

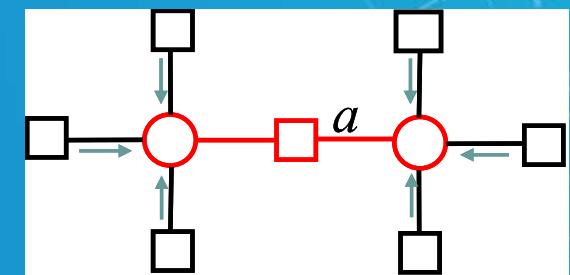
Probabilistic Graphical Models

Variational Inference: Loopy Belief Propagation

Eric Xing

Lecture 11a, February 20, 2019

Reading: see class homepage





Inference Problems

- Compute the likelihood of observed data $p(x_A)$
- Compute the marginal distribution $p(x_A|x_B)$ over a particular subset of nodes $A \subset V$
- Compute the conditional distribution $\hat{x} = \arg \max_{x \in \mathcal{X}^m} p(x)$ for disjoint subsets A and B
- Compute a mode of the density
- Methods we have

Brute force

Elimination



Message Passing

(Forward-backward , Max-product
/BP, Junction Tree)

Individual computations independent

Sharing intermediate terms

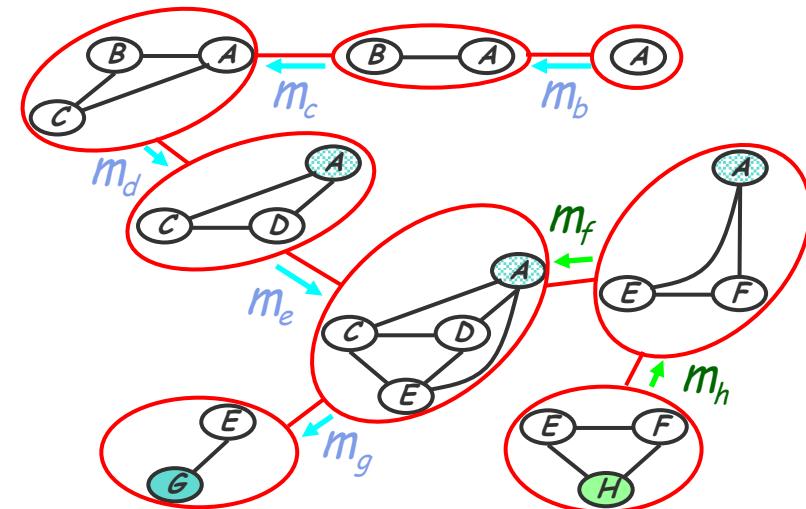




From Elimination to Message Passing

- Recall that Induced dependency during marginalization is captured in elimination cliques
 - Summation \leftrightarrow elimination
 - Intermediate term \leftrightarrow elimination clique

$$\begin{aligned} & P(a)P(b)P(c|b)P(d|a)P(e|c,d)P(f|a)P(g|e)P(h|e,f) \\ \Rightarrow & P(a)P(b)P(c|b)P(d|a)P(e|c,d)P(f|a)P(g|e)\phi_h(e,f) \\ \Rightarrow & P(a)P(b)P(c|b)P(d|a)P(e|c,d)P(f|a)\phi_g(e)\phi_h(e,f) \\ \Rightarrow & P(a)P(b)P(c|b)P(d|a)P(e|c,d)\phi_f(a,e) \\ \Rightarrow & P(a)P(b)P(c|b)P(d|a)\phi_e(a,c,d) \\ \Rightarrow & P(a)P(b)P(c|b)\phi_d(a,c) \\ \Rightarrow & P(a)P(b)\phi_c(a,b) \\ \Rightarrow & P(a)\phi_b(a) \\ \Rightarrow & \phi(a) \end{aligned}$$



