Universität Stuttgart
IPVS – Institute for Parallel and Distributed Systems

Analytic Computing

Advanced Topics in Machine Learning
5 Representation: Markov Networks

Prof. Dr. Steffen Staab
Dr. Rafika Boutalbi
Zihao Wang
https://www.ipvs.uni-stuttgart.de/departments/ac/

Learning Objectives

Undirected Graphical Models / Markov (random) networks

- General Gibbs distribution
- Conditional Random Fields

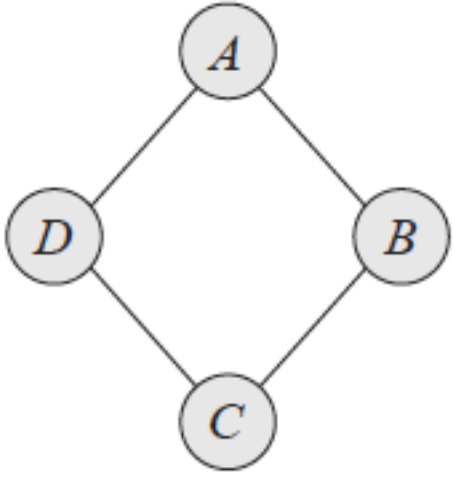
Disclaimer

Figures and examples not marked otherwise are taken from the book by Koller & Friedman

1 Undirected Graphical Models

A Modeling Problem for Bayesian Network

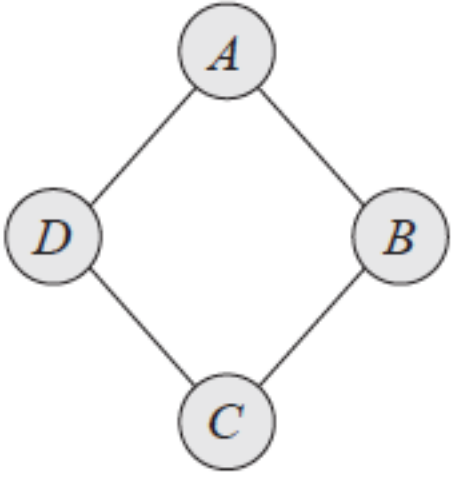
- Professor was ambiguous, two interpretations: $0,1$
- Four students Alice, Bob, Charlie, Debbie collaborating in pairs, but never Alice-Charlie, Bob-Debbie
- Target: $(B \perp D \mid A, C),\ (A \perp C \mid B, D)$
- Question:
  How to model as Bayesian Network?

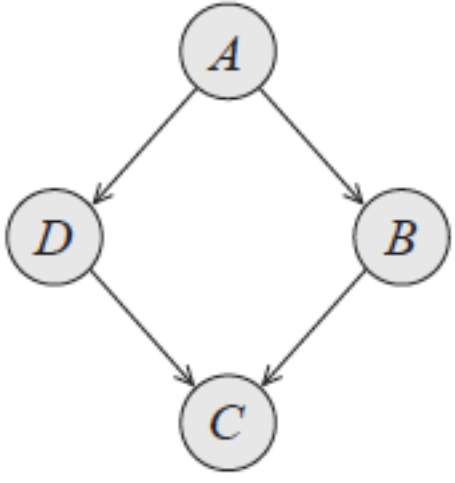


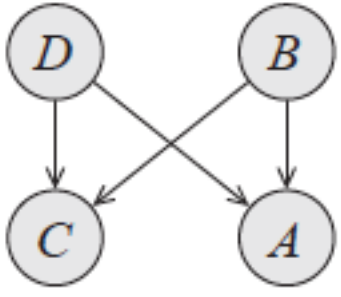
A Modeling Problem for Bayesian Network

- $(B \perp D \mid A, C)$ and $(A \perp C \mid B, D)$

  cannot be achieved using Bayesian Network







Undirected Graphical Model

- Random variables: $\mathcal{X} = \{X_1, \ldots, X_k\}$
- Assignment to $\mathcal{X}$: $x_{\mathcal{X}}$, or simply x
- Collection of subsets of $\mathcal{X}$: $\mathcal{A} = \{A_1, \ldots, A_l \,|\, \forall i\colon A_i \subset \mathcal{X}\}$
- Assignment to $A \in \mathcal{A}$: x_A
- Factors: $\Phi = \{\phi_{A_1}, \ldots, \phi_{A_l}\}$

Definition: An undirected graphical model is a probability distribution that can be written using factors ϕ_A with corresponding scopes A as

$$P(x) = \frac{1}{Z_\Phi} \prod_{A \in \mathcal{A}} \phi_A(x_A)$$

with normalization factor Z

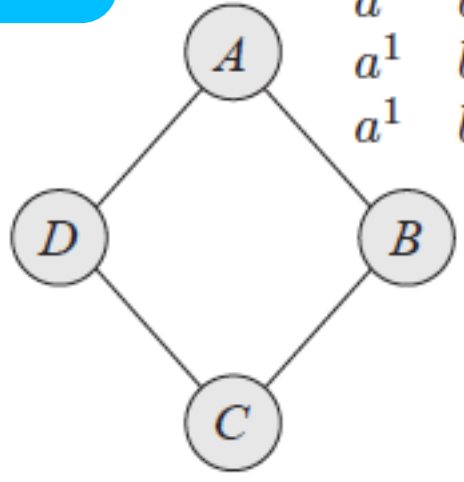
$$Z_\Phi = \sum_{x \in \mathcal{X}} \prod_{A \in \mathcal{A}} \phi(x_A)$$

