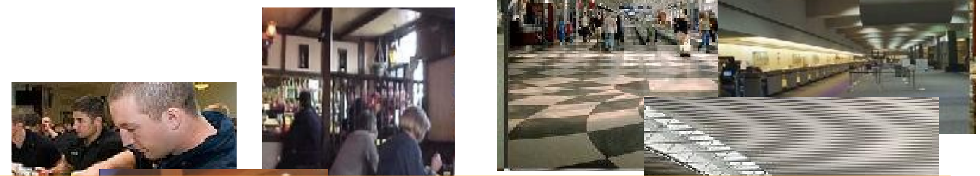
airport

Gesture Recognition

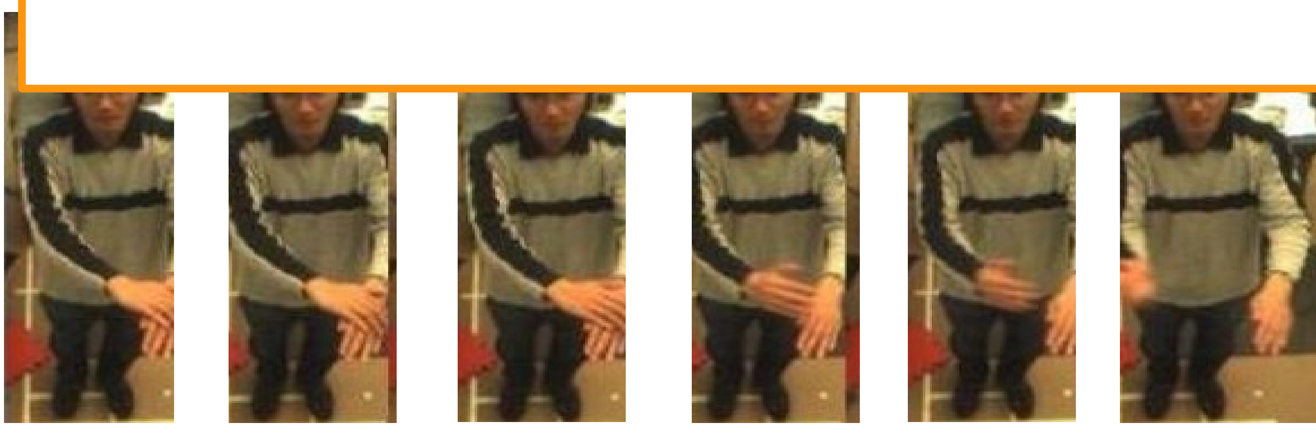
Hands Crossed	Hands Crossed	Hands Crossed	Hands Opened	Hands Opened	Hands Opened

Scene Recognition



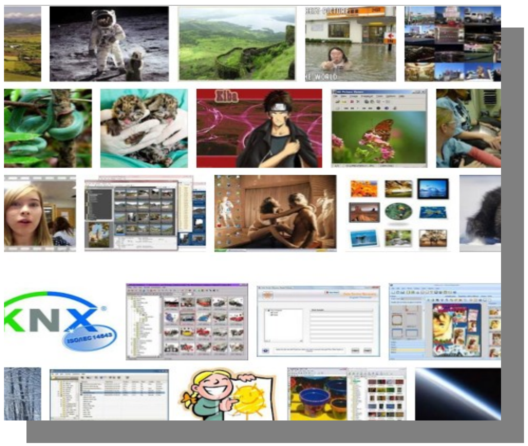


Many problems in vision involve learning mappings from complex image spaces to semantic categories.

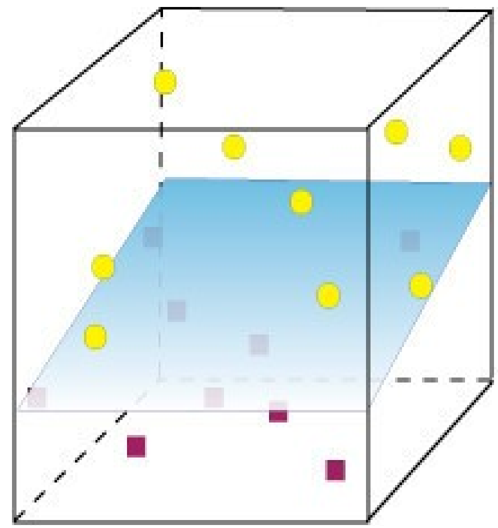


Hands Crossed	Hands Crossed	Hands Crossed	Hands Opened	Hands Opened	Hands Opened

Why Hidden Variables?



High dimensional



Low dimensional

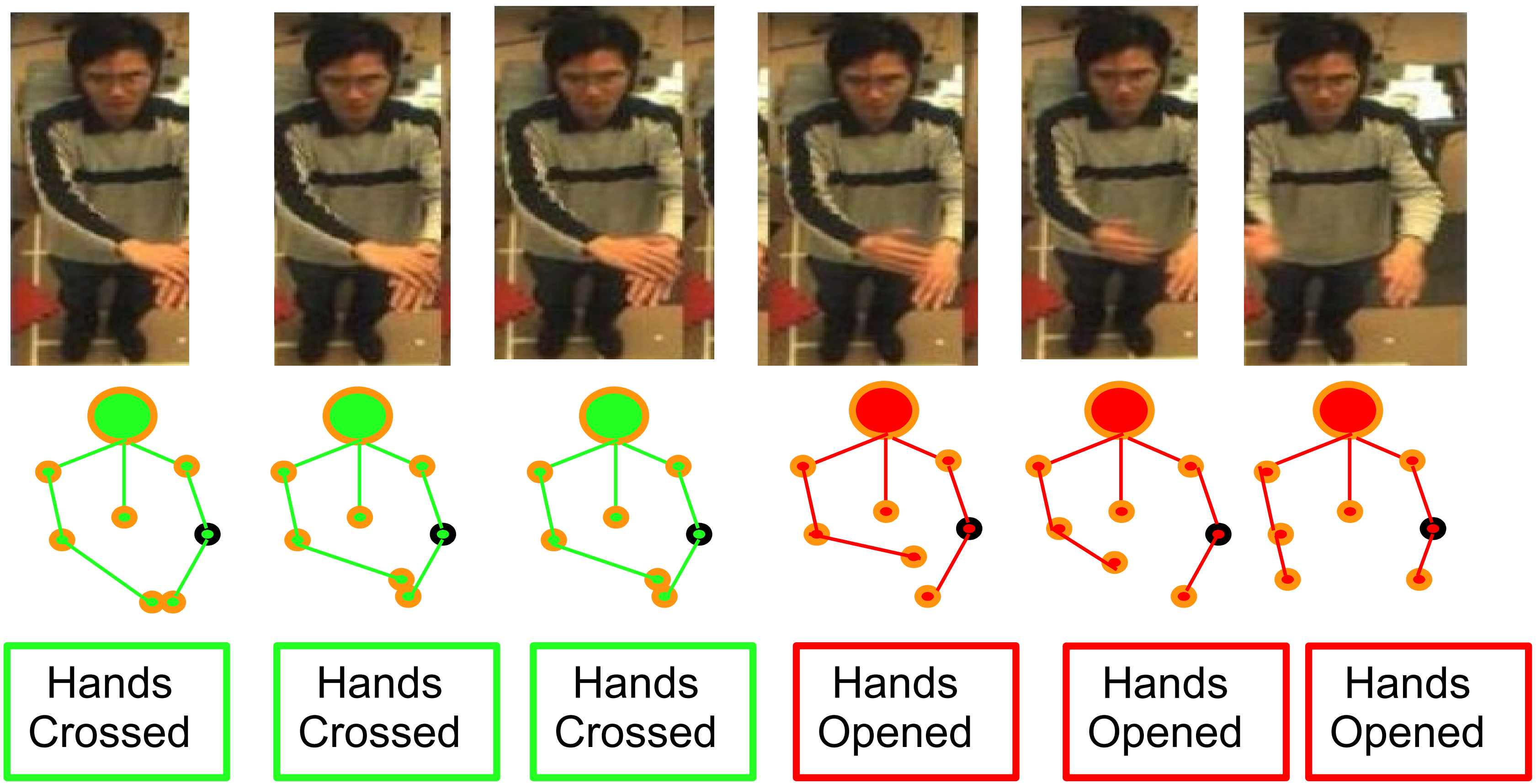
Semantic Classes

$$\mathbb{P}(y \mid x) = \sum_h \mathbb{P}(h \mid x)\mathbb{P}(y \mid x, h)$$

Classic mixture model

$$\mathrm{f}(x, y) = \sum_h \mathbb{P}(h \mid x)\ \mathrm{g}_h(x, y)$$

More general



$$\mathbb{P}(y_t \mid x_1 \ldots x_t) = \sum_{h_t} \mathbb{P}(h_t \mid x_1 \ldots x_t)\mathbb{P}(y_t \mid h_t)$$

