

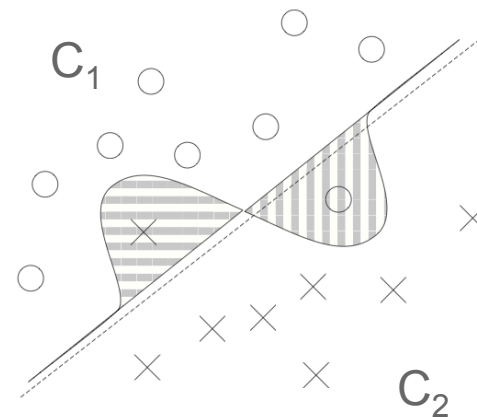
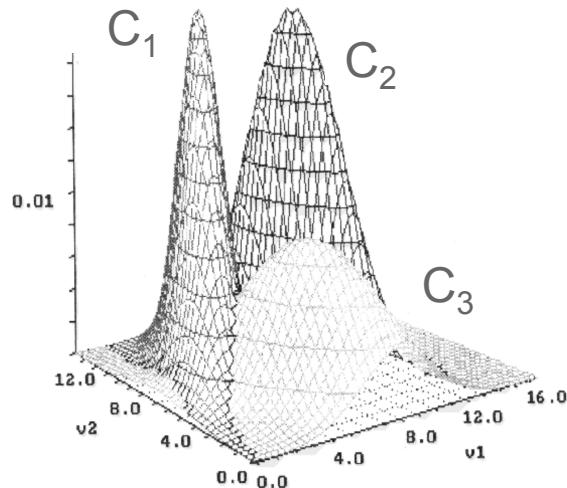
Pattern Recognition

Lecture 8 : String Matching II

Dr. Andreas Fischer
andreas.fischer@unifr.ch

Learning-Based String Matching

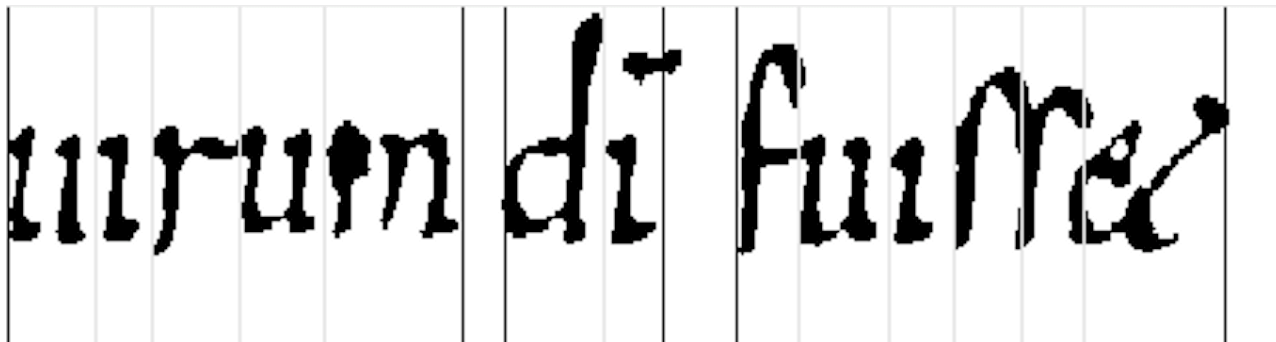
- Template matching
 - Compare two strings, e.g. string edit distance (SED).
- Learning-based matching
 - Generative approach: model probability density function for each element of the string, e.g. hidden Markov models (HMM).
 - Discriminative approach: model posterior probability for each element of the string, e.g. recurrent neural networks (RNN) with long short-term memory cells (LSTM).



Hidden Markov Models

Hidden Markov Models (HMM)

- Origins in speech recognition. Nowadays HMM are used in almost all areas of pattern recognition.
L. Rabiner. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proc. IEEE 77(2):257-285, 1989.
- Typical application domains:
 - Sequential patterns with different lengths.
 - Complex patterns that are composed of simpler parts. For example cursively written words that are composed of characters.
 - A fundamental advantage of HMM is that they do not require an explicit segmentation. Instead, recognition and segmentation are optimized at the same time.



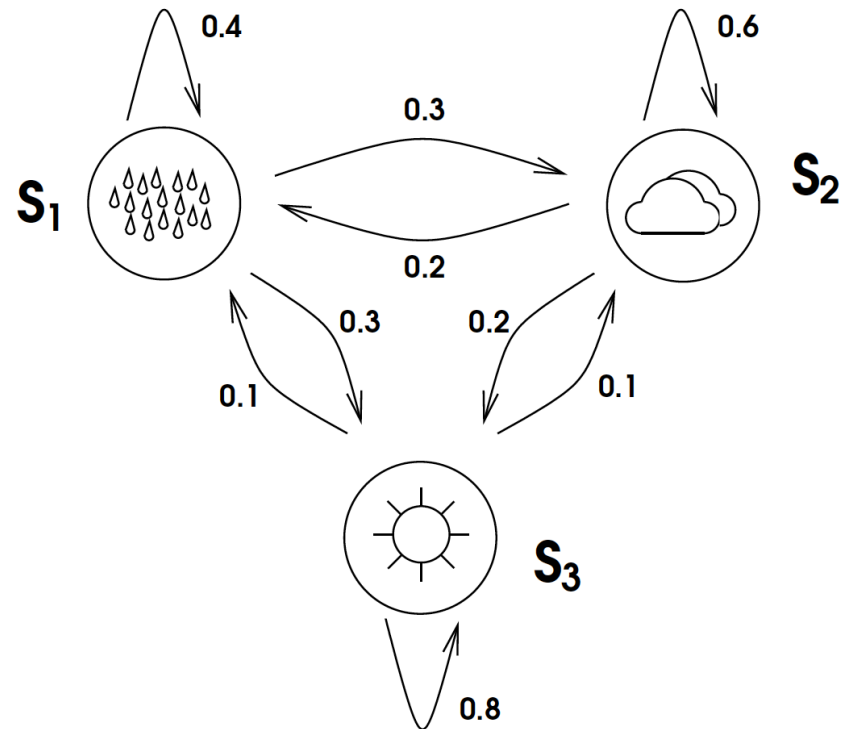
Markov Chains

- HMM are a generalization of *homogeneous Markov chains*:
 - Stochastic process $\{q_t \mid t=1,2,\dots\}$ over a set of states $Q=\{S_1,\dots,S_N\}$
 - $P(q_t=S_j \mid q_{t-1}=S_i, q_{t-2}=S_k, q_{t-3}=\dots) = P(q_t=S_j \mid q_{t-1}=S_i) = a_{ij}$
- Properties:
 - *stationary*: independent of absolute time t
 - *causal*: state probability depends only on previous states
 - *simple*: state probability depends only on one previous state (Markov property)
- State transition probabilities matrix $A = [a_{ij}]_{N \times N}$
 - $a_{ij} \geq 0 \quad \forall i,j$
 - $\sum_j a_{ij} = 1 \quad \forall i$
- If initial state is not defined, we consider initial state probabilities
 - $\Pi = (\pi_1, \dots, \pi_N)$
- The parameters (A, Π) completely define the stochastic behavior of the homogeneous Markov chain.