Simple Example: Pairwise Markov Network

Factors have pairs of variables in their scope

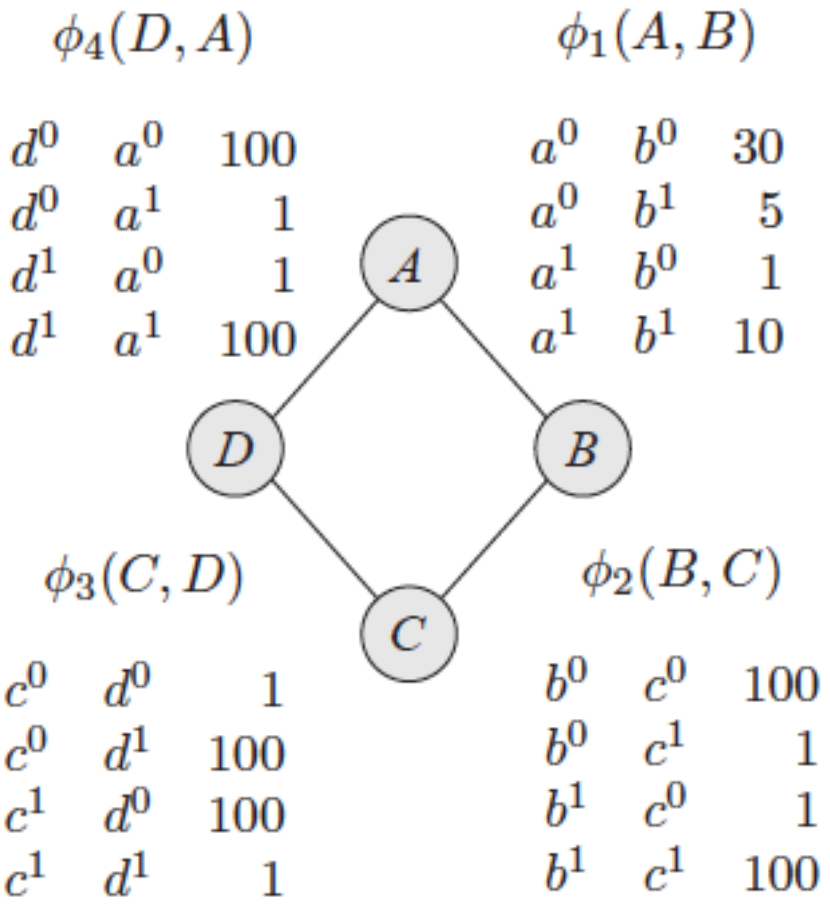
$\phi_1(A, B)$

a^0	b^0	30
a^0	b^1	5
a^1	b^0	1
a^1	b^1	10



No direction
No conditional probability tables

Simple Example: Pairwise Markov Network



affinity
compatibility
soft constraints

Simple Example: Pairwise Markov Network

$$\tilde{P}(A, B, C, D) = \phi_1(A, B)\phi_2(B, C)\phi_3(C, D)\phi_4(D, A)$$

$$P(A, B, C, D) = \frac{1}{Z}\tilde{P}(A, B, C, D)$$

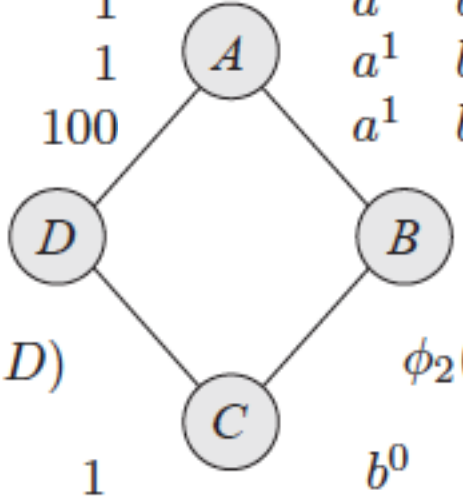
Assignment				Unnormalized	Normalized
a^0	b^0	c^0	d^0	300,000	0.04
a^0	b^0	c^0	d^1	300,000	0.04
a^0	b^0	c^1	d^0	300,000	0.04
a^0	b^0	c^1	d^1	30	$4.1 \cdot 10^{-6}$
a^0	b^1	c^0	d^0	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^0	d^1	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^1	d^0	5,000,000	0.69
a^0	b^1	c^1	d^1	500	$6.9 \cdot 10^{-5}$
a^1	b^0	c^0	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^0	d^1	1,000,000	0.14
a^1	b^0	c^1	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^1	d^1	100	$1.4 \cdot 10^{-5}$
a^1	b^1	c^0	d^0	10	$1.4 \cdot 10^{-6}$
a^1	b^1	c^0	d^1	100,000	0.014
a^1	b^1	c^1	d^0	100,000	0.014
a^1	b^1	c^1	d^1	100,000	0.014

$\phi_4(D, A)$

d^0	a^0	100
d^0	a^1	1
d^1	a^0	1
d^1	a^1	100

$\phi_1(A, B)$

a^0	b^0	30
a^0	b^1	5
a^1	b^0	1
a^1	b^1	10



$\phi_3(C, D)$

c^0	d^0	1
c^0	d^1	100
c^1	d^0	100
c^1	d^1	1

$\phi_2(B, C)$

b^0	c^0	100
b^0	c^1	1
b^1	c^0	1
b^1	c^1	100

Probability diverges from individual factors

Assignment				Unnormalized	Normalized
a^0	b^0	c^0	d^0	300,000	0.04
a^0	b^0	c^0	d^1	300,000	0.04
a^0	b^0	c^1	d^0	300,000	0.04
a^0	b^0	c^1	d^1	30	$4.1 \cdot 10^{-6}$
a^0	b^1	c^0	d^0	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^0	d^1	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^1	d^0	5,000,000	0.69
a^0	b^1	c^1	d^1	500	$6.9 \cdot 10^{-5}$
a^1	b^0	c^0	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^0	d^1	1,000,000	0.14
a^1	b^0	c^1	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^1	d^1	100	$1.4 \cdot 10^{-5}$
a^1	b^1	c^0	d^0	10	$1.4 \cdot 10^{-6}$
a^1	b^1	c^0	d^1	100,000	0.014
a^1	b^1	c^1	d^0	100,000	0.014
a^1	b^1	c^1	d^1	100,000	0.014

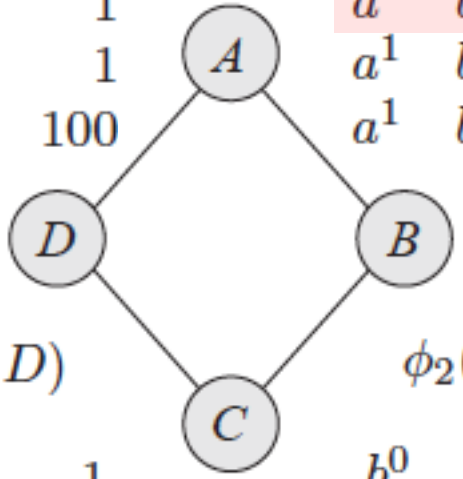
$$P\left(\begin{matrix} A = a^0, \\ B = b^1 \end{matrix}\right) = 0.69$$

$\phi_4(D, A)$

d^0	a^0	100
d^0	a^1	1
d^1	a^0	1
d^1	a^1	100

$\phi_1(A, B)$

a^0	b^0	30
a^0	b^1	5
a^1	b^0	1
a^1	b^1	10



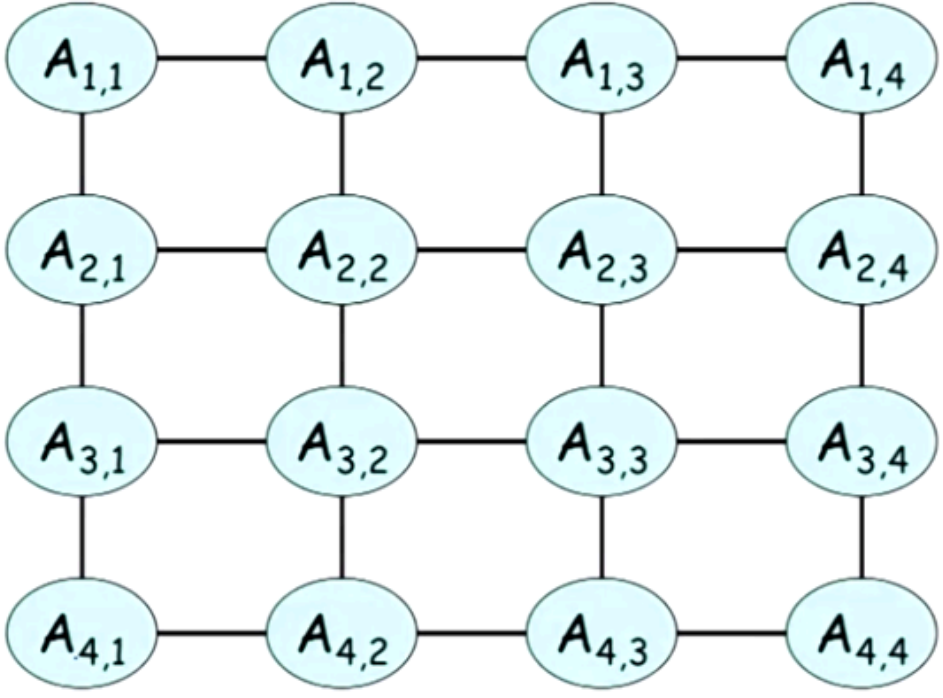
$\phi_3(C, D)$

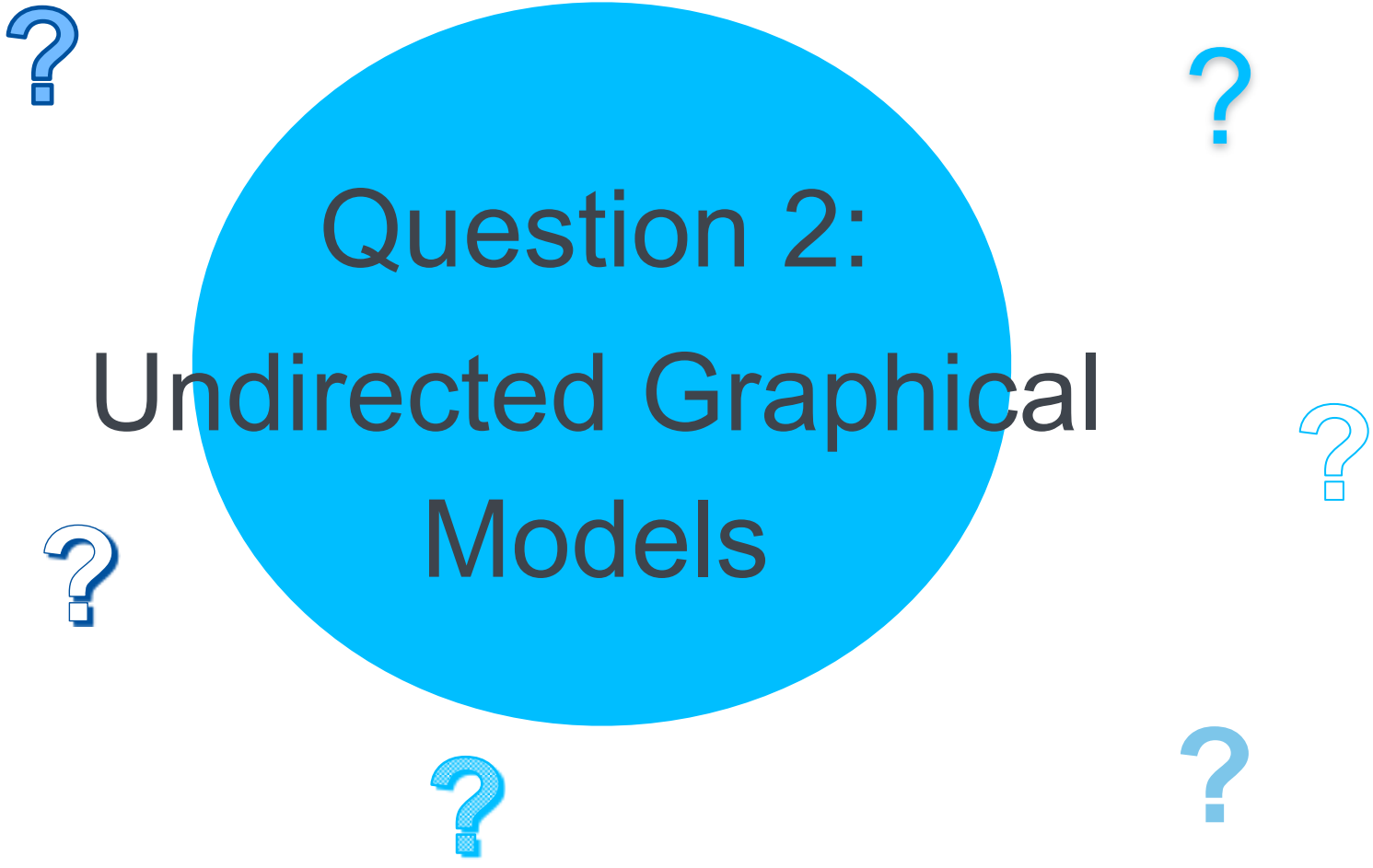
c^0	d^0	1
c^0	d^1	100
c^1	d^0	100
c^1	d^1	1