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Structured Prediction

Non-Structured Prediction

For each input x predict a single output y

- Binary Prediction: $y \in \{-1, +1\}$
- Multiclass Prediction: $y \in \{1, \ldots, L\}$

Structured Prediction

For each input x predict a structured set of outputs y

- Binary Sequence Prediction:

  $y = [y_1, \ldots, y_m]$ where each $y_t \in \{-1, +1\}$
- Goal: capture interactions between elements of y

Temporal Dependencies

Part of Speech Tagging

He reckons the current account deficit will narrow significantly

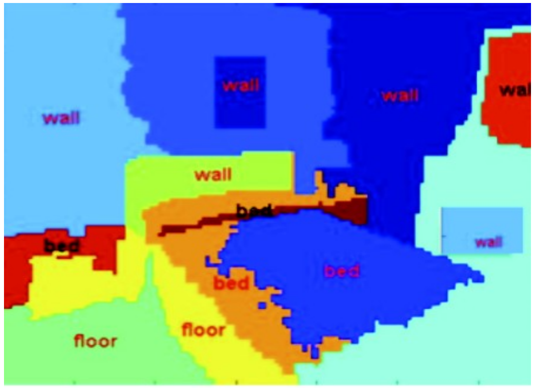
[PRP] [VB] [DT] [JJ] [NN] [NN] [MD] [VB] [RB]

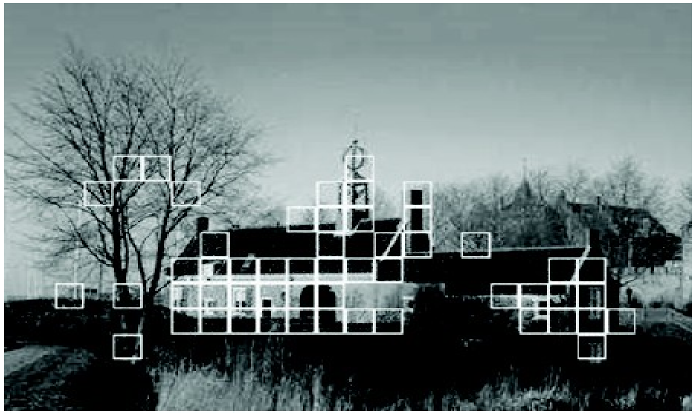
Gesture Recognition

[HTF] [HTF] [HTF] [HOF] [HOF] [HOS]