- Elimination \equiv message passing on a clique tree





Complexity

- ❑ The overall complexity is determined by the number of the largest elimination clique
- ❑ What is the largest elimination clique? – a pure graph theoretic question
- ❑ **Tree-width k** : one less than the smallest achievable value of the cardinality of the largest elimination clique, ranging over all possible elimination ordering
- ❑ “good” elimination orderings lead to **small cliques** and hence reduce complexity (**what will happen if we eliminate “e” first in the above graph?**)
- ❑ Find the best elimination ordering of a graph --- NP-hard
→ Inference is NP-hard
- ❑ But there often exist “obvious” optimal or near-opt elimination ordering

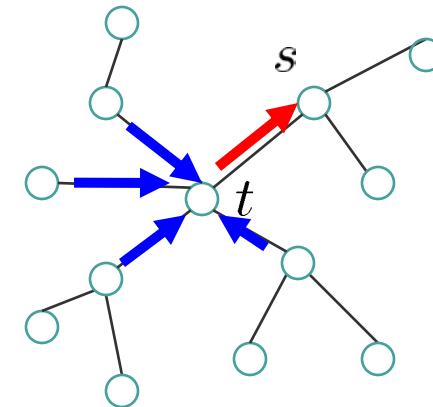




Sum-Product Algorithm

- Tree-structured GMs

$$p(x_1, \dots, x_m) = \frac{1}{Z} \prod_{s \in V} \psi_s(x_s) \prod_{(s,t) \in E} \psi_{st}(x_s, x_t)$$



- Message Passing on Trees:

$$M_{t \rightarrow s}(x_s) \leftarrow \kappa \sum_{x'_t} \left\{ \psi_{st}(x_s, x'_t) \psi_t(x'_t) \prod_{u \in N(t) \setminus s} M_{u \rightarrow t}(x'_t) \right\}$$

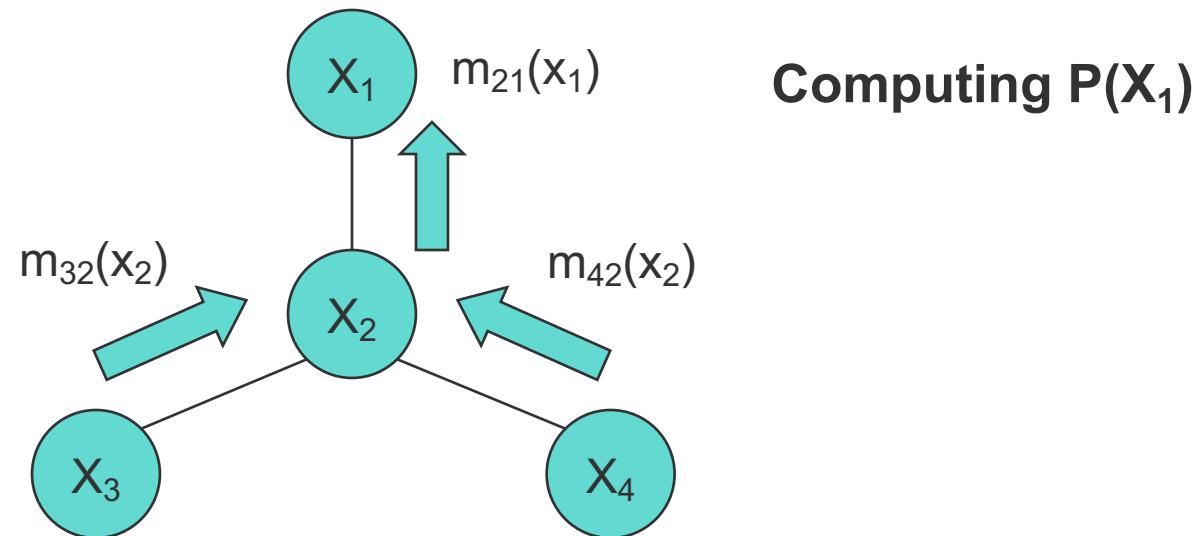
- On trees, converge to a unique fixed point after a finite number of iterations