$\phi_2(B, C)$

b^0	c^0	100
b^0	c^1	1
b^1	c^0	1
b^1	c^1	100

Application for Pairwise Markov Network: Modeling Pixels



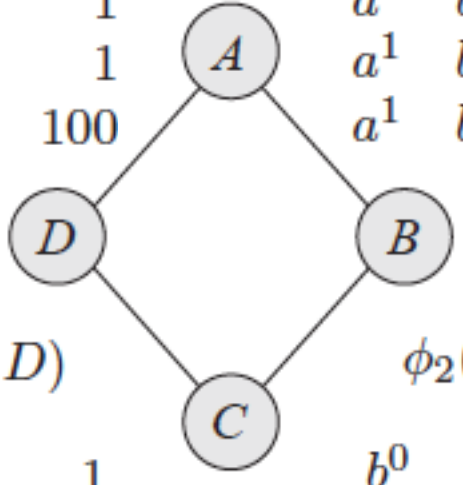

Question 2:
Undirected Graphical Models

$\phi_4(D, A)$

d^0	a^0	100
d^0	a^1	1
d^1	a^0	1
d^1	a^1	100

$\phi_1(A, B)$

a^0	b^0	30
a^0	b^1	5
a^1	b^0	1
a^1	b^1	10



$\phi_3(C, D)$

c^0	d^0	1
c^0	d^1	100
c^1	d^0	100
c^1	d^1	1

$\phi_2(B, C)$

b^0	c^0	100
b^0	c^1	1
b^1	c^0	1
b^1	c^1	100

Consider the pairwise factor $\phi_1(A,B)$. That potential is proportional to:

- ○ The marginal probability P(A,B)
- ○ The conditional probability P(A | B)
- ○ The conditional probability P(A, B | C, D)
- ○ None of the above

2 Gibbs Distribution

Are pairwise Markov Networks fully expressive?

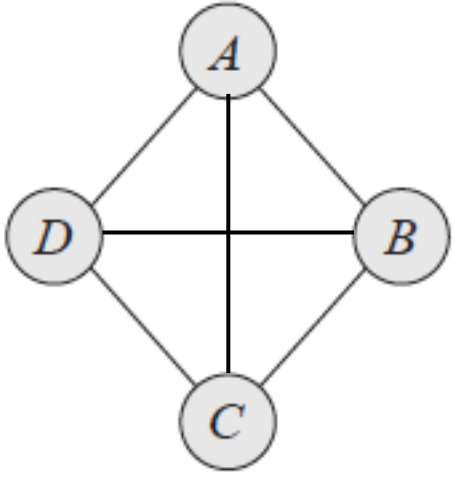
Consider a fully connected pairwise Markov network over $X_1, \ldots X_n$ where each X_i has d values.

How many parameters does the network have?

How many edges: $\frac{n}{2}(n-1)$

How many value combinations on each edge: d^2

$\frac{n}{2}(n-1)d^2 \ll d^n$



Not every distribution can be represented as a pairwise Markov network

Undirected Graphical Model

- Random variables: $\mathcal{X} = \{X_1, \ldots, X_k\}$
- Assignment to $\mathcal{X}$: $x_{\mathcal{X}}$, or simply x
- Collection of subsets of $\mathcal{X}$: $\mathcal{A} = \{A_1, \ldots, A_l \mid \forall i: A_i \subset \mathcal{X}\}$
- Assignment to $A \in \mathcal{A}$: x_A
- Factors: $\Phi = \{\phi_{A_1}, \ldots, \phi_{A_l}\}$

Definition: An undirected graphical model is a probability distribution that can be written using factors ϕ_A with corresponding scopes A as

$$P(x) = \frac{1}{Z_\Phi} \prod_{A \in \mathcal{A}} \phi_A(x_A)$$

with normalization factor Z

$$Z_\Phi = \sum_{x \in \mathcal{X}} \prod_{A \in \mathcal{A}} \phi(x_A)$$

Gibbs Distribution

$$\tilde{P}_\Phi(\mathcal{X}) = \prod_{A \in \mathcal{A}} \phi_A(A)$$

$$Z_\Phi = \sum_{\mathcal{X}} \tilde{P}_\Phi(\mathcal{X})$$

$$P_\Phi(\mathcal{X}) = \frac{1}{Z_\Phi} \tilde{P}_\Phi(\mathcal{X})$$

Induced Markov Network

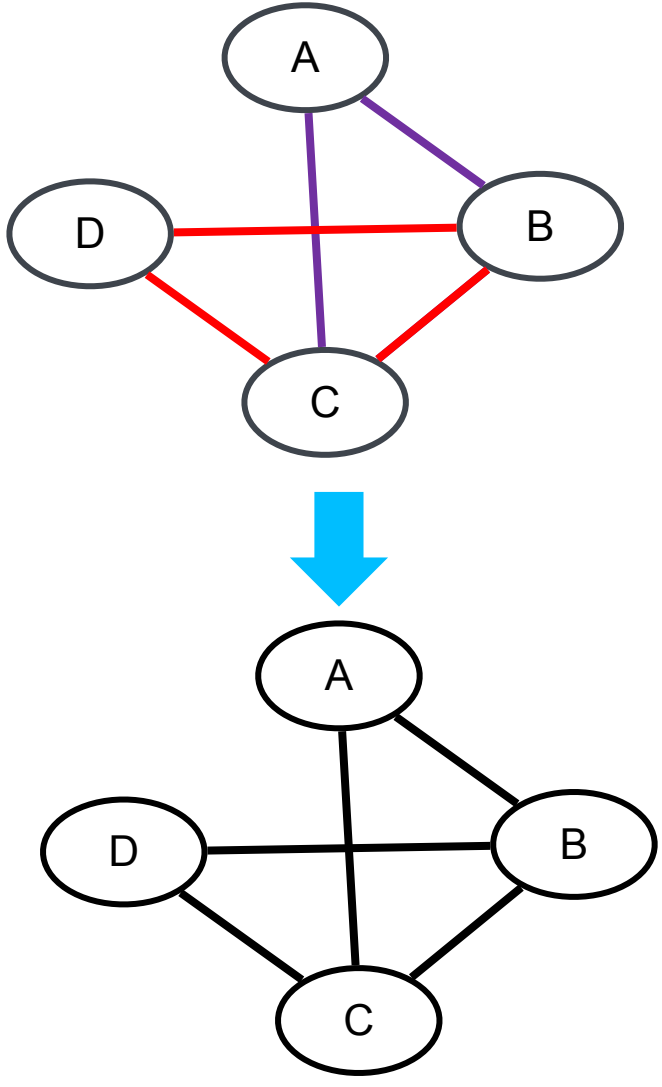
$\phi_1(A, B, C), \phi_2(B, C, D)$

Induced Markov network H_Φ

has an edge $X_i - X_j$

when there exists $\phi \in \Phi$

such that $\{X_i, X_j\} \subseteq \text{scope}(\phi)$



Factorization

$P(\mathcal{X})$ factorizes over H if there exists Φ

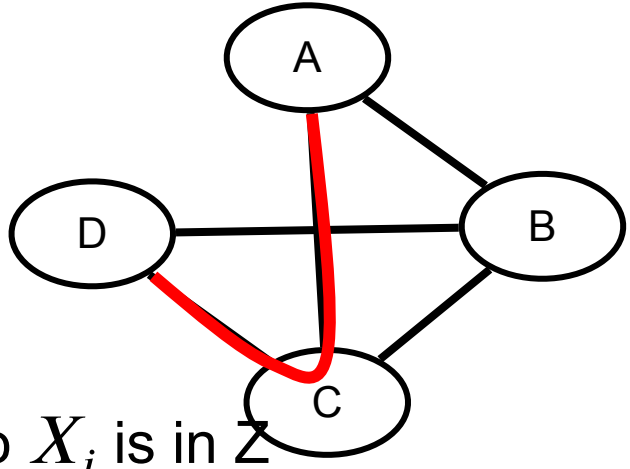
such that

$$P(\mathcal{X}) = P_\Phi(\mathcal{X}) \text{ and } H \text{ is the induced graph for } \Phi$$