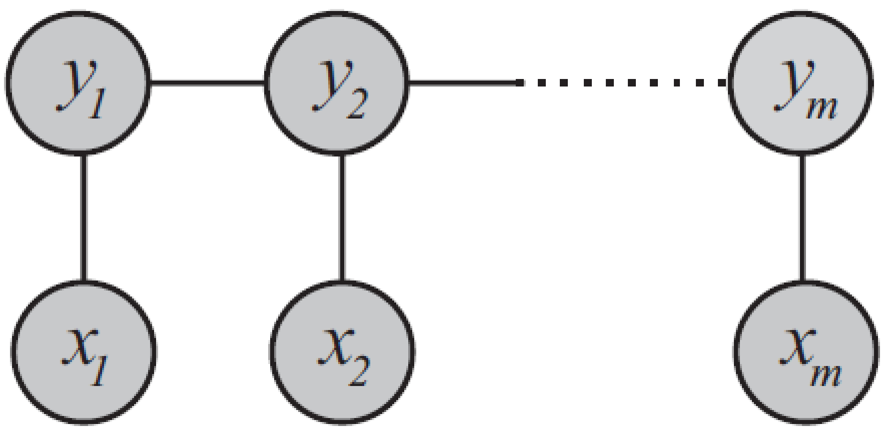
Spatial Dependencies

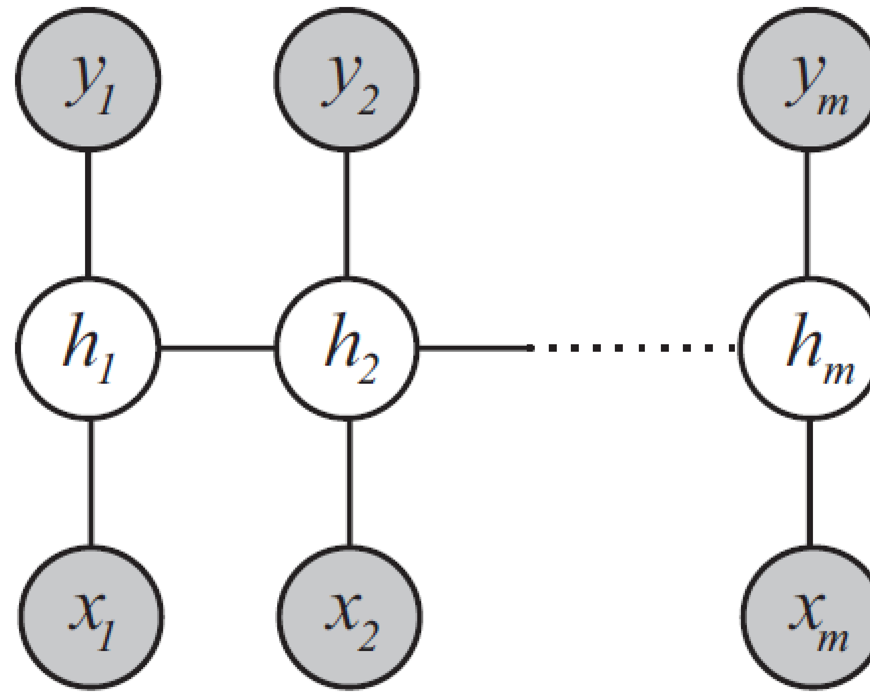
Image Annotation





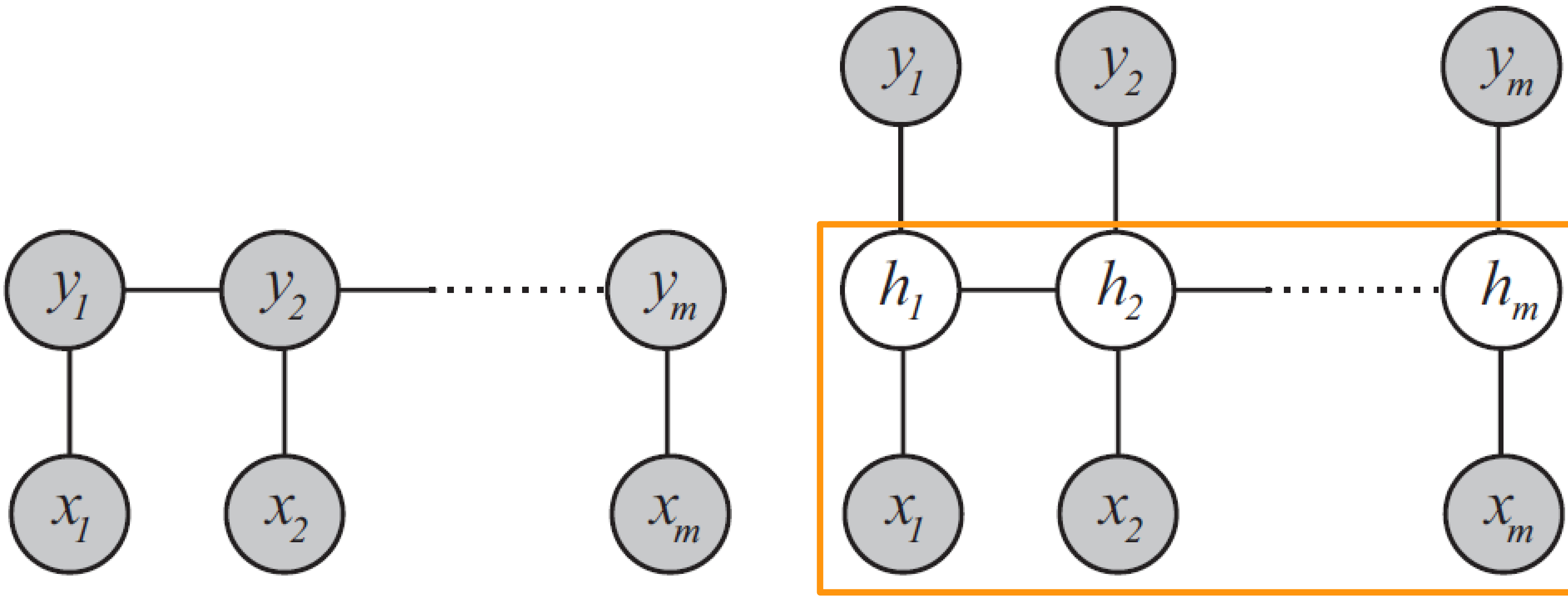
Sequence Prediction Models



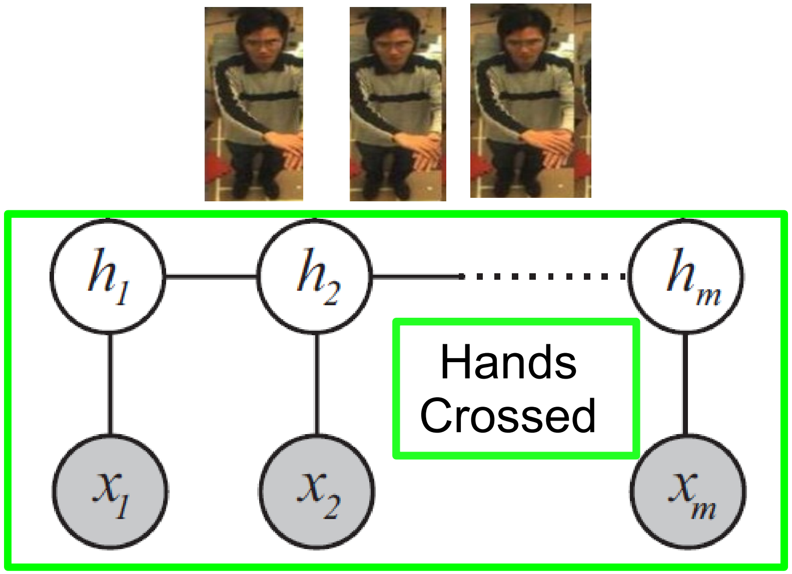


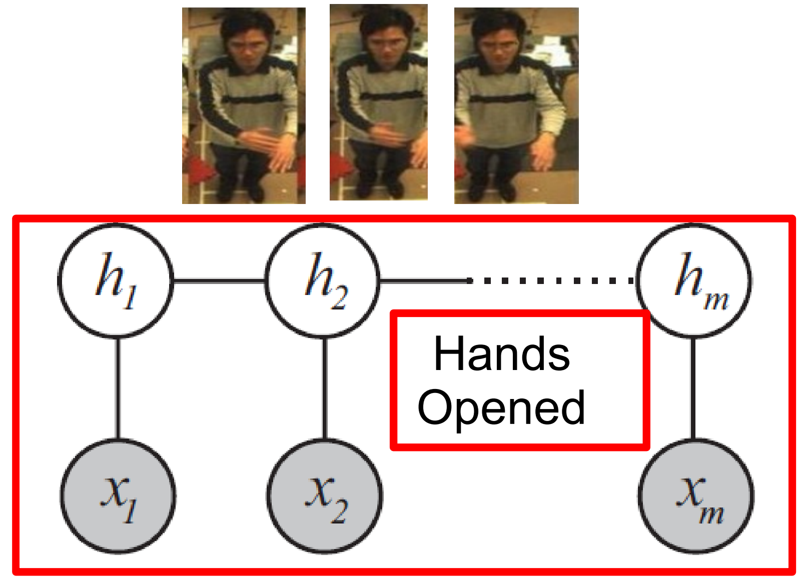
Hidden variables summarize
what is important about the past

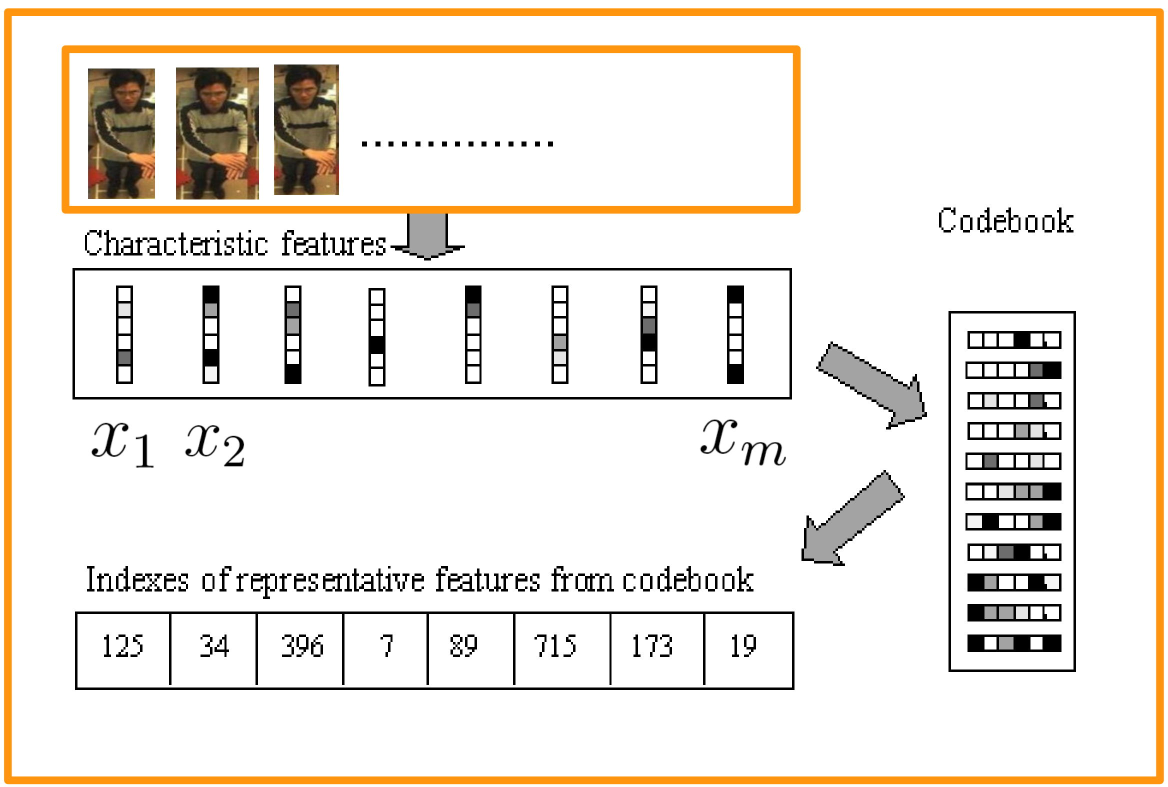
Sequence Prediction Models



Distributions over single strings.
X is discrete set.