Example: Weather

- $Q = \{S_1, S_2, S_3\}$ with $S_1 = \text{rainy}$, $S_2 = \text{cloudy}$, $S_3 = \text{sunny}$
- What is the probability that the weather is “sunny, sunny, rainy, rainy, sunny, cloudy, sunny” after a sunny day?
 - Observation $O = S_3 S_3 S_3 S_1 S_1 S_3 S_2 S_3$ and $p(q_1=S_3) = \pi_3 = 1$
 - Probability $P(O \mid A, \Pi) = \pi_3 a_{33} a_{33} a_{31} a_{11} a_{13} a_{32} a_{23} = 1.536 \cdot 10^{-4}$

$$\mathbf{A} = [a_{ij}] = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

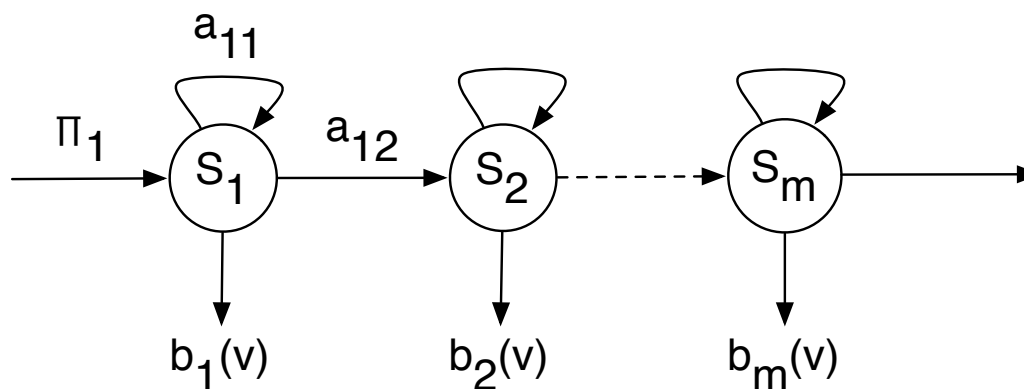


From Markov Chains to HMM

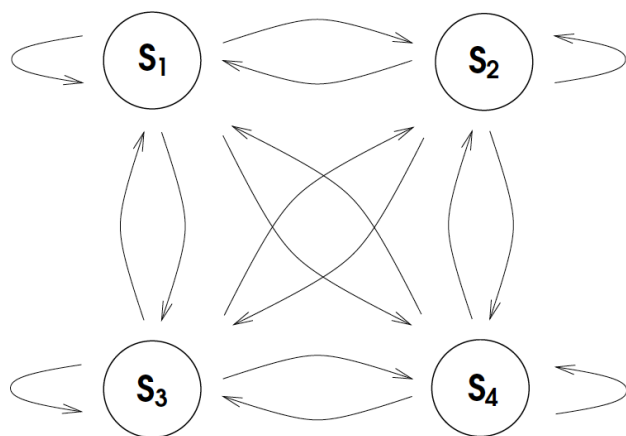
- At each time t not only the state is changed but, additionally, an output is emitted in form of a character from the alphabet $V=\{v_1, \dots, v_K\}$.
- The observation is a string:
 - $O = O_1 \dots O_T, O_t \in V$
 - The emission of character O_t only depends on the state q_t :
 $P(O_t \mid O_1 \dots O_{t-1}, q_1 \dots q_t) = P(O_t \mid q_t)$
 - The states that led to this observation are unknown, *hidden* (thus the name hidden Markov models).
- Emission probability matrix $B = [b_{jk}]_{N \times K}$
 - $b_{jk} = b_j(v_k) = P(O_t=v_k \mid q_t=S_j) \geq 0 \quad \forall j,k$
 - $\sum_k b_{jk} = 1 \quad \forall j$
- The parameters $\lambda=(A,B,\Pi)$ completely define the HMM.

String Generation

- In summary, the observable string is generated as follows:
 1. Set $t = 1$. Select $q_1 = S_i$ with respect to Π .
 2. Emit output $O_t = v_k$ with respect to $b_i(v_k)$.
 3. If $t < T$ change to state $q_{t+1} = S_j$ with respect to a_{ij} . Otherwise, stop the process.
 4. Set $t = t+1$. Go to step 2.
- There are also HMM variants that emit an output during state change. It can be shown that these variants are equivalent.

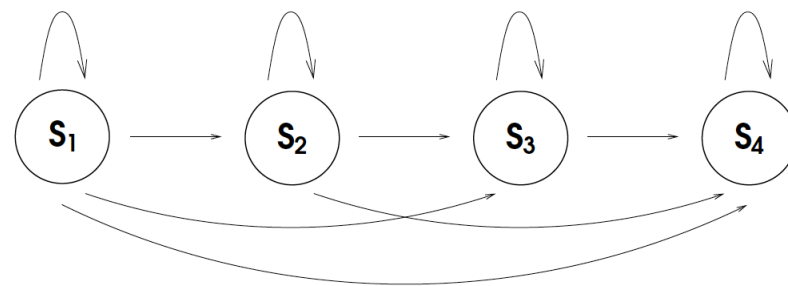


HMM Topologies



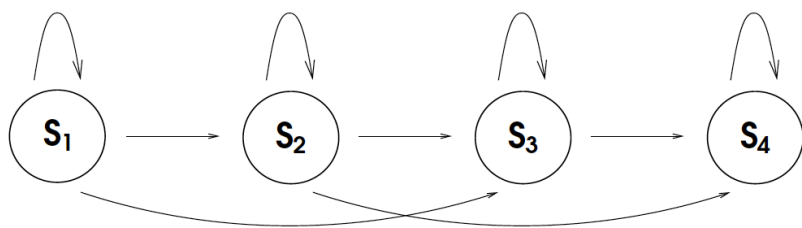
(a)

ergodic (fully connected)



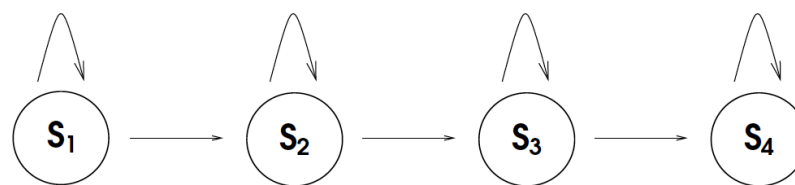
(b)

left-right



(c)

Bakis



(d)

linear

Continuous HMM

- So far, *discrete HMM* have been considered with a finite alphabet V .
- *Continuous HMM* emit feature vectors $\mathbf{x} \in \mathbb{R}^n$. The probability density function $b_j(\mathbf{x})$ is typically modeled with a mixture of Gaussians:

$$b_j(\mathbf{x}) = \sum_{k=1}^K c_{jk} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk})$$

- c_{jk} are the weights of the normal distributions with $\sum_k c_{jk} = 1 \quad \forall j$
- $\boldsymbol{\mu}_{jk}$ is the mean vector of state S_j and mixture k
- $\boldsymbol{\Sigma}_{jk}$ is the covariance matrix of state S_j and mixture k
- *Semi-continuous HMM* have a shared set of Gaussian mixtures over all states. Only the weights are state-specific:

$$b_j(\mathbf{x}) = \sum_{k=1}^K c_{jk} \mathcal{N}(\mathbf{x} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

String Matching with HMM

- Three classic algorithms:

1. Given the observation $O = O_1 \dots O_T$ and the model $\lambda = (A, B, \Pi)$. What is the probability $P(O \mid \lambda)$ that O was generated by λ ?

The solution of this problem allows us to select the model λ_i from a set of models $\{\lambda_1, \dots, \lambda_L\}$ that maximizes the probability $P(O \mid \lambda_i)$.

2. Given the observation $O = O_1 \dots O_T$ and the model $\lambda = (A, B, \Pi)$. What is the optimal sequence of hidden states $q_1 \dots q_T$ for explaining O ?

The solution of this problem allows us to perform a segmentation of a complex pattern into components based on the recognition result.

3. Parameter learning. How can the parameters $\lambda = (A, B, \Pi)$ be optimized based on learning samples?

The solution of this problem is a prerequisite for solving problems 1 and 2.