Flow of Influence & Active Trails

$\phi_1(A, B, C), \phi_2(B, C, D)$

Influence can flow along any trail regardless of the form of the factors



A trail $X_1 - \ \ldots - X_n$ is active given Z if no X_i is in Z

Examples:

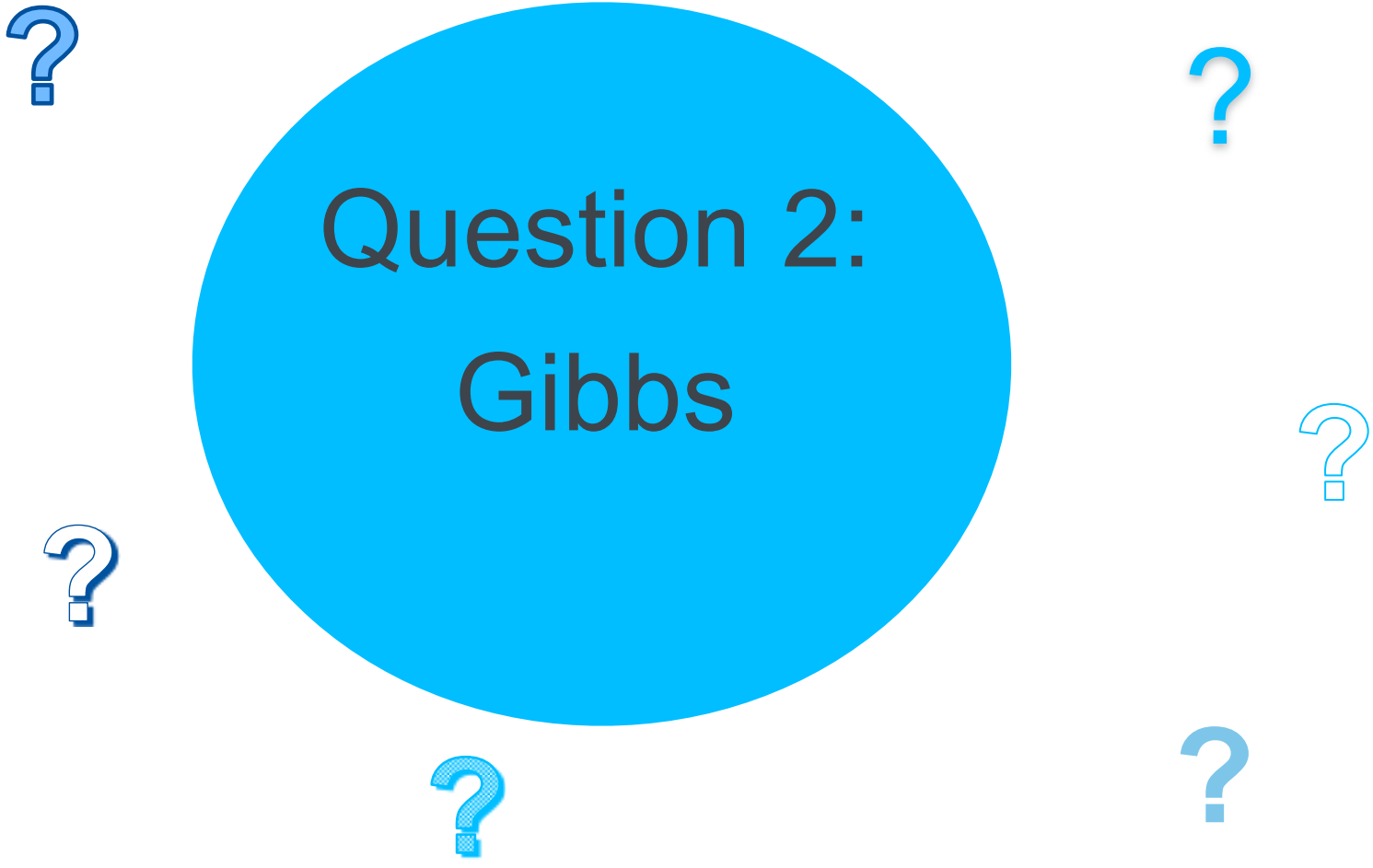
$A - C - D$ is active if $Z = \{\}$

$A - C - D$ is inactive if $Z = \{C\}$

There is no active trail from A to D if $Z = \{B, C\}$

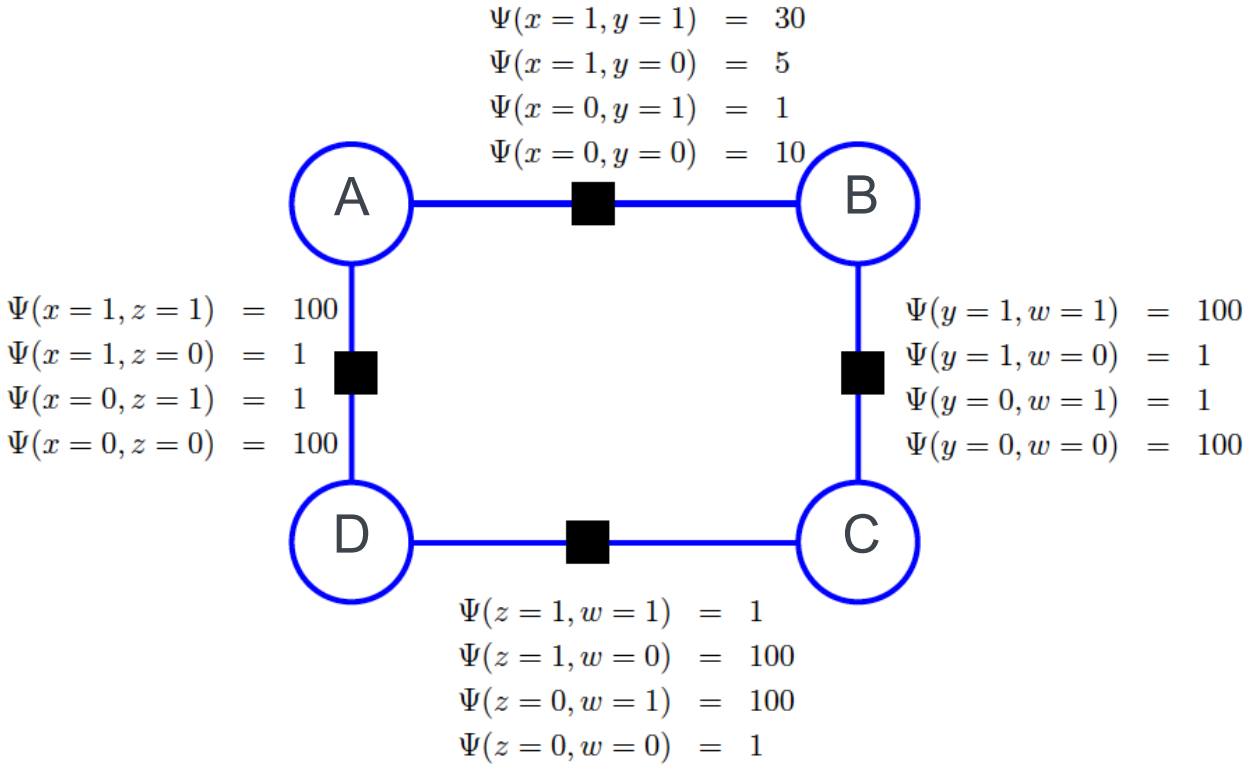
Summary

- Gibbs distribution represents distribution as a product of factors
- Induced Markov network connects every pair of nodes that are in the same factor
- Markov network structure does not fully specify the factorization of $P(\mathcal{X})$
- Active trails depend only on graph structure