Discretize features

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Weighted Automata Representation (WA)

Operator Model Representation (OOM)

k symbols – $x_t \in \{\sigma_1, \ldots, \sigma_k\}$

$\alpha_1 \in \mathbb{R}^n$ Initial State Vector

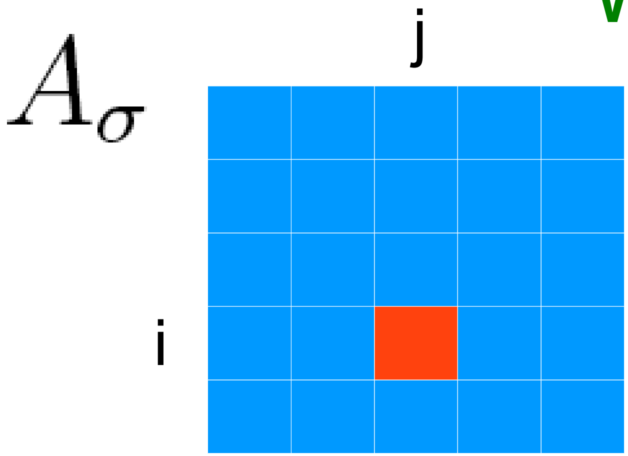
$A_\sigma \in \mathbb{R}^{n \times n}$ Model Operators

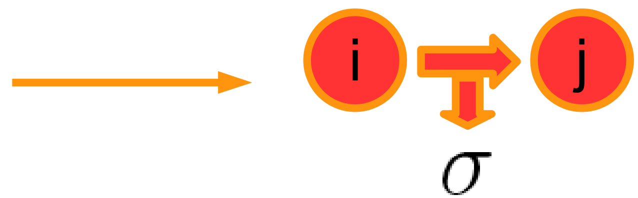
Function Parametrization

Describes the distribution as a dynamic process

$$\mathbb{P}(x_1 \ldots x_m) = \alpha_1^\top A_{x_1} \cdots A_{x_m} \vec{1}$$

Mapping from standard HMM to WA parametrization





$$
\begin{aligned}
A_\sigma(i,j) &= \mathbb{P}(h_t = j, x_t = \sigma \mid h_{t-1} = i) \\
&= \mathbb{P}(h_t = j \mid h_{t-1} = i)\ \mathbb{P}(x_t = \sigma \mid h_t = j) \\
&= T(i,j)\ O(j,\sigma)
\end{aligned}
$$

$$\mathbb{P}(x_1 \ldots x_m) = \alpha_1^\top A_{x_1} \cdots A_{x_m} \vec{1}$$

Forward-Backward Equations

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Why Spectral Learning?

Spectral Learning: Algebraic method for recovering model parameters from observable statistics.

These methods exploit directly the markovianity of the process

They are fast, simple and scale easily to large datasets

Much faster than alternative approaches based on Expectation Minimization

Hankel Matrix

$$H_{\mathbb{P}}(p, s) = \mathbb{P}(p \cdot s)$$

$$\begin{array}{c|c} \Sigma^\star & \begin{matrix} \lambda & a & b & aa & ab & \dots \end{matrix} \\ \begin{matrix} \lambda \\ a \\ b \\ aa \\ ab \\ \vdots \end{matrix} & \begin{bmatrix} 1 & 0.3 & 0.7 & 0.05 & 0.25 & \dots \\ 0.3 & 0.05 & 0.25 & 0.02 & 0.03 & \dots \\ 0.7 & 0.6 & 0.1 & 0.03 & 0.2 & \dots \\ 0.05 & 0.02 & 0.03 & 0.017 & 0.003 & \dots \\ 0.25 & 0.23 & 0.02 & 0.11 & 0.12 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \end{array}$$

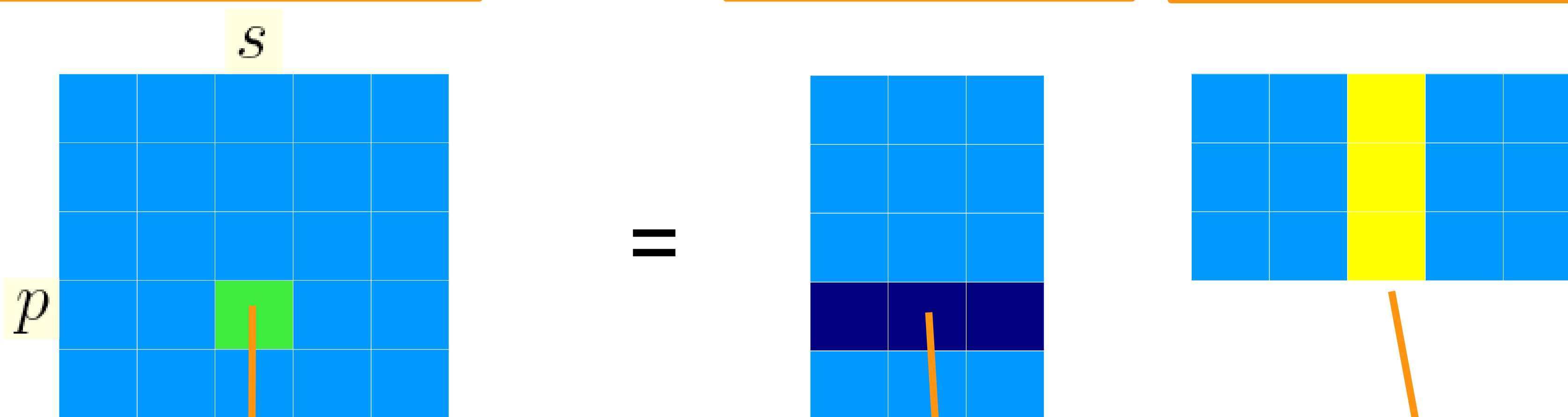
Distribution generated by a WA with *n* states → $\mathrm{rank}(H_{\mathbb{P}}) = n$

H Sub-block – defined by a basis $P, S \in \Sigma^*$

$$\begin{array}{c|c} \Sigma^\star & \begin{matrix} \lambda & a & b & aa & ab & \dots \end{matrix} \\ \begin{matrix} \lambda \\ a \\ b \\ aa \\ ab \\ \vdots \end{matrix} & \begin{bmatrix} 1 & 0.3 & 0.7 & 0.05 & 0.25 & \dots \\ 0.3 & \boxed{0.05} & \boxed{0.25} & 0.02 & 0.03 & \dots \\ 0.7 & \boxed{0.6} & \boxed{0.1} & 0.03 & 0.2 & \dots \\ 0.05 & 0.02 & 0.03 & 0.017 & 0.003 & \dots \\ 0.25 & 0.23 & 0.02 & 0.11 & 0.12 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \end{array}$$

Duality between *n*-rank factorizations of Hankel and WAs

$H \in \mathbb{R}^{|P|\times|S|}$ $\quad F \in \mathbb{R}^{|P|\times n}$ $\quad B \in \mathbb{R}^{n\times|S|}$