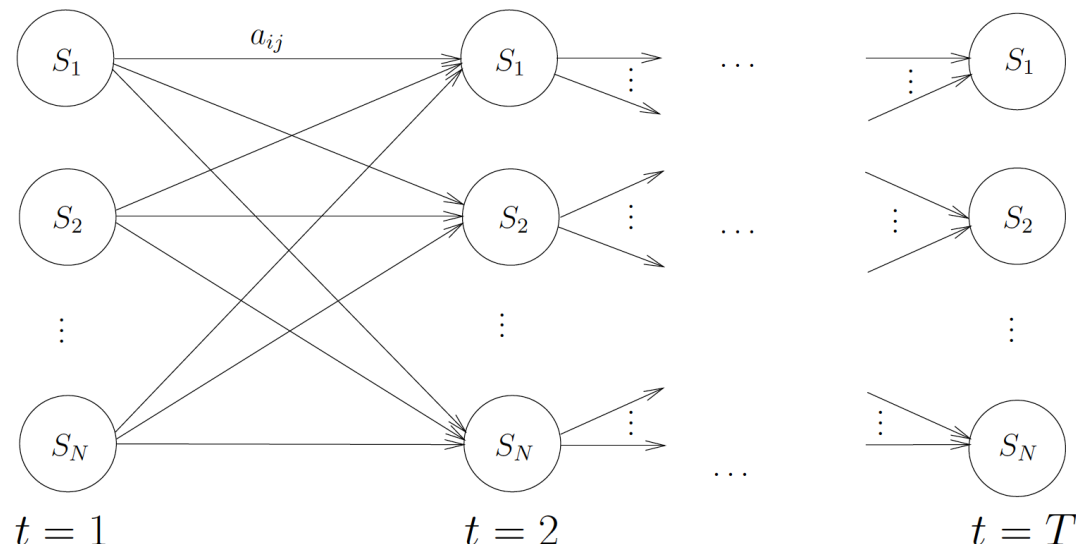
Problem 1

- Naive solution: sum up the probabilities over all possible state sequences.

$$P(O | \lambda) = \sum_{\text{all } q} P(O | q, \lambda) = \sum_{\text{all } q} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \dots b_{q_T}(O_T)$$

However, the number of possible state sequences is exponential $O(N^T)$.

- Fortunately, the problem can be solved efficiently similar to the computation of string edit distance.



Forward-Backward Algorithm

- Forward variable $\alpha_t(i) = p(O_1 \dots O_t, q_t = S_i \mid \lambda)$.
- Recursive definition:

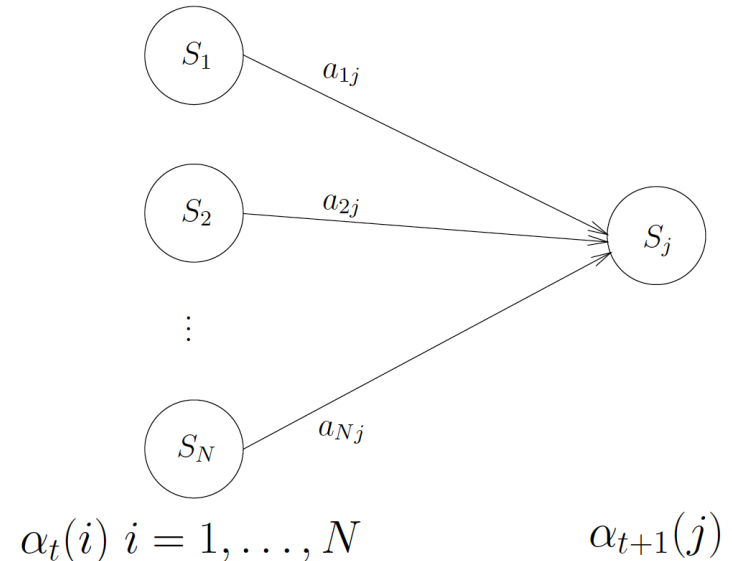
$$\alpha_1(i) = \pi_i b_i(O_1) \quad \forall 1 \leq i \leq N$$

$$\alpha_{t+1}(j) = \sum_{i=1}^N [\alpha_t(i) a_{ij}] b_j(O_{t+1}) \quad \forall 1 \leq j \leq N$$

- Finally, we compute in $O(N^2T)$:

$$P(O \mid \lambda) = \sum_{i=1}^N \alpha_T(i)$$

- Can also be applied backwards (needed for solving problem 3).



Problems 2 and 3

- Problem 2
 - Compute the maximum instead of the sum in the forward algorithm to obtain the best sequence of states, that is the sequence $q=q_1 \dots q_T$ that maximizes $P(O, q \mid \lambda)$.
 - This dynamic programming method is also known as the *Viterbi algorithm*.
- Problem 3
 - The standard approach for solving this problem is the *Baum-Welch algorithm*, which is an expectation maximization (EM) method.
 - Based on the forward-backward algorithm, it iteratively optimizes the model parameters $\lambda=(A, B, \Pi)$ with respect to learning samples.
 - Similar to the backpropagation algorithm for MLP, this procedure only finds a local optimum.

Example: HMM-Based Keyword Spotting

Keyword Spotting

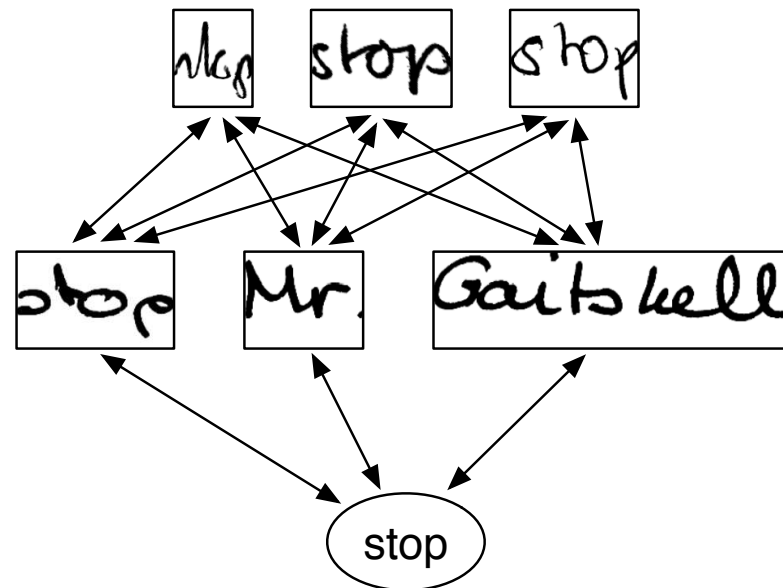
- High error rate and slow computational speed for transcribing general handwriting images (multiple writers, natural language).
- Keyword spotting less demanding for document indexing and retrieval.

In the **first** place it is not a great **deal**
In ~~Gethsemane~~ He prayed that the day
Was a the necessity being dissolved - She really did feel tired
One of the greatest steps forward that has been
As it is, with so much of our **life** already

Letters, orders and Instructions. **October 1755-277.**
shed to-morrow, and to be very particular in their Accounts
of what they receive. They will also receive **Arms** so
soon as they arrive from Fort Cumberland, to complete their
Accounts. They are to see that each man distinguishes
his Firelock by some particular mark, which the Sub-

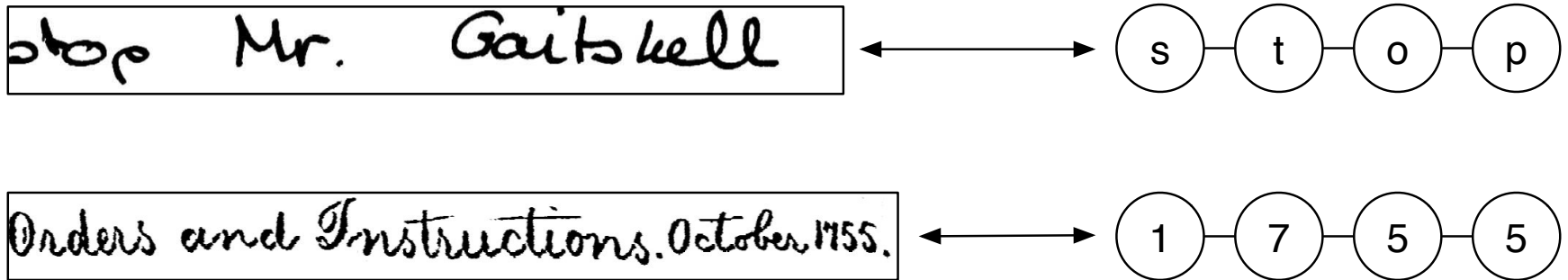
Matching Complete Words

- Matching word images (*template-based*) or models (*learning-based*).
- Limitations:
 - Keyword sample images
 - Text line segmentation