The message passing protocol:

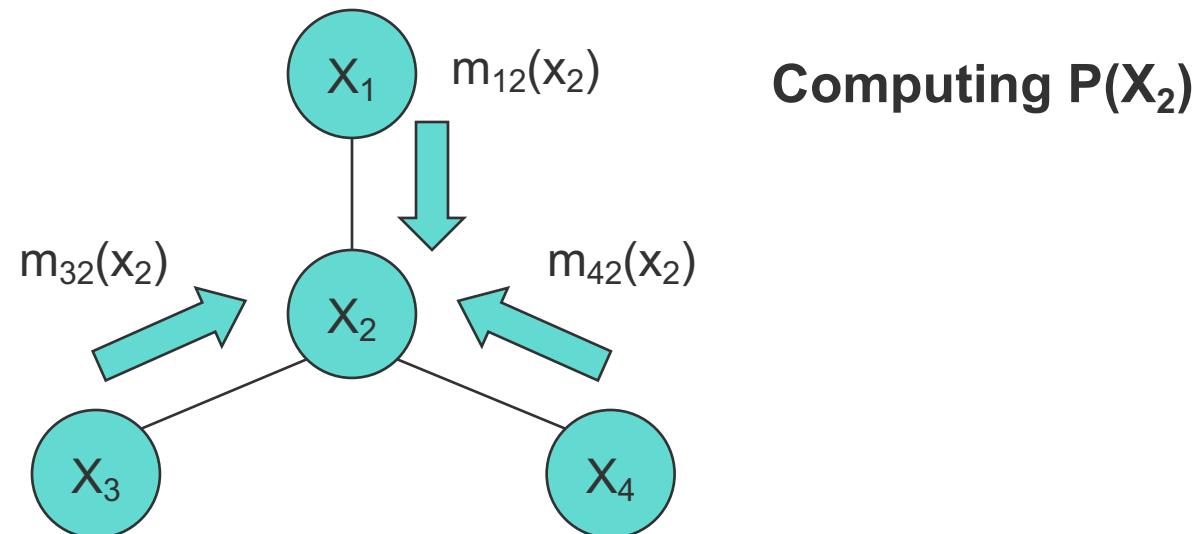
- ❑ A node can send a message to its neighbors when (and only when) it has received messages from all its *other* neighbors.
- ❑ Computing node marginals:
 - ❑ Naïve approach: consider each node as the root and execute the message passing algorithm





The message passing protocol:

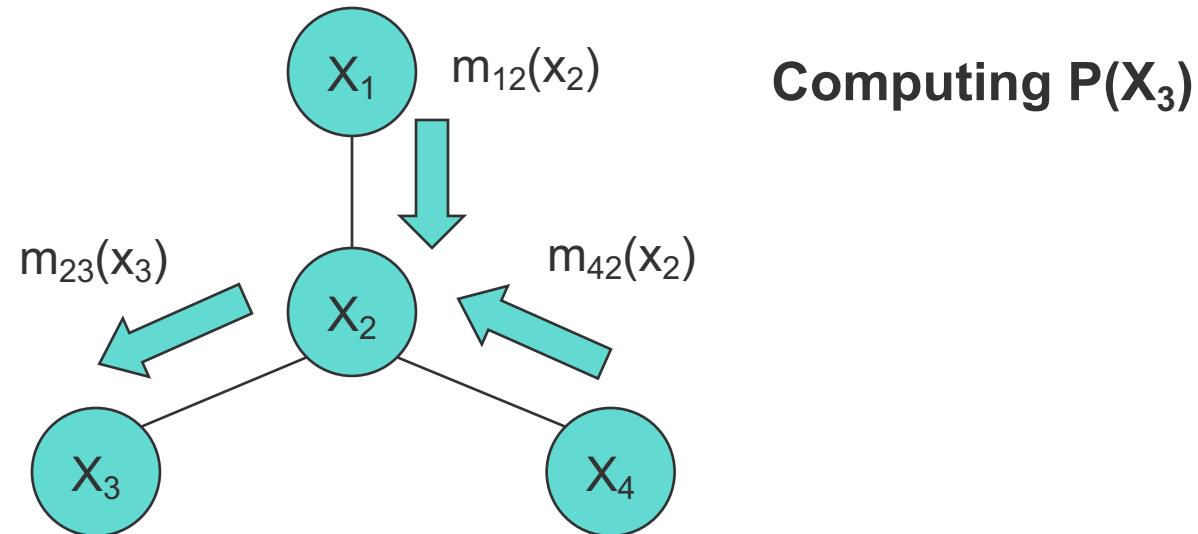
- ❑ A node can send a message to its neighbors when (and only when) it has received messages from all its *other* neighbors.
- ❑ Computing node marginals:
 - ❑ Naïve approach: consider each node as the root and execute the message passing algorithm