Question 2:
Gibbs

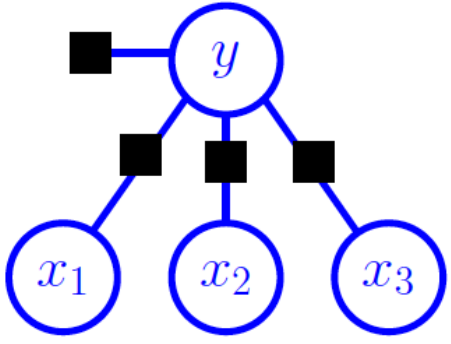
Intermezzo: Factor Graphs

Example factor graph

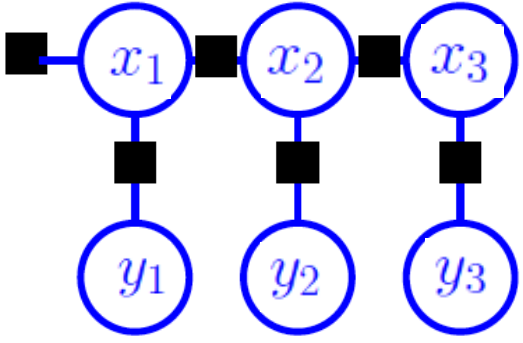


Further example factor graphs

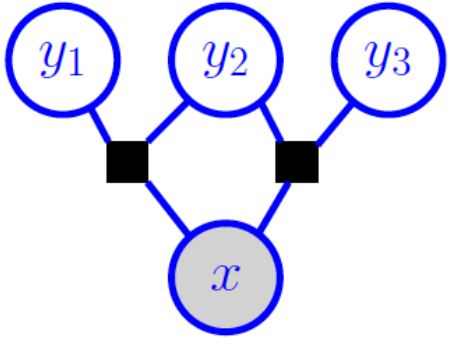
Naïve Bayes



Hidden Markov Model

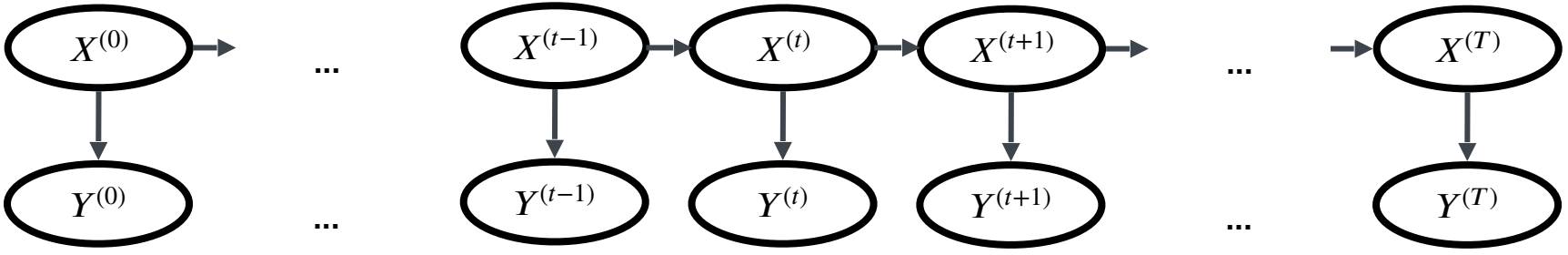


Linear-Chain
Conditional Random Field



3 Conditional Random Field (CRF)

Problem 1: Re-consider Hidden Markov Model



Example application:

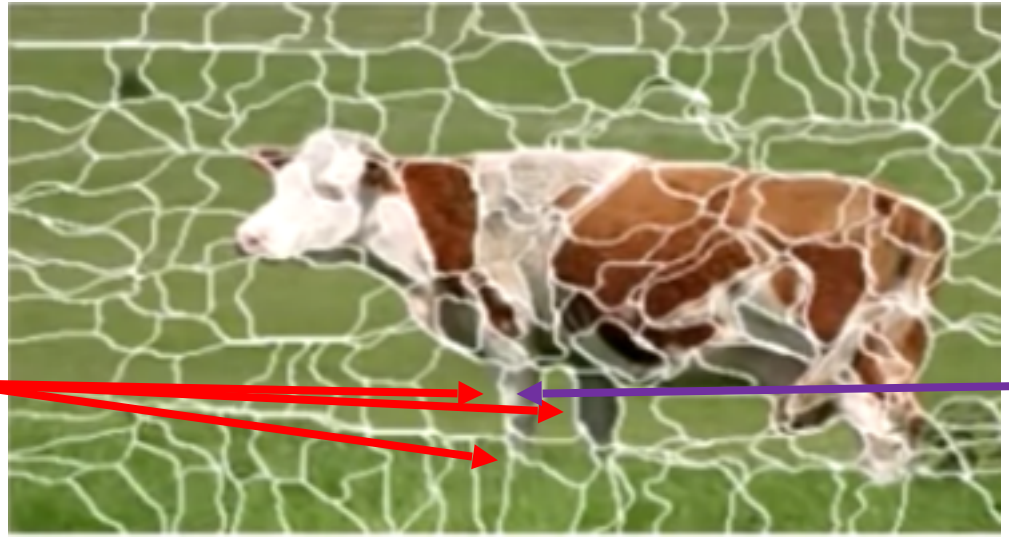
- „The horse chased by the barn fell.“

active verb ~~simple past~~ past participle

Long range dependency

- If a word is unknown („Emthe“), its identity is not a useful feature, neighboring words are needed, context is needed

Problem 2: Correlated Features



- Naive Models:
  - redundant features *swamp* decision
  - *label bias:* prior training occludes observations (Sutton & McCallum 2007)
- Model correlations:
  - *too many correlations* to model
    - intractable if too many dependencies

Generative vs. discriminative classification

- Generative approach
  - Assume functional form for $P(Y), P(X \mid Y)$
  - Estimate parameters of $P(Y), P(X \mid Y)$ from training data
  - Use Bayes rule to calculate $P(Y \mid X)$
- Bayesian Networks
  - Naive Bayes
  - HMM
  - Markov random fields
- Model $P(\mathcal{Y}, \mathcal{X})$
  - then query this model

- Discriminative approaches
  - Assume some functional form for $P(Y \mid X)$
  - Estimate parameters of $P(Y \mid X)$ directly from training data
- Approaches
  - Logistic regression
  - Support vector machines
  - (most) neural networks
  - Conditional random fields
- Model $P(\mathcal{Y} \mid \mathcal{X})$
  - then query this model

Ng, Andrew Y., and Michael I. Jordan. "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes." *Advances in neural information processing systems*. 2002.

Core Ideas for Conditional Random Field

1. Distinguish two sets of random variables:
   - $\mathcal{X} = \{X_1, \ldots, X_k\}$ are always observed
   - $\mathcal{Y} = \{Y_1, \ldots, Y_l\}$ are the target variables

2. Model $\boldsymbol{P(\mathcal{Y} \mid \mathcal{X})}$
   - Model factors
     - with scopes $\mathrm{A} \subset (\mathcal{Y} \cup \mathcal{X})$
     - no scope A such that $\mathrm{A} \subseteq \mathcal{X}$
   - Normalize over $\sum_{\mathcal{Y}} \mathrm{P}(\mathcal{Y}, \mathcal{X})$

A conditional model specifies the probabilities of possible **label sequences** given an **observation sequence**.

Different from Gibbs distribution

Core Ideas for Conditional Random Field: DO NOTs

- **Do not** model $\mathrm{P}(\mathcal{Y}, \mathcal{X})$ at all
- **Do not** train by maximizing joint likelihood:

$$\operatorname{argmax}_{\theta} \prod_{x,y \in \text{Data}} \mathrm{P}(x, y \mid \theta)$$

- **Do not** model correlations in $\mathcal{X}$
- **Do not** normalize over $\sum \mathrm{P}(\mathcal{Y}, \mathcal{X})$

Avoid sources of intractability
(as far as possible)

Task specific prediction

	input variables x	target variable y
Image segmentation	pixel values, processed features (SIFT, SURF, CNN)	class for every pixel (cow,...)
Text	words in sentence	labels of words (person, location,...)