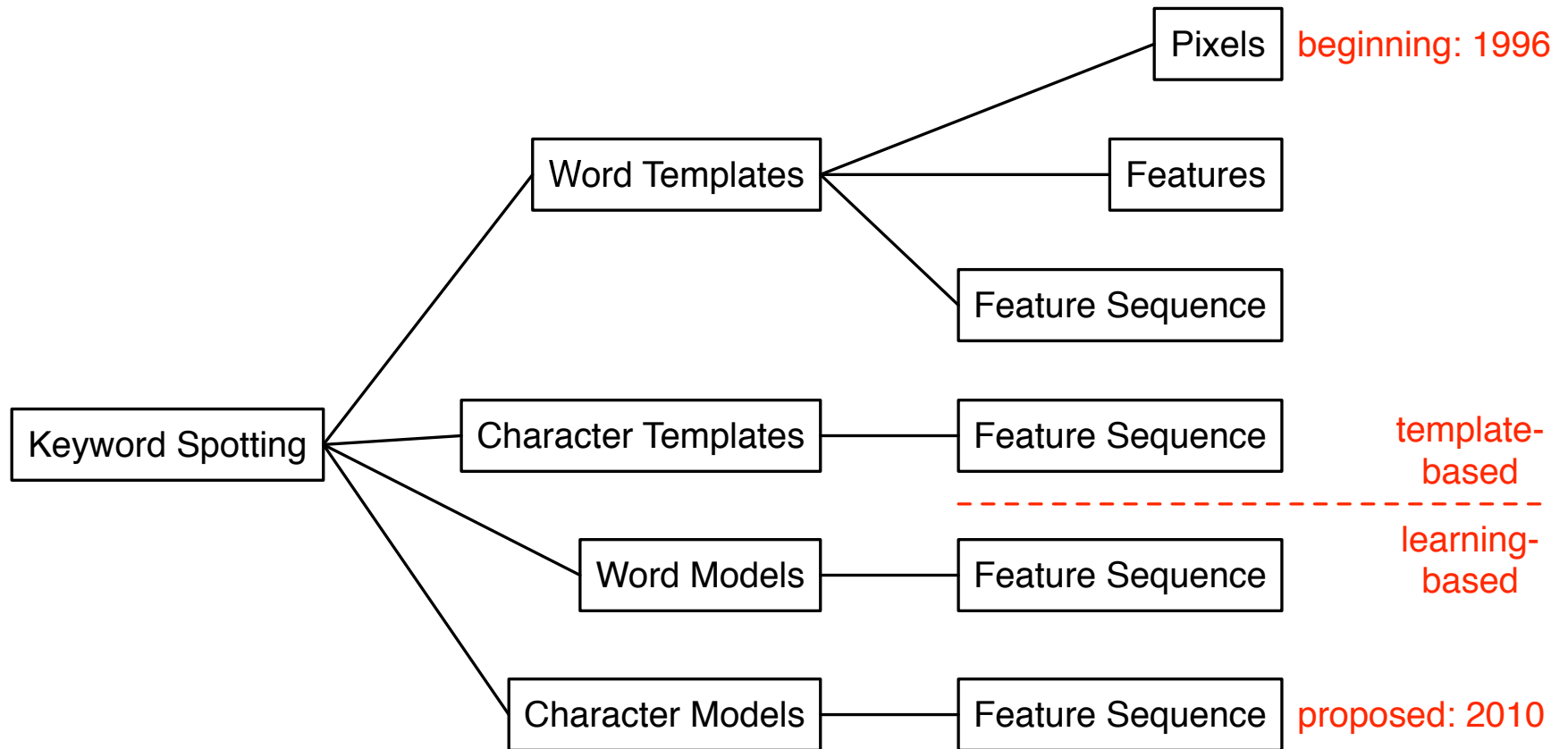


Matching HMM Character Models

- Inspired by the application of character HMMs for transcription.
- Advantages:
 - + Small number of character classes
 - + Arbitrary keywords
 - + No text line segmentation



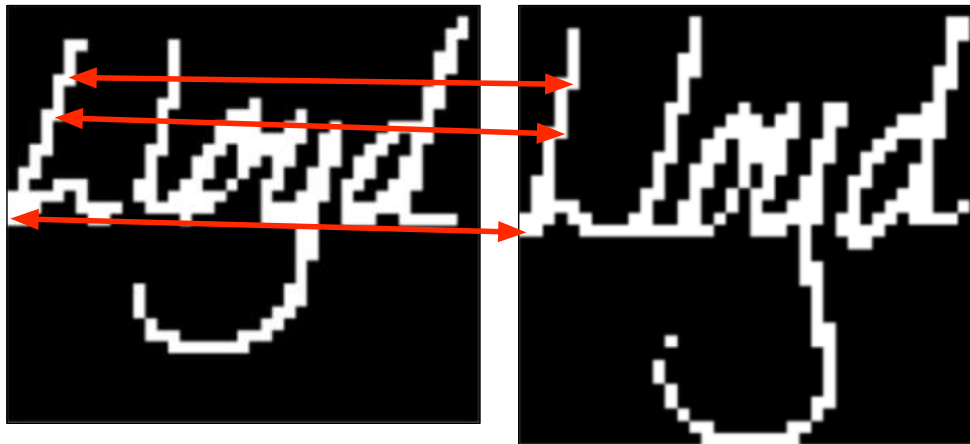
Related Work



Word Template: Pixels

R. Manmatha, C. Han, E. Riseman, *Word Spotting: A New Approach to Indexing Handwriting*, in Proc. Int. Conf. on Computer Vision and Pattern Recognition, pp. 631–637, 1996.

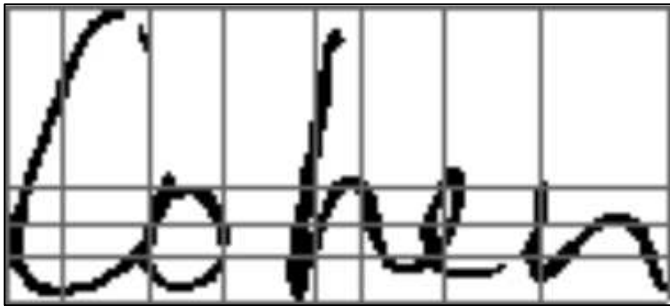
- Pixel assignment with the Scott and Longuet-Higgins algorithm, singular value decomposition of $D_{ij} = \exp(-\frac{\|I_i - J_j\|^2}{2\sigma^2})$.
- Optimization of affine transform $E_{SLH} = \sum_i (I_i - AJ_i - b)^2$.



Word Template: Features

B. Zhang, S. N. Srihari, C. Huang, *Word Image Retrieval Using Binary Features*, in Proc. Int. Conf. on Document Recognition and Retrieval, pp. 45–53, 2004.