The message passing protocol:

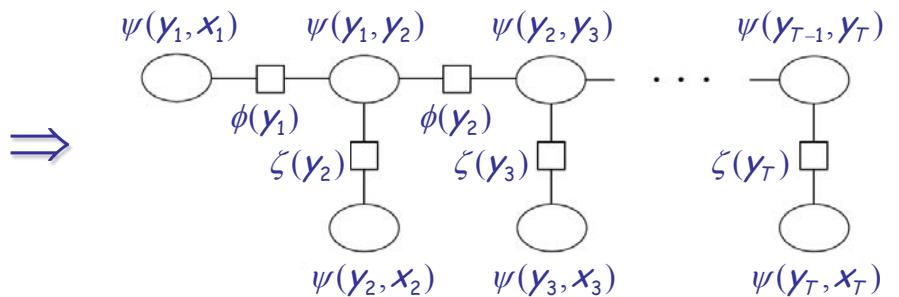
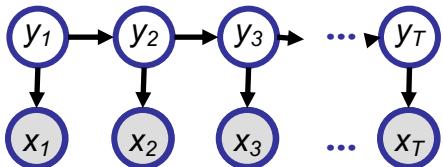
- ❑ A node can send a message to its neighbors when (and only when) it has received messages from all its *other* neighbors.
- ❑ Computing node marginals:
 - ❑ Naïve approach: consider each node as the root and execute the message passing algorithm





Message Passing for HMMs

- A “click tree” for the HMM



- Rightward pass

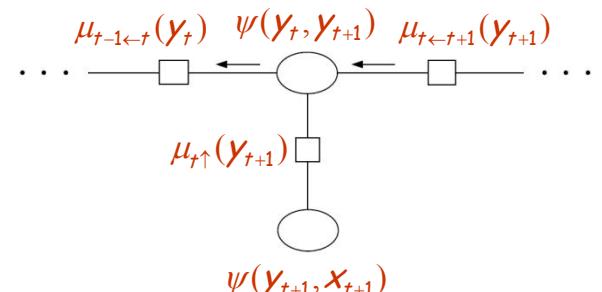
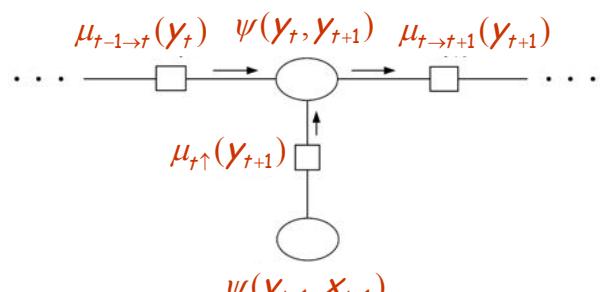
$$\begin{aligned}\mu_{t \rightarrow t+1}(y_{t+1}) &= \sum_{y_t} \psi(y_t, y_{t+1}) \mu_{t-1 \rightarrow t}(y_t) \mu_{t \uparrow}(y_{t+1}) \\ &= \sum_{y_t} p(y_{t+1} | y_t) \mu_{t-1 \rightarrow t}(y_t) p(x_{t+1} | y_{t+1}) \\ &= p(x_{t+1} | y_{t+1}) \sum_{y_t} a_{y_t, y_{t+1}} \mu_{t-1 \rightarrow t}(y_t)\end{aligned}$$