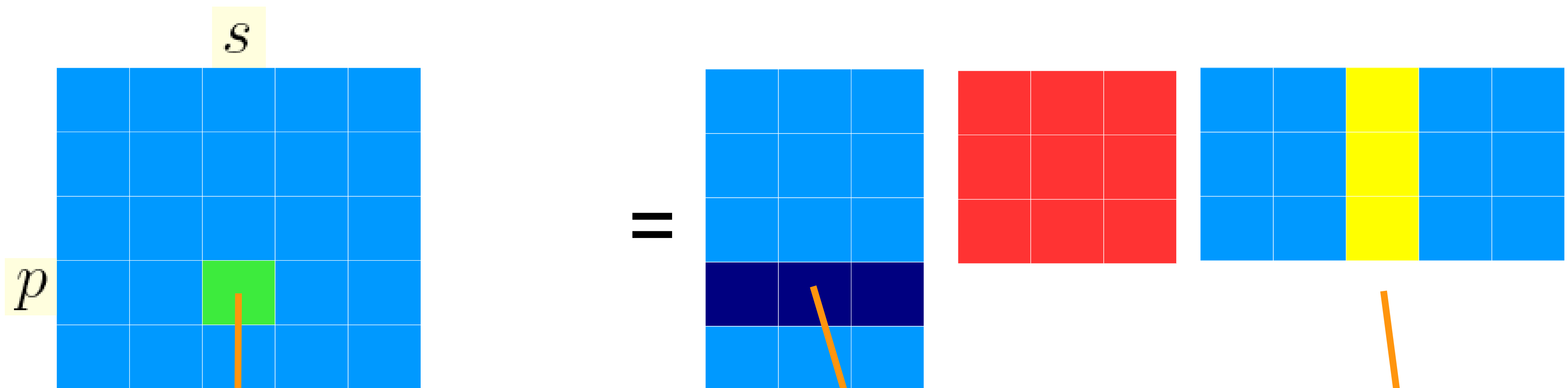


$$\mathbb{P}(p_1 \ldots p_m \cdot s_1 \ldots s_{m'}) = \overbrace{\alpha_1^\top A_{p_1} \cdots A_{p_m}} \cdot \overbrace{A_{s_1} \ldots A_{s_{m'}} \vec{1}}$$

$$H = FB$$

Recovering Operators

$H_\sigma \in \mathbb{R}^{|P|\times|S|}$ $\quad$ $F \in \mathbb{R}^{|P|\times n}$ $\quad$ $B \in \mathbb{R}^{n\times|S|}$



$$\mathbb{P}(p_1 \dots p_m \cdot \sigma \cdot s_1 \dots s_{m'}) = \overbrace{\alpha_1^\top A_{p_1} \cdots A_{p_m}} \cdot A_\sigma \cdot \overbrace{A_{s_1} \dots A_{s_{m'}} \vec{1}}$$

$$H_\sigma = F A_\sigma B \qquad A_\sigma = B^+ H F^+$$

Spectral Method

We can recover a parametrization for the distribution from (almost) any rank-*n* factorization of *H*.

The spectral method uses the thin SVD factorization.

- Input: a training set of sequences;
  the number of states n;
- Output: Model Parameters $\alpha_1, A_{\sigma_1}, \ldots, A_{\sigma_k}$
- Algorithm
  1. Choose a set of prefixes and suffixes P and S
  2. Estimate H from training samples
  3. Obtain the *thin SVD* of $H = [UD][V^\top]$
  4. Compute $A_\sigma = (HV)^+(H_\sigma V)$

Costs depends on number of prefixes and suffixes

Discrete Homogeneous HMM

- n states
- k symbols – $x_t \in \{\sigma_1, \ldots, \sigma_k\}$
- Probabilities arranged into matrices $H, H_{\sigma_1}, \ldots, H_{\sigma_k} \in \mathbb{R}^{k \times k}$

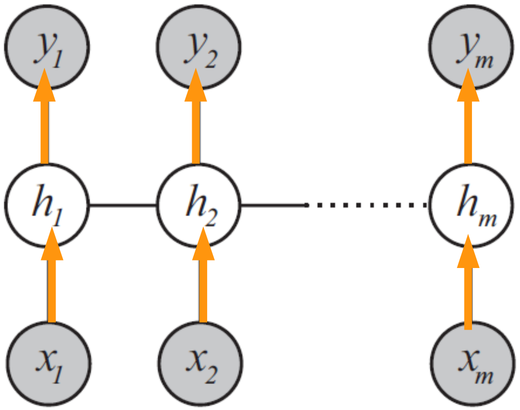
$$
\begin{aligned}
H(i,j) &= \mathbb{P}(x_t = \sigma_i, x_{t+1} = j) \\
H_{\sigma}(i,j) &= \mathbb{P}(x_{t-1} = \sigma_i, x_t = \sigma, x_{t+1} = \sigma_j)
\end{aligned}
$$

1. Compute SVD $H = UDV^{\top}$ and take top n right singular vectors V_n
2. $A_\sigma = (HV_n)^{+}(H_\sigma V_n)$

Modeling paired sequences

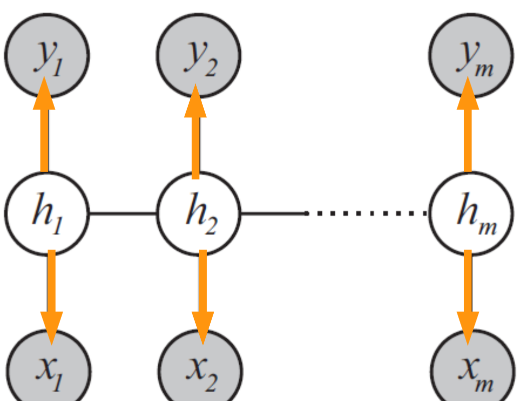
k input symbols – $x_t \in \sigma_1, \ldots, \sigma_k$
l output symbols – $y_t \in \tau_1, \ldots, \tau_l$

Conditional



$$\mathbb{P}(y_1 \ldots y_m \mid x_1 \ldots x_m) = \alpha_1^\top A_{x_1}^{y_1} \cdots A_{x_m}^{y_m} \vec{1}$$
$$H(i,j) = \mathbb{P}(y_t = \tau_i, y_{t+1} = \tau_j)$$
$$H_{\sigma,\tau}(i,j) = \mathbb{P}(y_{t-1} = \tau_i, y_t = \tau, y_{t+1} = \tau_j \mid x_t = \sigma)$$

Joint



$$\mathbb{P}(x_1 \ldots x_m, y_1 \ldots y_m) = \alpha_1^\top A_{x_1}^{y_1} \cdots A_{x_m}^{y_m} \vec{1}$$
$$H(i,j) = \mathbb{P}(x_t = \sigma_i, x_{t+1} = \sigma_j)$$
$$H_{\sigma,\tau}(i,j) = \mathbb{P}(x_{t-1} = \sigma_i, x_t = \sigma, x_{t+1} = \sigma_j, y_t = \tau)$$