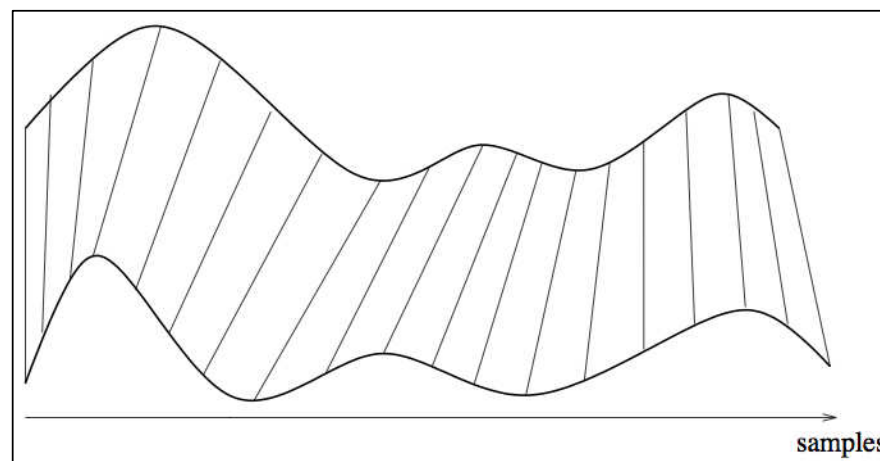
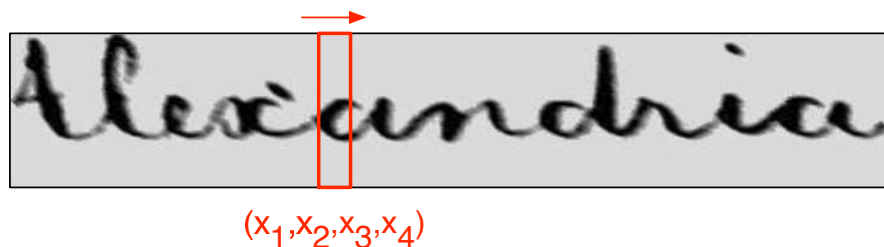
- Zoning with equi-mass splits into 4×8 regions.
- Matching 1024 binary gradient, structural and convexity features (GSC).



Word Template: Feature Sequence

T. M. Rath, R. Manmatha, *Word Image Matching Using Dynamic Time Warping*, in Proc. Int. Conf. on Computer Vision and Pattern Recognition, pp. 521–527, 2003.

- Sliding window feature extraction.
- Alignment costs by means of dynamic programming.



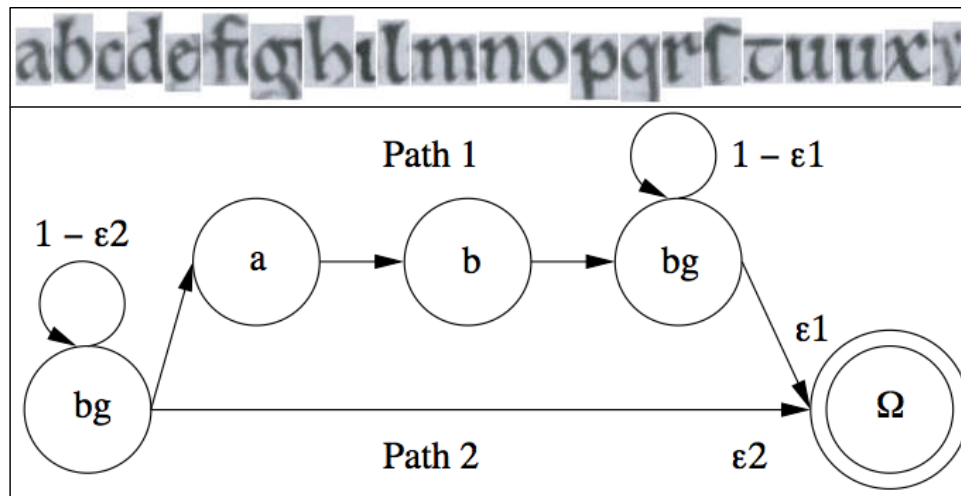
Milestone: Dynamic Time Warping

- T. M. Rath, R. Manmatha, *Word spotting for historical documents*, Int. Journal on Document Analysis and Recognition 9, pp. 139–152, 2007.
- T. Adamek, N. E. Connor, A. F. Smeaton, *Word Matching Using Single Closed Contours for Indexing Historical Documents*, Int. Journal on Document Analysis and Recognition 9, pp. 153–165, 2007.
- J. Rodriguez, F. Perronnin, *Local Gradient Histogram Features for Word Spotting in Unconstrained Handwritten Documents*, in Proc. Int. Conf. on Frontiers in Handwriting Recognition, pp. 7–12, 2008.
- K. Terasawa, Y. Tanaka, *Slit Style HOG Features for Document Image Word Spotting*, in Proc. Int. Conf. on Document Analysis and Recognition, pp. 116–120, 2009.

Character Template: Feature Sequence

J. Edwards, Y. W. Teh, D. Forsyth, R. Bock, M. Maire, G. Vesom, *Making Latin Manuscripts Searchable Using gHMM's*, in Proc. Int. Conf. on Advances in Neural Information Processing Systems, pp. 385–392, 2004.

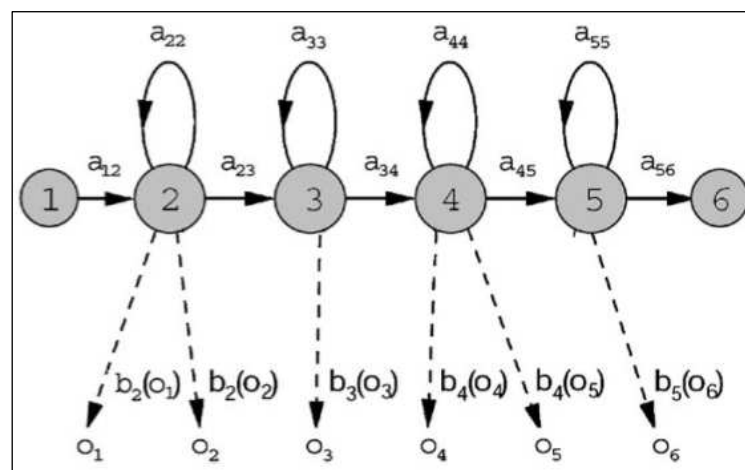
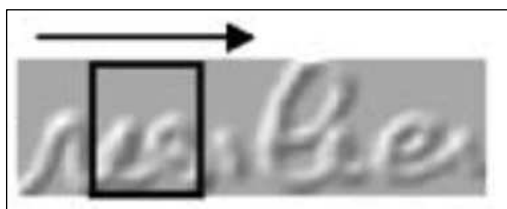
- Gaussian emission model with respect to character templates.
- Viterbi recognition with character unigrams, comparison with general a-z background model, text line retrieval.



Word Model: Feature Sequence

J. Rodriguez, F. Perronnin, *Handwritten Word-Spotting Using Hidden Markov Models and Universal Vocabularies*, Pattern Recognition 42 (9), pp. 2106–2116, 2009.

- Sliding window gradient features.
- HMM word models, Baum-Welch training, Viterbi recognition, posterior spotting confidence $\frac{p(X|W)}{p(X)}$, approximate $p(X)$ with GMM.



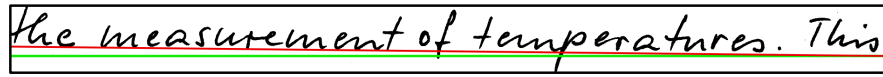
Character Model: Feature Sequence

- A. Fischer, A. Keller, V. Frinken, H. Bunke, *HMM-based word spotting in handwritten documents using subword models*, in Proc. Int. Conf. on Pattern Recognition, pp. 3416–3419, 2010.
 - A. Fischer, A. Keller, V. Frinken, H. Bunke, *Lexicon-free handwritten word spotting using character HMMs*, in Pattern Recognition Letters 33 (7), pp. 934–942, 2012.
 - V. Frinken, A. Fischer, R. Manmatha, H. Bunke, *A novel word spotting method based on recurrent neural networks*, in IEEE Trans. PAMI 34 (2), pp. 211–224, 2012.
- + Learning-based: copes with multiple writing styles
 - + Character-based: spots arbitrary keywords
 - + Line-based: needs no line segmentation for training and recognition
 - + Lexicon-free: achieves high computational speed

Image Preprocessing

- Binary images are normalized to cope with different writing styles.