- This is exactly the *forward algorithm!*

- Leftward pass ...

$$\begin{aligned}\mu_{t-1 \leftarrow t}(y_t) &= \sum_{y_{t+1}} \psi(y_t, y_{t+1}) \mu_{t \leftarrow t+1}(y_{t+1}) \mu_{t \uparrow}(y_{t+1}) \\ &= \sum_{y_{t+1}} p(y_{t+1} | y_t) \mu_{t \leftarrow t+1}(y_{t+1}) p(x_{t+1} | y_{t+1})\end{aligned}$$

- This is exactly the *backward algorithm!*





Correctness of BP on tree

- ❑ Collollary: the synchronous implementation is "non-blocking"
- ❑ Thm: The Message Passage Guarantees obtaining all marginals in the tree

$$m_{ji}(x_i) = \sum_{x_j} \left(\psi(x_j) \psi(x_i, x_j) \prod_{k \in N(j) \setminus i} m_{kj}(x_j) \right)$$





Local Consistency → Global Consistency

- Given a set of functions $\{\tau_C, C \in \mathcal{C}\}$ and $\{\tau_S, S \in \mathcal{S}\}$ associated with the cliques and separator sets
- They are locally consistent if:

$$\sum_{x'_S} \tau_S(x'_S) = 1, \forall S \in \mathcal{S}$$

$$\sum_{x'_C | x'_S = x_S} \tau_C(x'_C) = \tau_S(x_S), \forall C \in \mathcal{C}, S \subset C$$

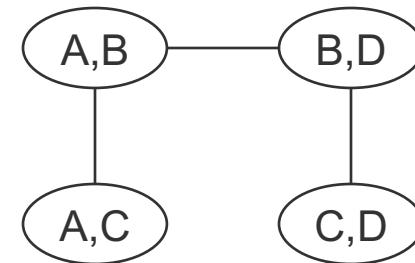
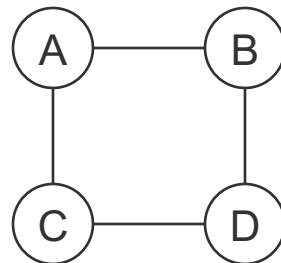
- For junction trees, local consistency is equivalent to global consistency!
- What about non-tree?





A problem

- Consider the following graph and a corresponding clique tree



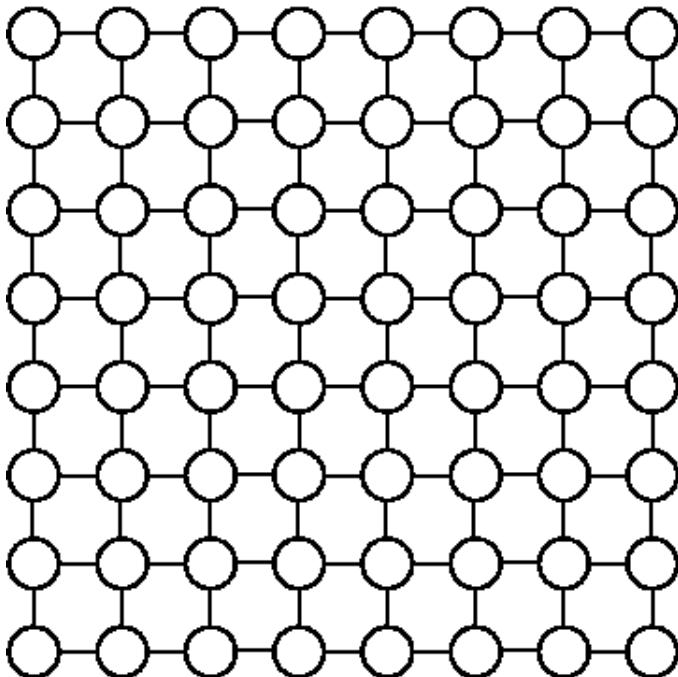
- Note that C appears in two non-neighboring cliques
- *Question:* with the previous message passage, can we ensure that the probability associated with C in these two (non-neighboring) cliques consistent?
- Answer: No. It is not true that in general local consistency implies global consistency
- What else do we need to get such a guarantee?