Observation Sequence

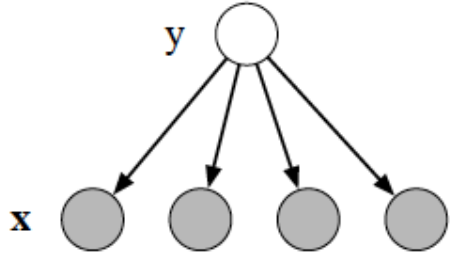
- Sequence / Matrix / other structure
- Raw observation:
  - „The horse chased by the barn fell.“
- Extended observation:
  - „The/det horse/n chased/pp by/prep the/det barn/n fell/v.“
- Transformed observation:
  - Sequence of BERT embedding vectors for each of the words in this sentence:
    $(\vec{w}_1\ \vec{w}_2 \ldots \vec{w}_7)$

feature engineering

From Naive Bayes to Naive Markov model (I)

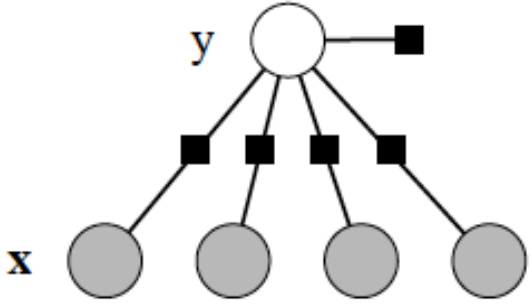
Remember Naive Bayes:

$$P(y, x_1, \ldots x_k) = P(y, \boldsymbol{x}) = P(y) \prod_{i=1}^{k} P(x_i \mid y)$$



Reformulate as factor graph:

$$\phi_Y(y) = P(y), \phi_i(y, x_i) = P(x_i \mid y)$$



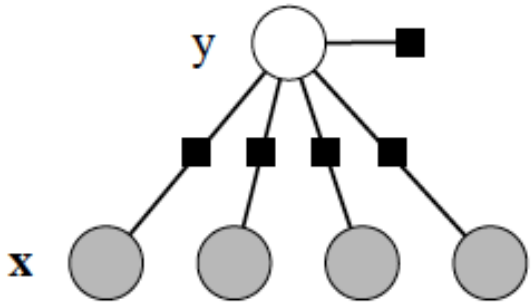
Naive Markov model is a special case of CRF

From Naive Bayes to Naive Markov model (II)

Reformulate as factor graph:

$$\phi_0(y) = P(y),$$

$$\phi_i(y, x_i) = P(x_i \mid y)$$



Model $P(y \mid x)$ as in logistic regression:

$$\log P(y, \boldsymbol{x}) = \lambda_{y,0} + \sum_{i=1}^{k} \lambda_{y,i} x_i,$$

$$P(y, \boldsymbol{x}) = e^{\left(\lambda_{y,0} + \lambda_y \boldsymbol{x}\right)}$$

$$P(\boldsymbol{x}) = Z(\boldsymbol{x}) = \sum_y e^{\left(\lambda_{y,0} + \lambda_y \boldsymbol{x}\right)}$$

Different from Gibbs distribution

$$P(y \mid \boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} e^{\left(\lambda_{y,0} + \lambda_y \boldsymbol{x}\right)}$$

From Naive Bayes to Naive Markov model (III)

Model $P(y \mid \boldsymbol{x})$ as in logistic regression:

$$P(y \mid \boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} e^{\left(\lambda_{y,0} + \lambda_y \boldsymbol{x}\right)}$$

Make λ independent of y by using feature functions:

- for bias: $\lambda_{y,0}$ write $f_{y'}(y, \boldsymbol{x}) = \mathbf{1}_{\{y=y'\}}$
- for weights: $\lambda_{y,i}$ write $f_{y',i}(y, \boldsymbol{x}) = \mathbf{1}_{\{y=y'\}} x_i$
- re-index f

$$P(y \mid \boldsymbol{x}) = \frac{1}{Z(\boldsymbol{x})} e^{\left(\sum_{i=0}^{k} \lambda_i f_i(y, \boldsymbol{x})\right)}$$

Note that $\mathbf{1}_{\{y=y'\}}$ is a function of y, which is 1 if the condition is true

Random fields came from 70ies physics. They invented names and notations that you may encounter nowadays such as this second notation.

Difference:
Now weights are shared between classes!
(before: different per class)

Sigmoid / Softmax

For $Y = \{0,1\}$

Unnormalized:

$$\tilde{P}(y=0 \mid \boldsymbol{x}) = e^{(\lambda_{0,0} + \boldsymbol{\lambda_0 x})} = e^{z_0}$$

$$\tilde{P}(y=1 \mid \boldsymbol{x}) = e^{(\lambda_{1,0} + \boldsymbol{\lambda_1 x})} = e^{z_1}$$

Therefore:

$$P(y=0 \mid \boldsymbol{x}) = \frac{e^{z_0}}{e^{z_0} + e^{z_1}}$$

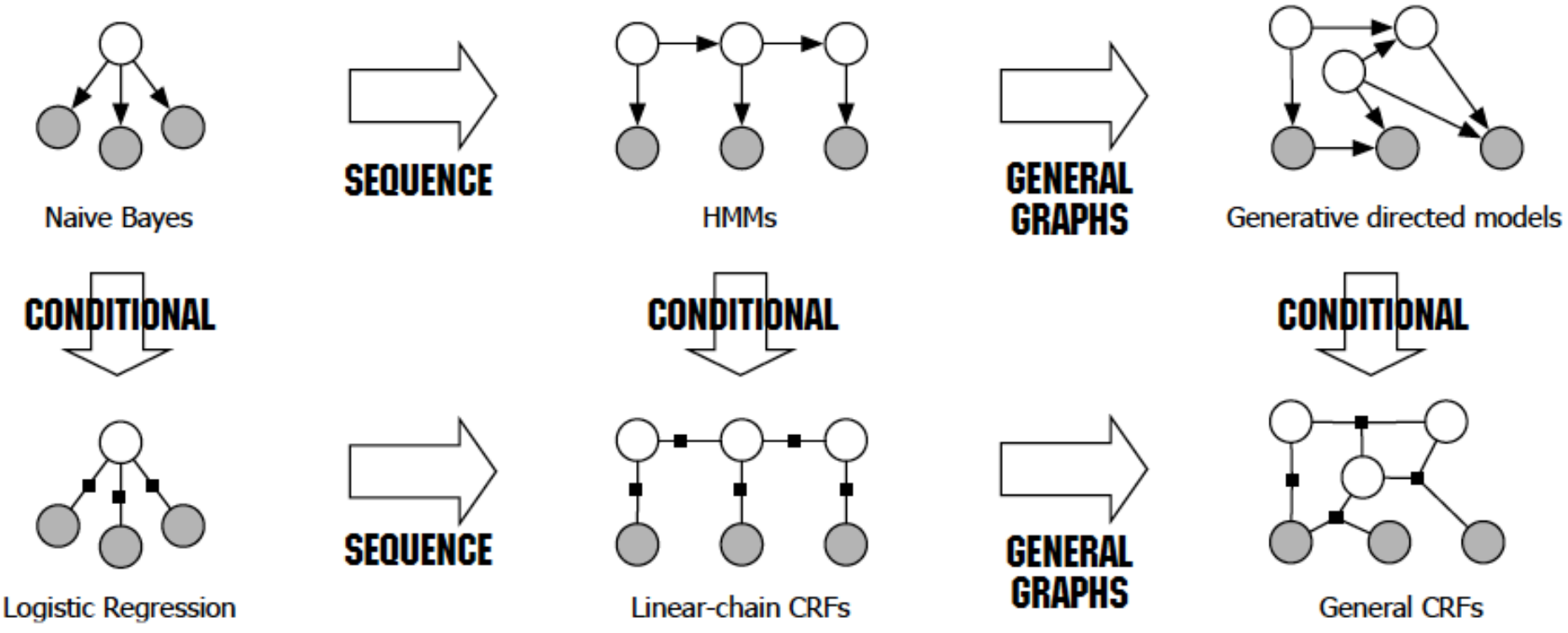
Sigmoid

For $Y = \{1 \ldots n\}$

Generalized:

$$P(y=1 \mid \boldsymbol{x}) = \frac{e^{z_1}}{\sum e^{z_i}}$$

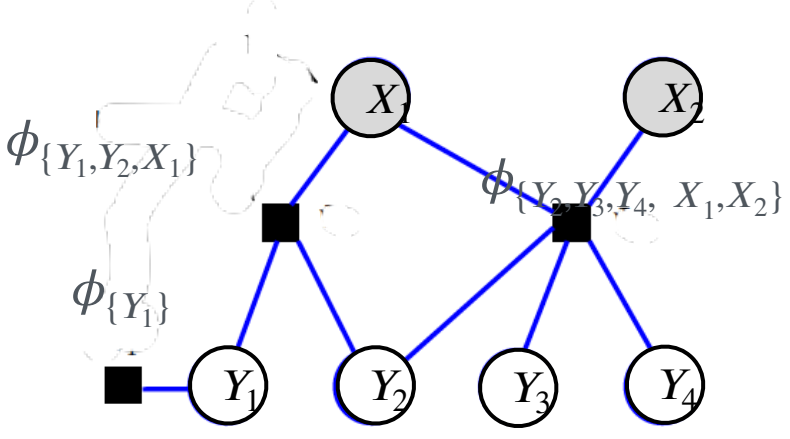
Softmax


Naive Bayes
SEQUENCE
HMMs
GENERAL GRAPHS
Generative directed models
CONDITIONAL
CONDITIONAL
CONDITIONAL
Logistic Regression
SEQUENCE
Linear-chain CRFs
GENERAL GRAPHS
General CRFs

Conditional Random Field as Factor Graph

A factor graph for a CRF is a bi-partite graph $G = (V, F, E)$ of

- variable nodes $V = \mathcal{X} \cup \mathcal{Y}$,
- factor nodes $F = \{\phi_A\}$, with $\text{scope}(\phi_A) = A \subset V$
- edges $E, (\phi_A, v) \in E$, iff $v \in A$