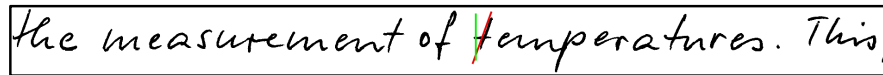
Skew



the measurement of temperatures. This,



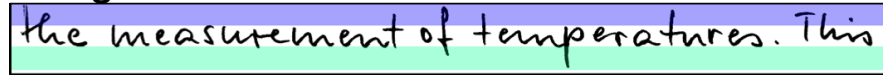
Slant



the measurement of temperatures. This,



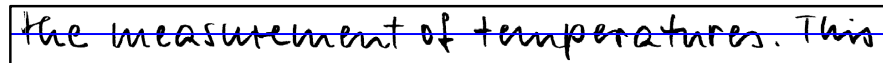
Height



the measurement of temperatures. This,



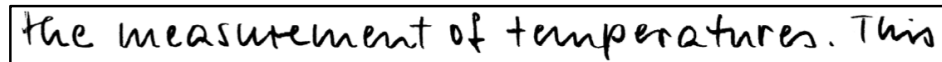
Width



the measurement of temperatures. This,



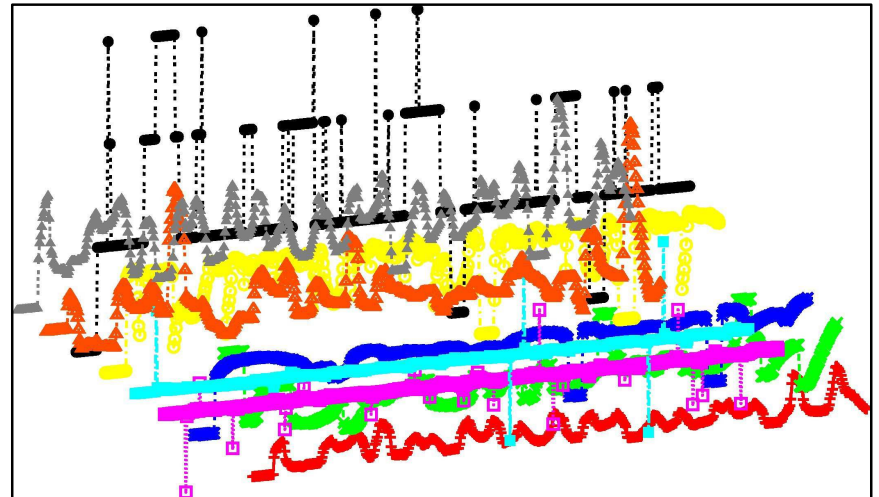
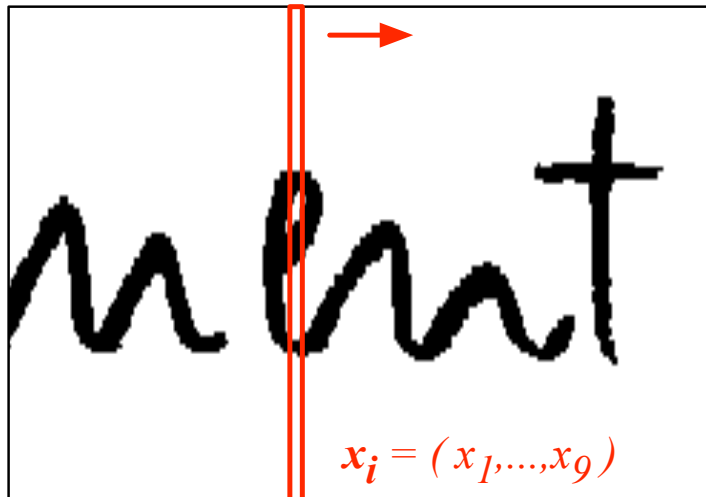
Result



the measurement of temperatures. This,

Feature Sequence Extraction

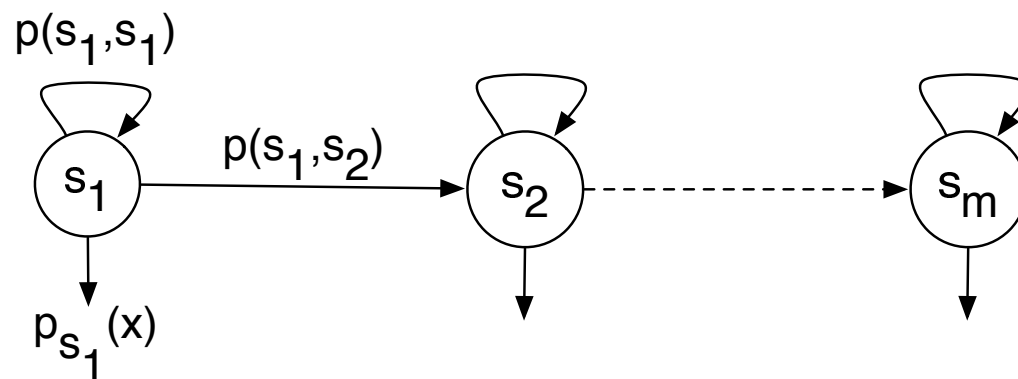
- Nine geometric features are extracted by a sliding window.
- Local descriptor includes contour positions, contour deviations, black-white transitions, foreground fractions, and moments.



Character Models

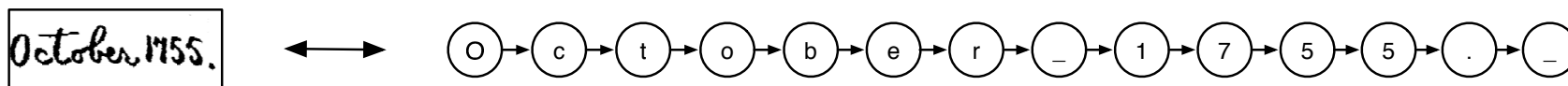
- Several hidden states per character, linear topology.
- Transition probabilities $p(s_i, s_i), p(s_i, s_{i+1})$.
- States emit observable feature vectors with a mixture of Gaussians.

$$p_{s_i}(x) = \sum_{j=1}^G w_{ij} \mathcal{N}(x \mid \mu_{ij}, \Sigma_{ij})$$



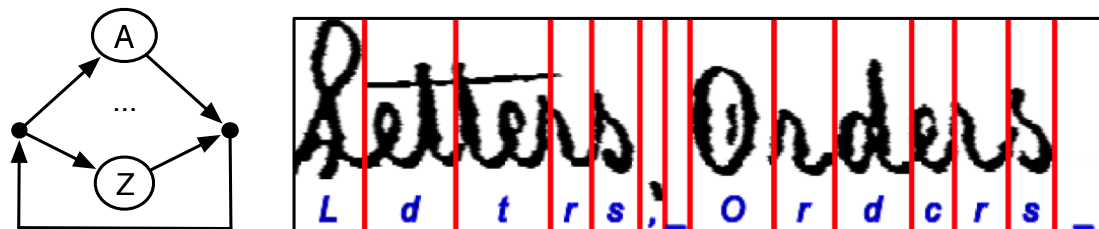
Text Line Models

- Baum-Welch training with transcribed text lines.



- Viterbi recognition of vector sequence $\mathbf{x} = x_1, \dots, x_N$ with character sequence $\mathbf{c} = c_1, \dots, c_{|\mathbf{c}|}$ based on a text line model M .