Why Approximate Inference?

- Why can't we just run “clique tree” on this graph?



$$p(X) = \frac{1}{Z} \exp \left\{ \sum_{i < j} \theta_{ij} X_i X_j + \sum_i \theta_{i0} X_i \right\}$$

- If $N \times N$ grid, tree width at least N
- N can be a huge number (~ 1000 s of pixels)
 - If $N \sim O(1000)$, we have a clique with 2^{100} entries





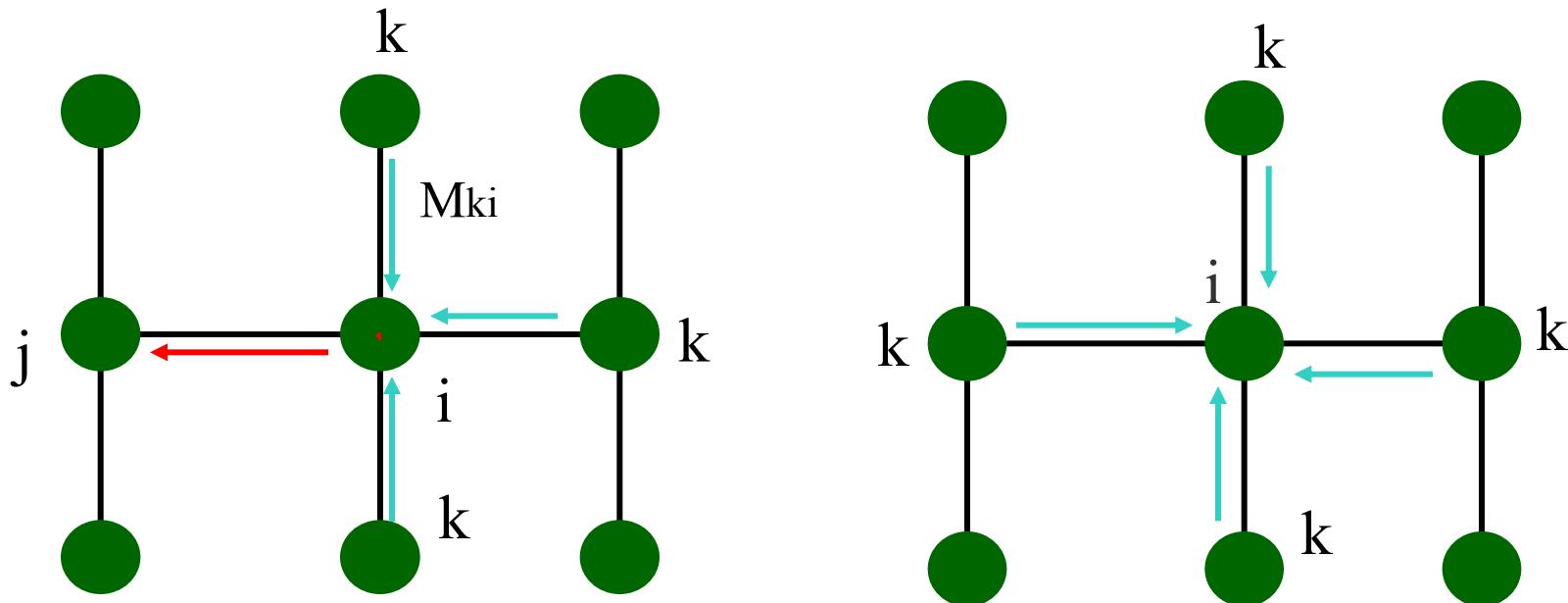
Approaches to inference

- ❑ Exact inference algorithms
 - ❑ The elimination algorithm
 - ❑ Message-passing algorithm (sum-product, belief propagation)
 - ❑ The junction tree algorithms
- ❑ Approximate inference techniques