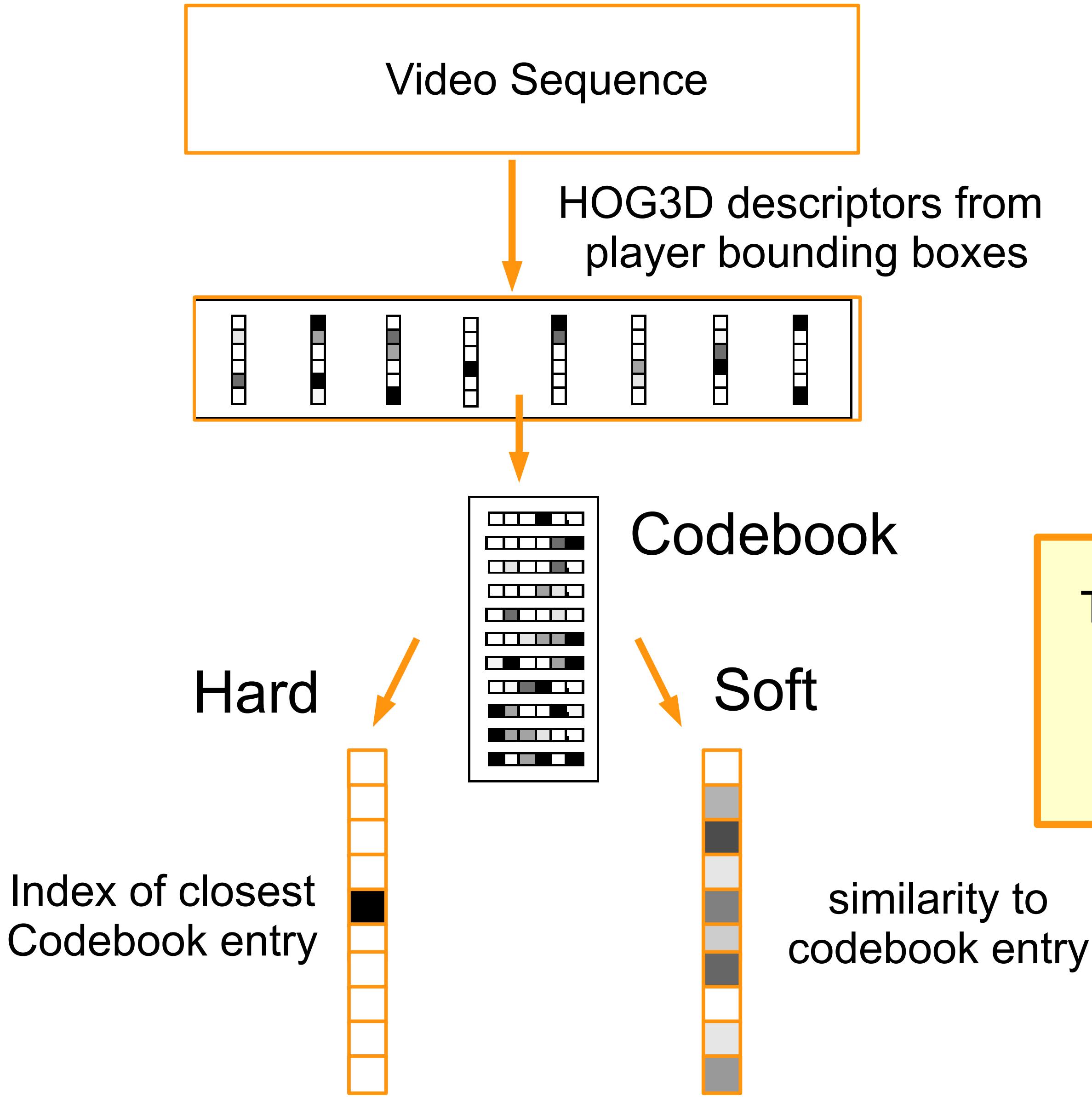
Experiments

The task: Recognize actions in tennis (serve, hit , non-hit, ...)

The experimental setting:

- Take sequences from 4 games and cut them in subsequences.
- Random partition sub-sequences into training and test.
- Evaluation metric: average F1 (geometric mean of precision and recall)

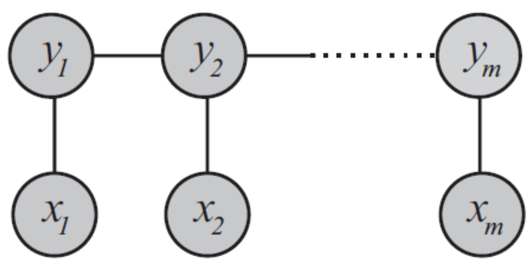
The features



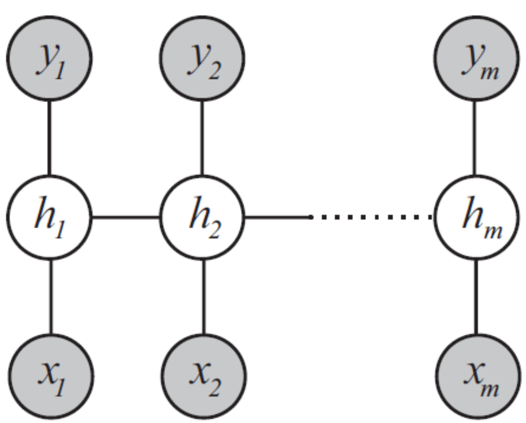
The spectral method can also be used for the soft representation