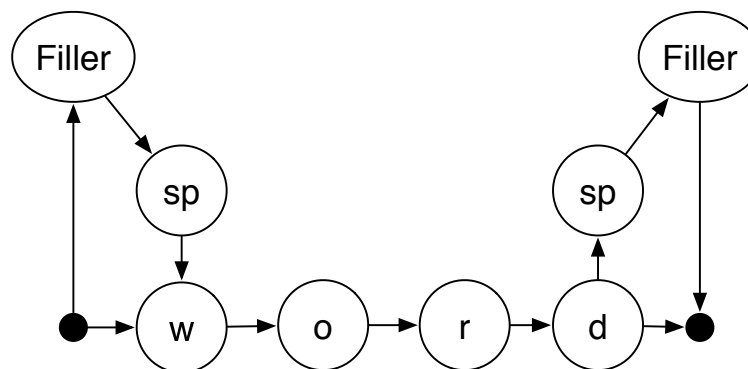
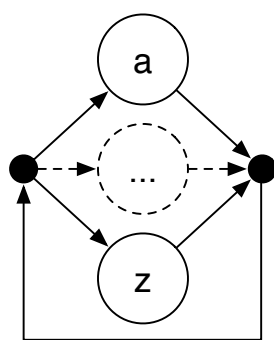
$$\log p(\mathbf{x}|M) = \max_{\mathbf{c} \in \mathcal{C}_M} (\log p(\mathbf{x}|\mathbf{c}))$$



Spotting Confidence

- Log likelihood difference between two text line models.
- The filler model F is an arbitrary sequence of characters.
- The keyword model K is constrained to contain the keyword w .

$$c(\mathbf{x}, w) = \frac{\log p(\mathbf{x}|K) - \log p(\mathbf{x}|F)}{|w|_K} \leq 0$$



Justifications

- Posterior approximation:

$$p(w|\mathbf{x}) = \frac{p(\mathbf{x}|w)p(w)}{p(\mathbf{x})} = \frac{p(\mathbf{x}|w)p(w)}{\sum_w p(\mathbf{x}|w)p(w)} \approx \frac{p(\mathbf{x}|\mathbf{c}_K)p(\mathbf{c}_K)}{p(\mathbf{x}|\mathbf{c}_F)p(\mathbf{c}_F)}$$

- Neyman-Pearson test:

$$\frac{p(\mathbf{x}|H_0)}{p(\mathbf{x}|H_1)} = \frac{p(\mathbf{x}|K)}{p(\mathbf{x}|F)}$$

Computational Speed

- Speed lexicon-based transcription \propto lexicon size²
Typically 20'000 – 100'000 word classes.
- Speed lexicon-free spotting \propto alphabet size²
Typically less than 100 character classes.
- Example: 140'000 pages of the George Washington collection
 - 1 min / line \rightarrow several years for an automatic transcription
 - 1 ms / line \rightarrow one hour for spotting a keyword

Data Sets

<http://www.iam.unibe.ch/fki/databases/>

Database	Writers	Train	Valid	Test	Char.	Keywords
IAM	657	6161	920	929	81	882
GW	2	328	164	164	83	105
PAR	3	2236	911	1328	96	1217

A MOVE to stop Mr. Gaitskell from
nominating any more Labour life Peers
is to be made at a meeting of Labour
MPs tomorrow. Mr. Michael Foot has
put down a resolution on the subject

IAM

270. *Letters, Orders and Instructions. October 1755.*
only for the publick use, unless by particu-
lar Orders from me. You are to send
down a Barrel of Flint with the Arms, to
Winchester, and about two thousand weight
of Flour, for the two companies of Rangers;
twelve hundred of which to be delivered
Captain. Arkley and Company, at the

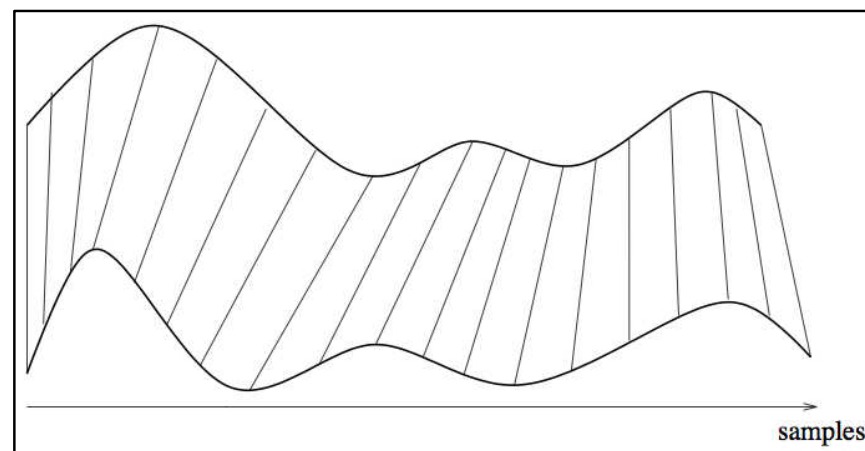
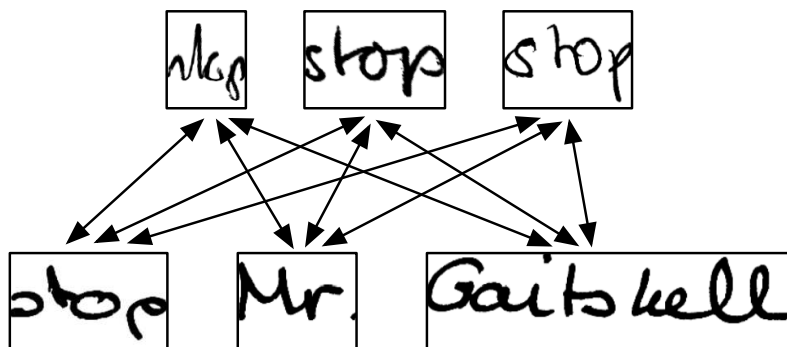
George Washington

ob ich der quater nide vergess
z machet trvrich mir den lip.
dax also meringv hazet wip.
ir stimme sint geliche bel.
genüge sint gein valsehe siel.
etliche valsehes lere.
svs teilent sich div mare.
dax die geliche sint genannt.

Parzival

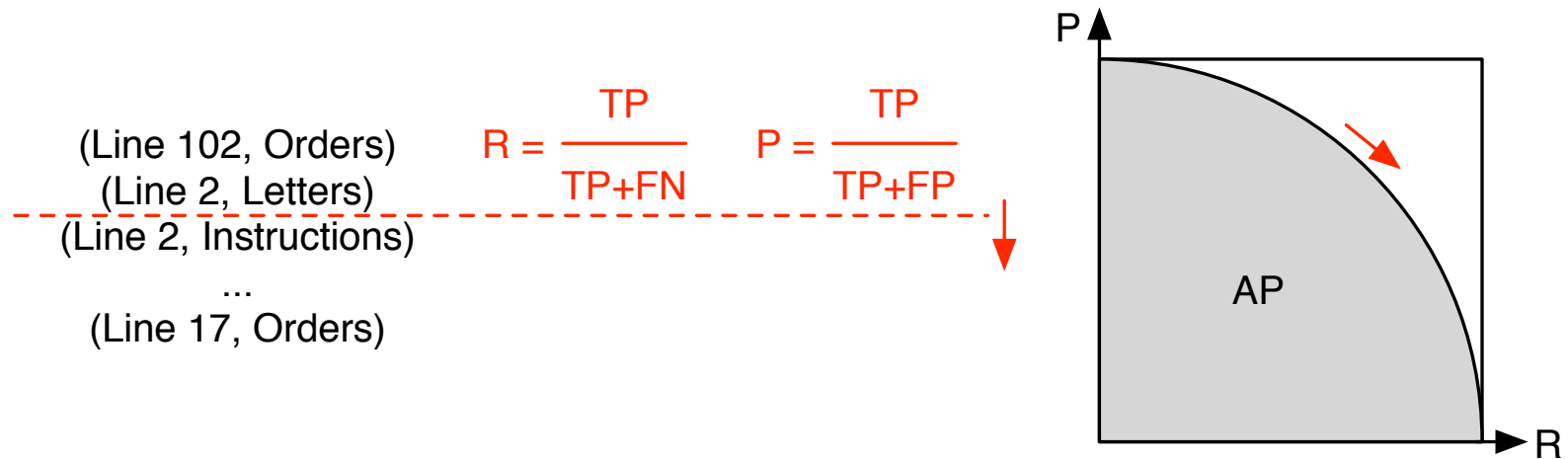
Reference System

- Word template matching with dynamic time warping (DTW).
- Based on the same feature vector sequence.
- Based on manually corrected word segmentation.