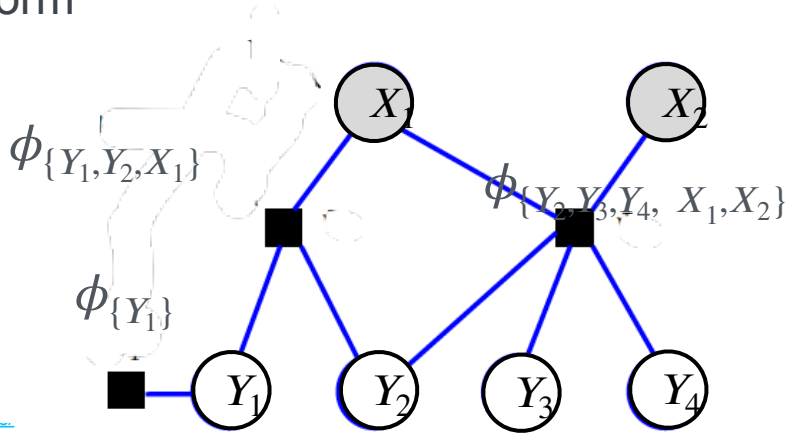
Conditional Random Field as Factor Graph

A factor graph for a CRF is a bi-partite graph $G = (V, F, E)$ of

- variable nodes $V = \mathcal{X} \cup \mathcal{Y}$,
  - Input variables: $\mathcal{X} = \{X_1, X_2\}$, Target variables: $\mathcal{Y} = \{Y_1, Y_2, Y_3, Y_4\}$
- factor nodes $F = \{\phi_A\}$, with $\text{scope}(\phi_A) = A \subset V$
  - Given feature functions: $\{f_{A,k}\}$
  - Local function / factor $\phi_A(x_A, y_A)$ has the form

$$\phi_A(x_A, y_A) = e^{(\sum_k \theta_{A,k} f_{A,k}(x_{A,k}, y_{A,k}))}$$

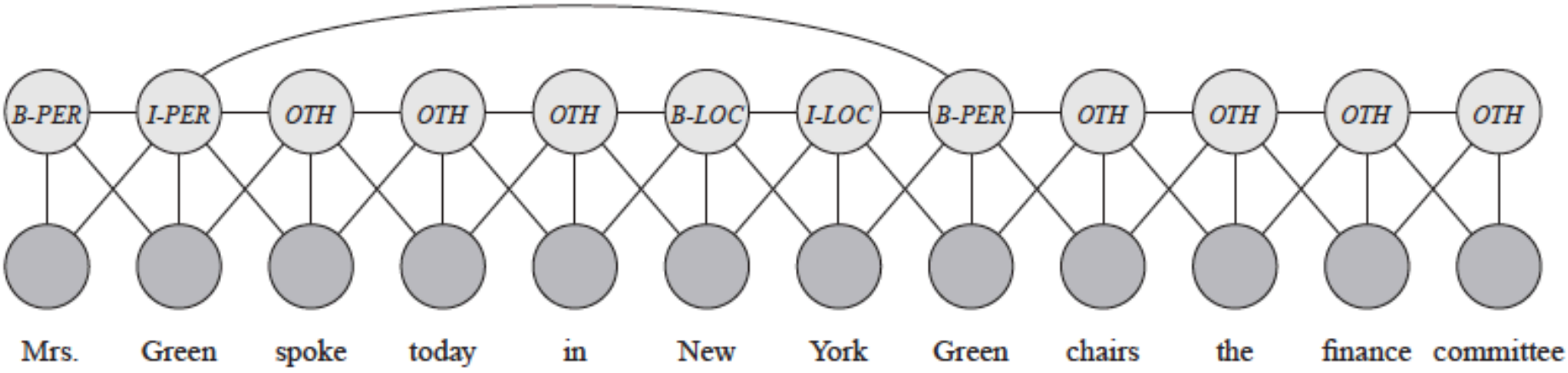
- edges E, $(\phi_A, v) \in E$, iff $v \in A$
- $$P(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_A \phi_A(\mathbf{x}, \mathbf{y})$$




Question 3:
CRF

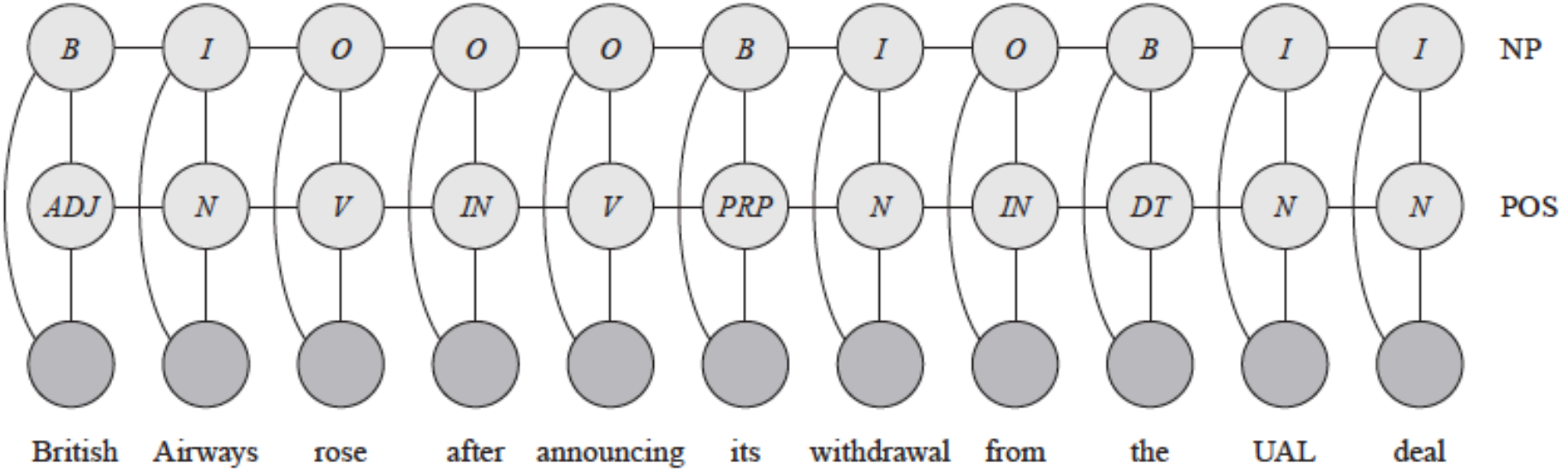
4 CRF Applications

CRFs for Language



KEY

B-PER	Begin person name	*I-LOC*	Within location name
I-PER	Within person name	*OTH*	Not an entitiy
B-LOC	Begin location name		

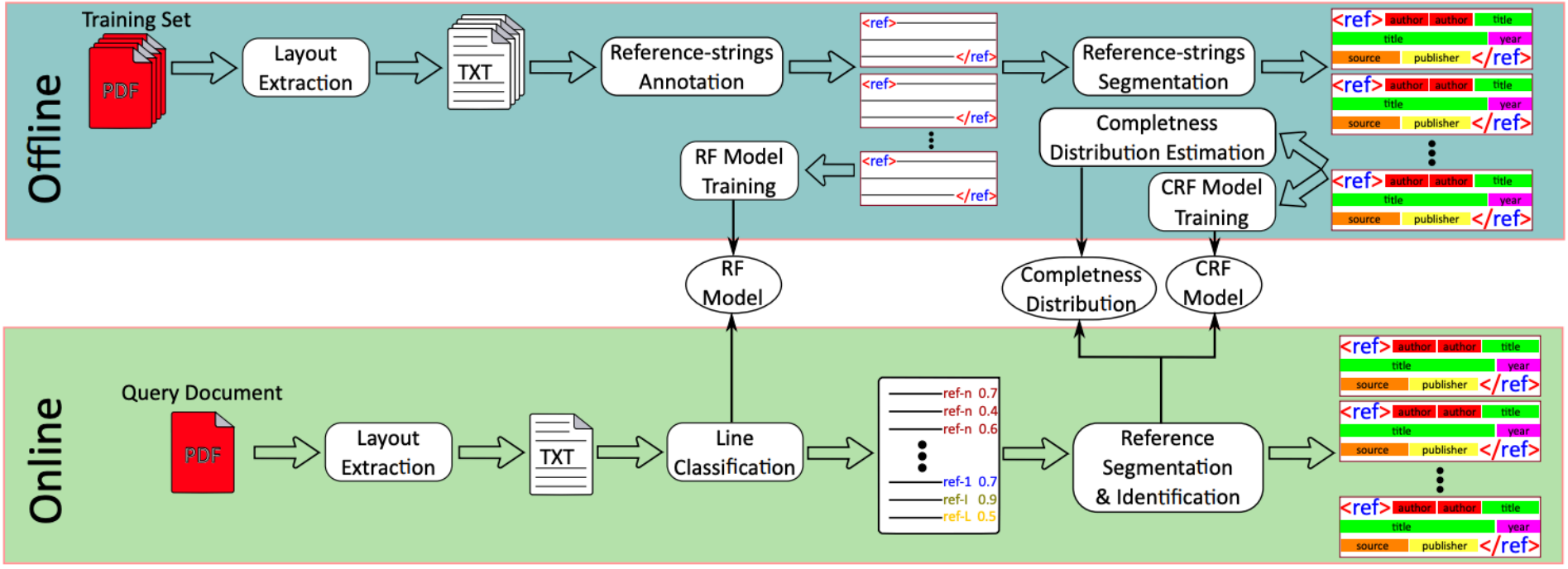


KEY

B	Begin noun phrase	*V*	Verb
I	Within noun phrase	*IN*	Preposition
O	Not a noun phrase	*PRP*	Possesive pronoun
N	Noun	*DT*	Determiner (e.g., a, an, the)
ADJ	Adjective		

Example: Excite

https://excite.informatik.uni-stuttgart.de/



Z. Boukhers, S. Ambhore and S. Staab, "An End-to-End Approach for Extracting and Segmenting High-Variance References from PDF Documents," *2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, Champaign, IL, USA, 2019, pp. 186-195, doi: 10.1109/JCDL.2019.00035.