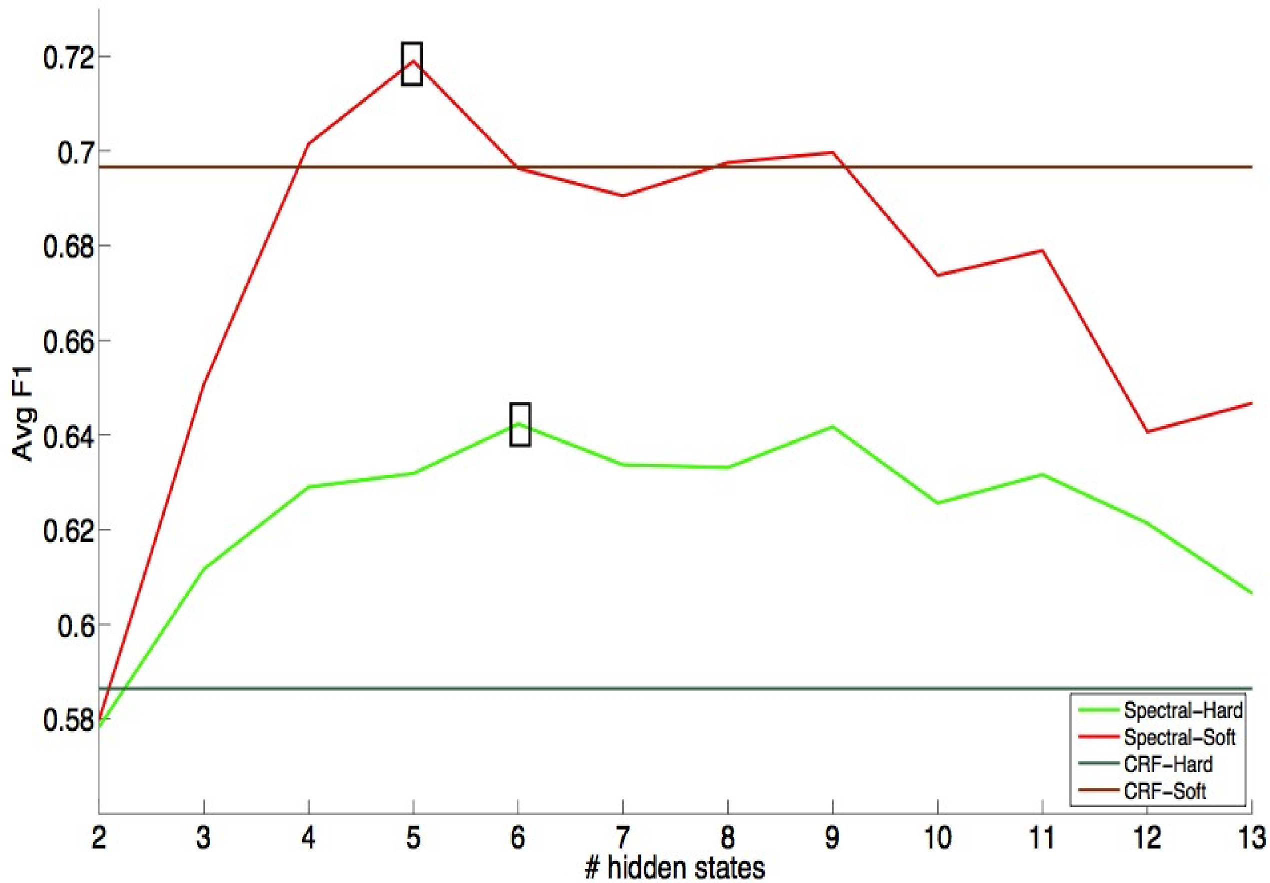
Results

CRF



Spectral Joint





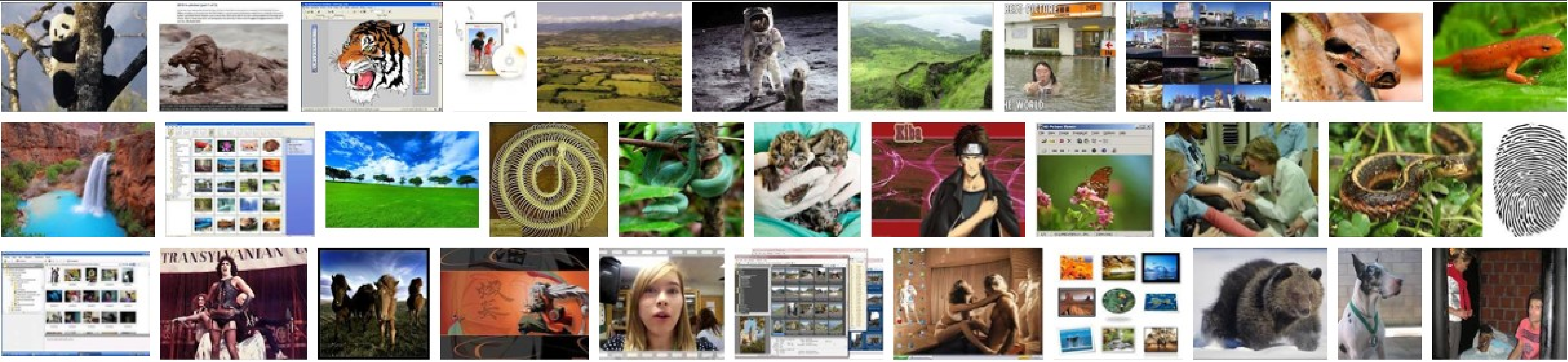
Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments



Página 2


Complex Image Space

Color important for outdoor images less important for indoor images

Latent classes can model variability

Outline

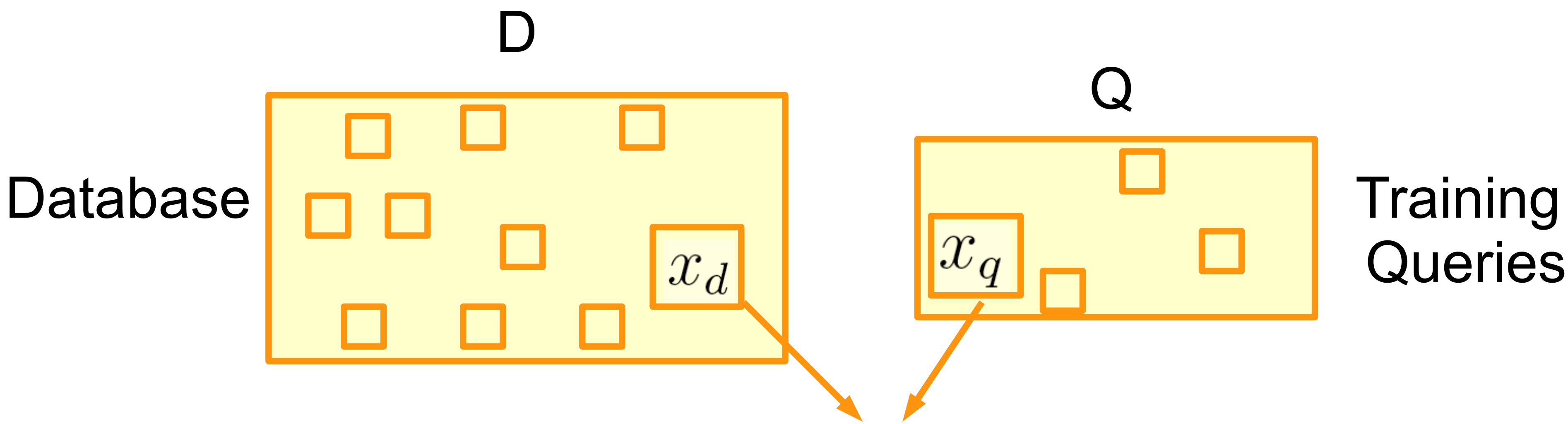
Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Global Ranking Model



elementary relevance functions

$$k_{q,d}(i,j) = r_j(x_q, x_d)$$

global ranking function

$$R(z, x_q, x_d) = z^\top k_{q,d}$$

Global Ranking Model

$$C = \{c_1, \ldots, c_l\}$$

set of triplet constraints

$$\langle q, a, b \rangle$$

database item *a* is more relevant to *q* than item *b*

Ranking Loss function

$$L(C, z) = \sum_{\langle q,a,b \rangle \in C} \max \left[0 \ , \ R(z, q, b) - R(z, q, a) + 1 \right]$$

$$\min_z \left\{ L(C, z) + \frac{\lambda}{2} \|z\|^2 \right\}$$

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Ranking model with latent variables

Mixture of *specialized* ranking functions

$$R(q,d) = \sum_{g \in G} \mathbb{P}(h_q = g)\ z_g^\top k_{q,d}$$

One for each latent class

Log-linear model

Feature representation

$$\mathbb{P}(h_q = g) = \frac{\exp^{w_g^\top x_q}}{\sum_{g' \in G} \exp^{w_{g'}^\top x_q}}$$

Ranking model with latent variables

Ranking Loss function

$$\min_{Z,W}\left\{L(C,Z,W)+\frac{\lambda_z}{2}\|Z\|^2+\frac{\lambda_w}{2}\|W\|^2\right\}$$

$$W=[w_1,\ldots,w_G]$$

$$Z=[z_1,\ldots,z_G]$$

Alternating optimization strategy

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

Parameter estimation

constraints with non-zero loss

$$\boxed{\Delta = \{\langle q, a, b\rangle \in C \text{ such that } R(q,a) - R(q,b) < 1\}}$$

subgradient with respect to $Z_{g,j}$

$$\boxed{\sum_{\langle q,a,b\rangle \in \Delta} \mathbb{P}(h_q = g)(k_{q,b}(j) - k_{q,a}(j))}$$

The influence of query *q* in the update of relevance function *g* is weighted by the probability that *q* belongs to class *g*.

Parameter Estimation

$$\epsilon_{q,g} = \sum_{a,b \text{ s.t. } \langle q,a,b \rangle \in \Delta} z_g^\top (k_{q,b} - k_{q,a})$$

more negative = better ranking performance of class *g* for query *q*

$$\sum_{q \in Q} \epsilon_{q,g} \left[\mathbb{P}(h_q = g) - \mathbb{P}(h_q = g)^2 \right] x_q(j)$$

subgradient with respect to $W_{g,j}$

If class *g* predicts good rankings for constraints of query *q*, then the update will increase the probability that *q* belongs to *g*.

Outline

Latent Variable Models for Structured

- Structure Prediction Problem
- Representing distributions using WA
- Spectral learning algorithm
- Examples

Mixture Model for Content-Based Image Retrieval

- Global Ranking Model
- Mixture Ranking Model
- Learning a Ranking Function
- Experiments

SUN Dataset

12000 images, indoor and outdoor scenes.

Images are annotated with object tags.

Ground-truth relevance function derived from object annotations.

5 Random Partitions:
- 6000 database images
- 2000 train queries
- 1000 validation queries
- 2000 test queries
- 1000 novel-database

Ground-Truth Constraints

We create ranking constraints for each train query:

1- Find top K database nearest neighbors according to ground-truth relevance function.

2- Sample L items from remaining items

3- Generate KL ranking triplets

320,000 total number of ranking triplets.