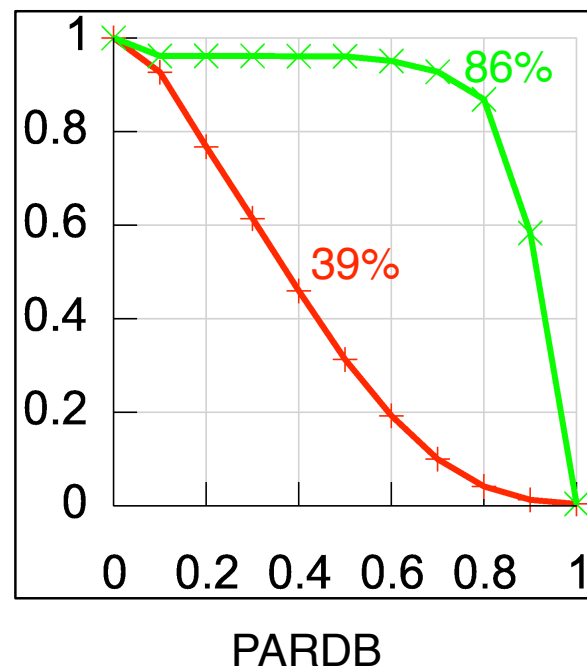
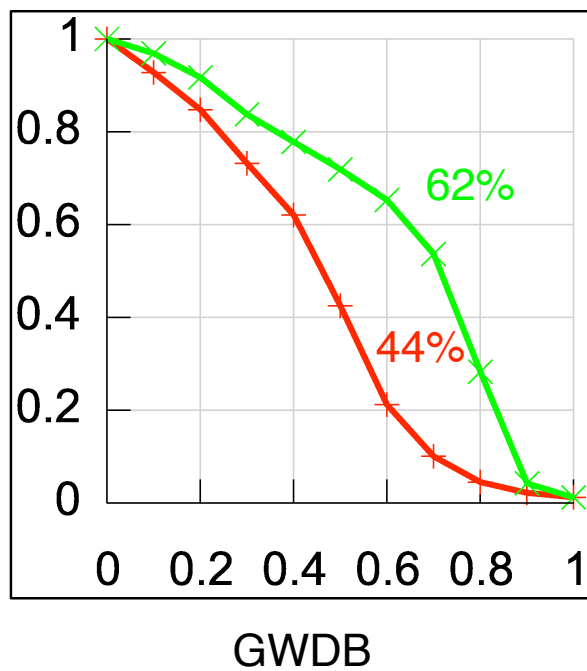
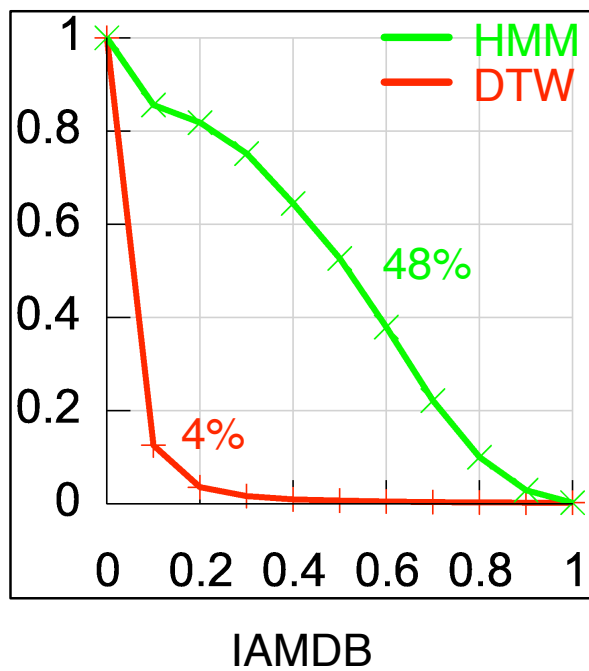


Performance Evaluation

- Text line retrieval, useful for document indexing and retrieval.
- Return top- N pairs (\mathbf{x}, w) sorted by spotting confidence $c(\mathbf{x}, w)$.
- Performance measure is the average precision (AP), i.e. the area under the recall-precision curve.



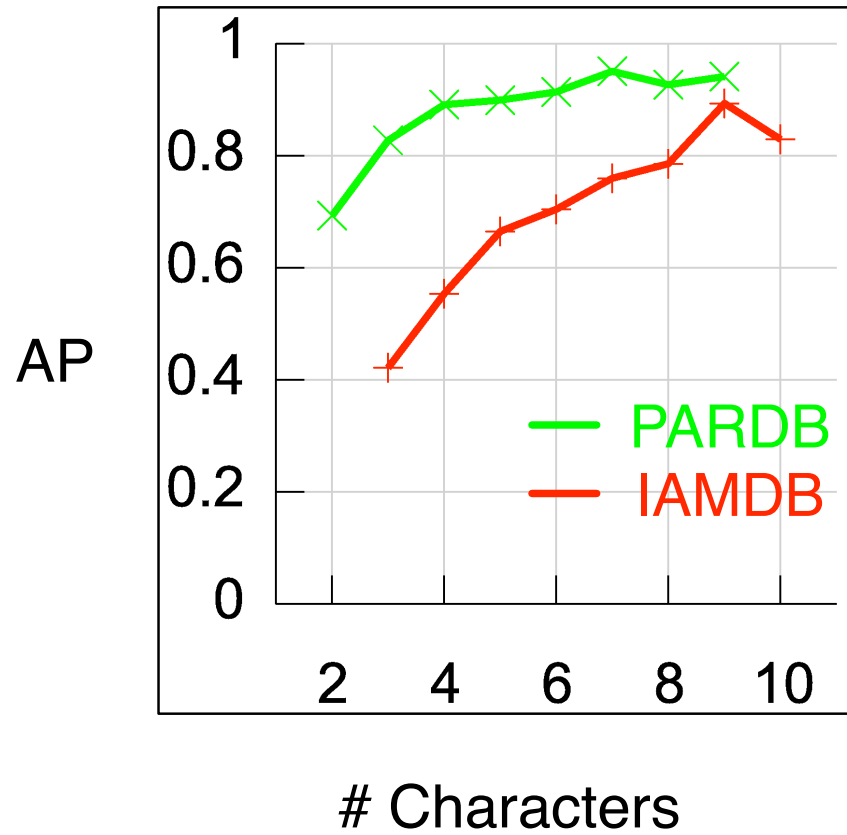
Results



Parameters and Software

- Character HMM:
 - Number of states per character $m \in \{3, 4, \dots, 30\}$.
 - Number of Gaussian mixtures $G \in \{1, 2, \dots, 30\}$.
 - Baum-Welch training, Viterbi recognition: HTK toolkit
<http://htk.eng.cam.ac.uk/>
- Performance evaluation:
 - Ranking and AP: `trec_eval` software
http://trec.nist.gov/trec_eval/

Keyword Size



Error Samples

- Keywords can be wrongly recognized by the filler model.
- In such a case the text line is returned in the top ranks with maximum confidence $c(\mathbf{x}, w) = 0$ since $p(\mathbf{x}|K) = p(\mathbf{x}|F)$.

and the B orders of Carolina: I am confident,

must be deducted next payment: as also

stores and medicines arrives, you are to embrace

from the Regiments into which they were draught

Questions

And up to ten years' imprisonment can be imposed on anyone convicted of sabotage. These stern measures had the desired effect today at Kumasi where the strikers gave in, but Tekoradi, the chief storm centre, they still holding out despite the presence of 1400 police and 16 armoured cars. And how did the Government react when the strikers demonstrated in Accra?