Try out yourself!

excite.informatik.uni-stuttgart.de/excite

UniStuttgart | LEO English-Germ... | Google Suche | Google | Ilias | Soton | UniKO | Begutachtung | Privat | Projects | Reisen | Suchen | Watch Last week t... | Veranstaltungen | Literatur

EXCITE

Updated: EXparser v1.0.1 is released and running from 22/02/2019 (test it below!)

EXCITE (**Ex**traction of **Cit**ations from PDF Documents) is a toolchain of citation extraction software and particular focus on the German-language social sciences and this is a public service for the project. In the background of this page we are using CERMINE for extracting content from PDF files and Exparser for reference string extraction and segmentation.

How to start the process:

- **First:** Choose a pdf file by click on "Choose File" button. (size of the file should be less than 20 MB)
- **Second:** Click on "Upload file" button to start the process. Please enter your email address if you would like to receive follow up code by email. (It is optional)
- **tip:** After uploading a file a code will be displayed on the screen. This code is necessary for follow up the result of reference extraction. If you would like to load last follow up code related to last uploaded file in your browser.

How to check the result:

- **First:** Enter the follow-up code in the appropriate box in the right-hand of page then click on "Display References" button.
- **Second:** The result will be displayed on screen if it available.
- **tip:** Extracting References process will take a little time, at least 30 seconds, and it completely depends on the size of the file.
- **tip:** Click on "Load last follow up code" If you would like load last follow up code related to last uploaded file in your browser.

This is a first version of the our public web service. How we can improve it? **Please let us know if you have any feedback for us by clicking on this link.**

Datei auswählen | Keine ausgewählt

Enter your email if you would like to receive follow up code by email.(It is optional)

Upload File

If you already have a code, Enter code to load data.

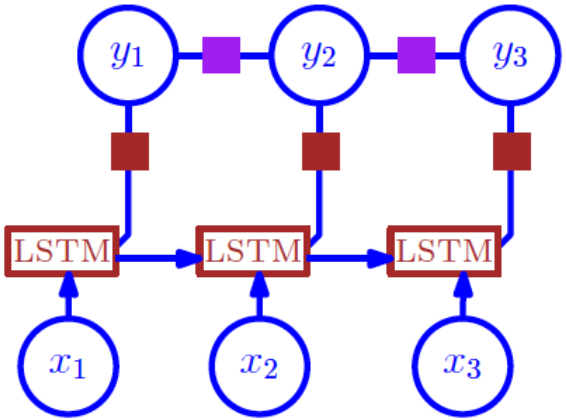
Enter code to display References.

Load Last follow up code | Display References

Current pattern in 2020

LSTM-CRF

Idee: Modelliere Eigenschaften der Ausgabevariablen mit einem PGM, nutze flexibles RNN um Dateneigenschaften zu lernen.



- Standard linear-chain CRF:
$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \prod_i \exp(\sum_j \lambda_j f_j(y_{i-1}, y_i, \vec{x}, i))$$
- Andere Formulierung:
$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \cdot \prod_i \exp(\sum_j \lambda_j f(y_{i-1}, y_i)) \cdot \prod_i \exp(\sum_j \lambda_j f(\vec{x}, y_i, i))$$
- Ersetze datenbezogenes Log-lineares Modell durch LSTM:
$$p(\vec{y} \mid \vec{x}) = \frac{1}{Z(\vec{x})} \cdot \prod_i \exp(\sum_j \lambda_j f(y_{i-1}, y_i)) \cdot \prod_i \text{LSTM}(\vec{x}_i, y_i)$$

Dies ist ein etabliertes Modell.

LSTM-CRF Anwendungsbeispiele (1)

Labeling von Wörtern in Text mit einer IOBE-Sequenz (Huang et al., 2015)

$\vec{x} =$	the	Severe	acute	respiratory	syndrome	coronavirus	2	...
$\vec{y} =$	O	B	I	I	I	I	E	...

CRF-Schicht zeigt Fehlerreduktion von etwa 10 %

Lineare CRF-Schicht, z.B. PyTorch oder AllenNLP

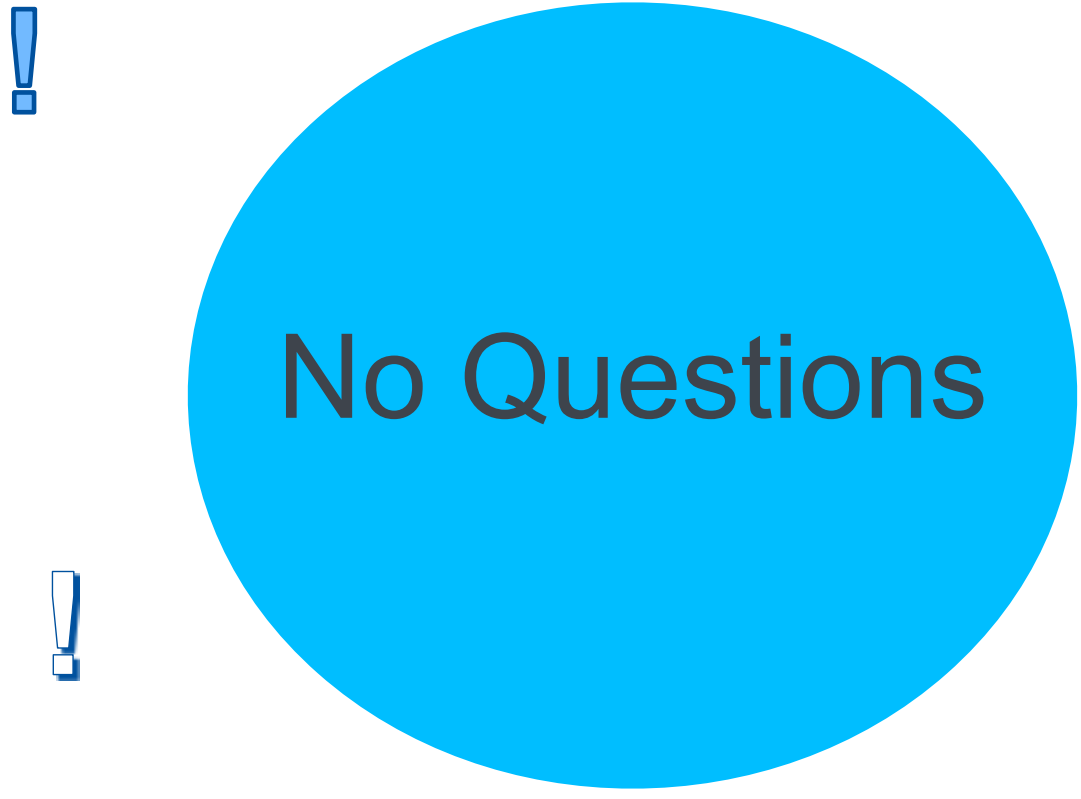
LSTM-CRF Anwendungsbeispiele (2)

2D-Variante: Bildsegmentierung (Zheng et al., 2015)

Original image (hover to highlight segmented parts)

Semantic segmentation

Verbesserungen insbesondere bei filigranen Strukturen
(http://www.robots.ox.ac.uk/~szheng/crfasrnndemo/)


No Questions

Universität Stuttgart
IPVS

Thank you!

Steffen Staab

E-Mail Steffen.staab@ipvs.uni-stuttgart.de
Telefon +49 (0) 711 685 to be defined
www. ipvs.uni-stuttgart.de/departments/ac/

Universität Stuttgart
Analytic Computing, IPVS
Universitätsstraße 32, 50569 Stuttgart