Model Comparisons

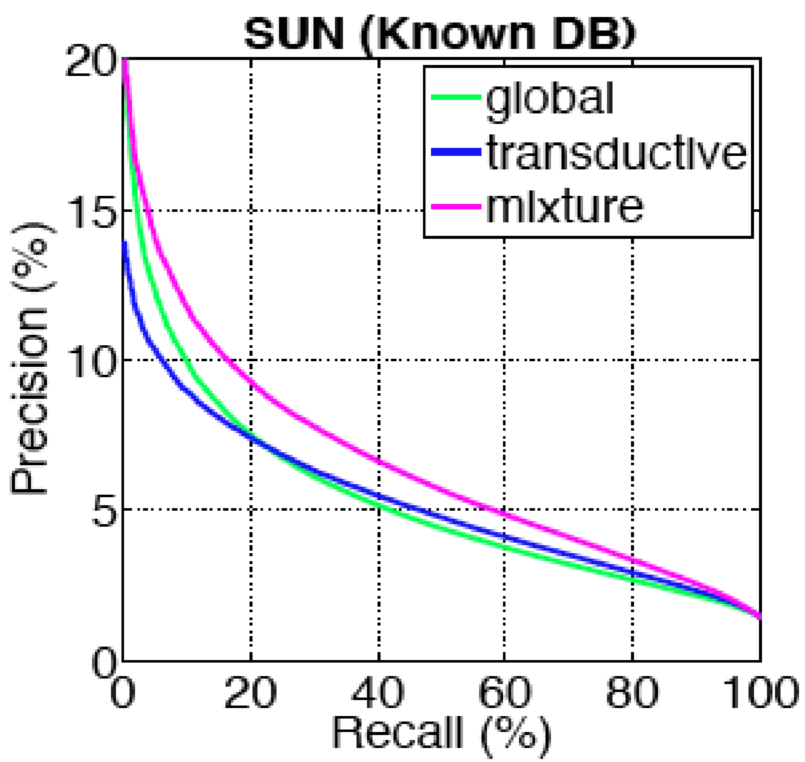
Global SVM:
Learns a single weighted combination of elementary relevance functions.

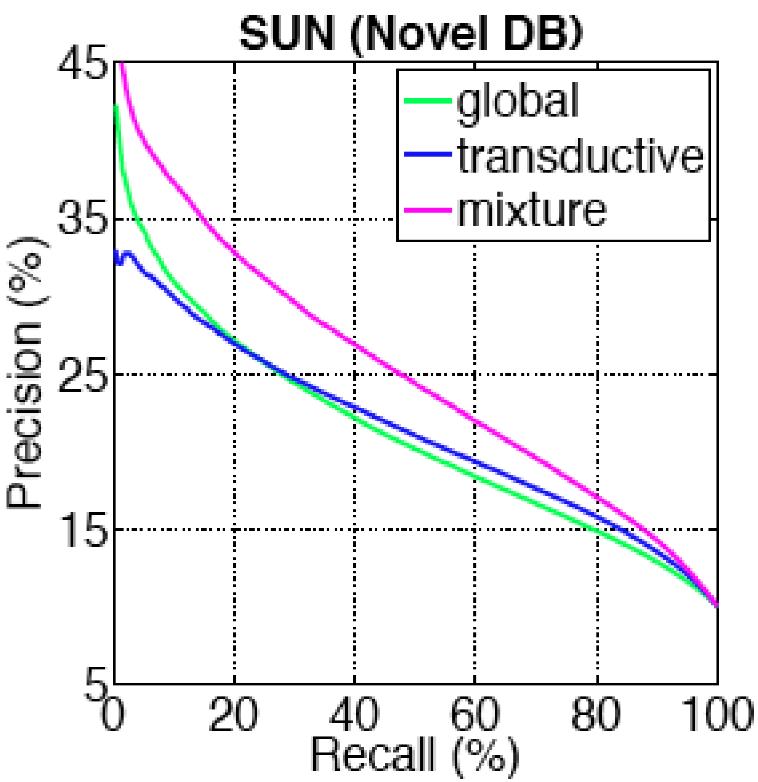
Transductive SVM:
For each test query, it learns a relevance function using ranking contraints from k nearest neighbors.

Mixture: A mixture ranking model, the number of hidden states is chosen using validation queries.

We report precision and recall curves for predicting the 100 most relevant items for each test query.

Results

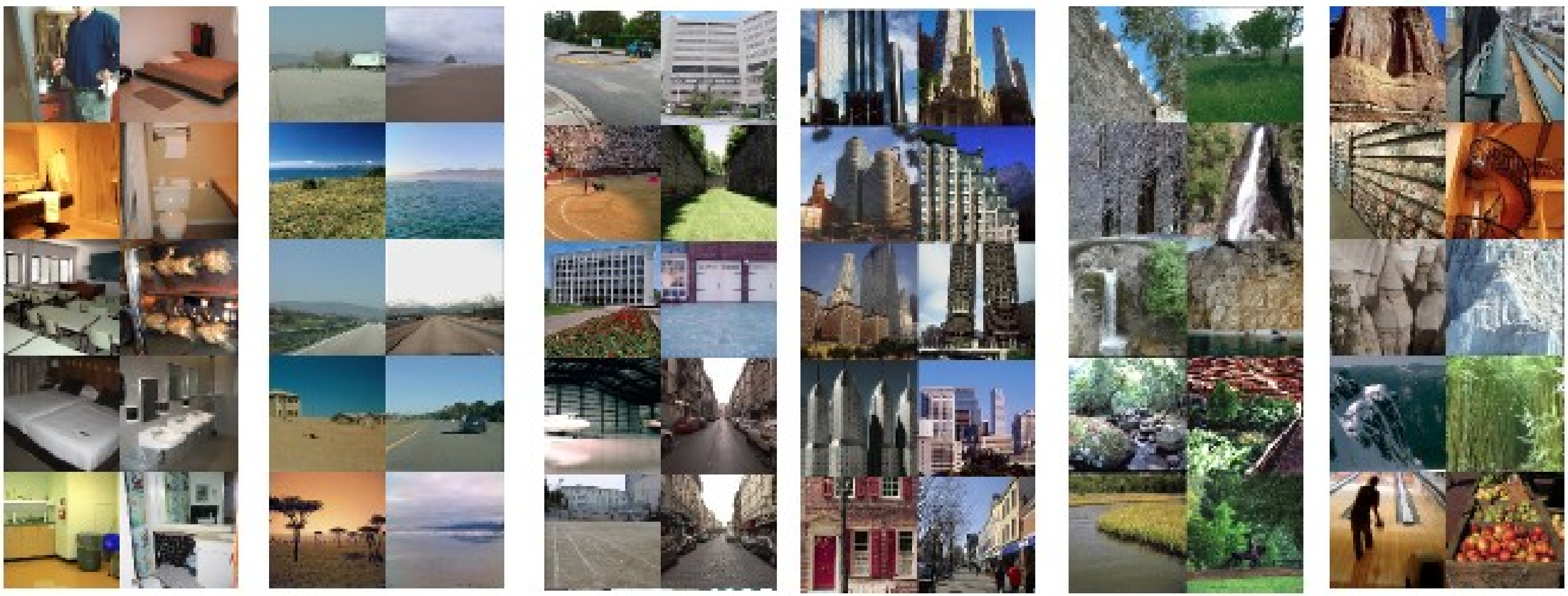




To get 20 relevant images, 220 images need to be browsed with the mixture model vs 286 with other models.

Results

Latent Classes



Summary

- Hidden variables can be useful for a variety of problems involving complex data.

- Spectral learning methods are a good tool for inducing hidden structure.

Future Directions

- Apply the spectral method to large-scale computer vision tasks.

- Spectral methods for unsupervised learning over structured data