 - ❑ Variational algorithms
 - ❑ Loopy belief propagation
 - ❑ Mean field approximation
 - ❑ Stochastic simulation / sampling methods
 - ❑ Markov chain Monte Carlo methods





Recap: Belief Propagation



- BP Message-update Rules

$$M_{i \rightarrow j}(x_j) \propto \sum_{x_i} \psi_{ij}(x_i, x_j) \psi_i(x_i) \prod_k M_{k \rightarrow i}(x_i)$$

↑ ↑
Compatibilities (interactions) external evidence

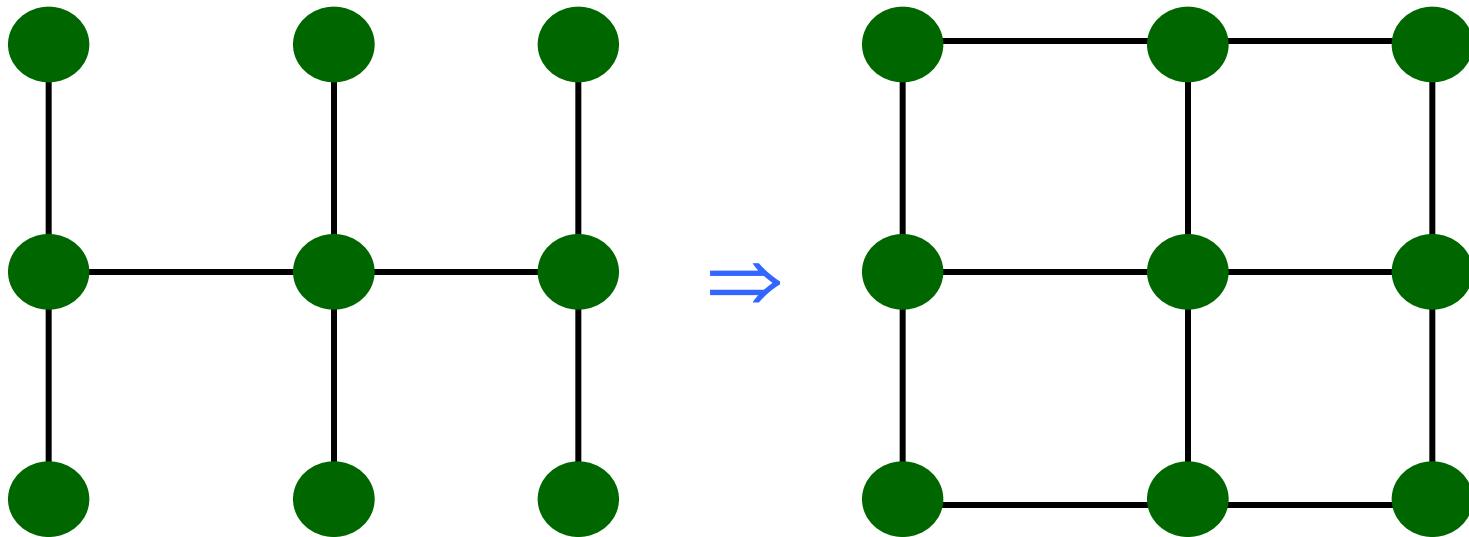
$$b_i(x_i) \propto \psi_i(x_i) \prod_k M_k(x_k)$$

- BP on trees always converges to exact marginals (cf. Junction tree algorithm)



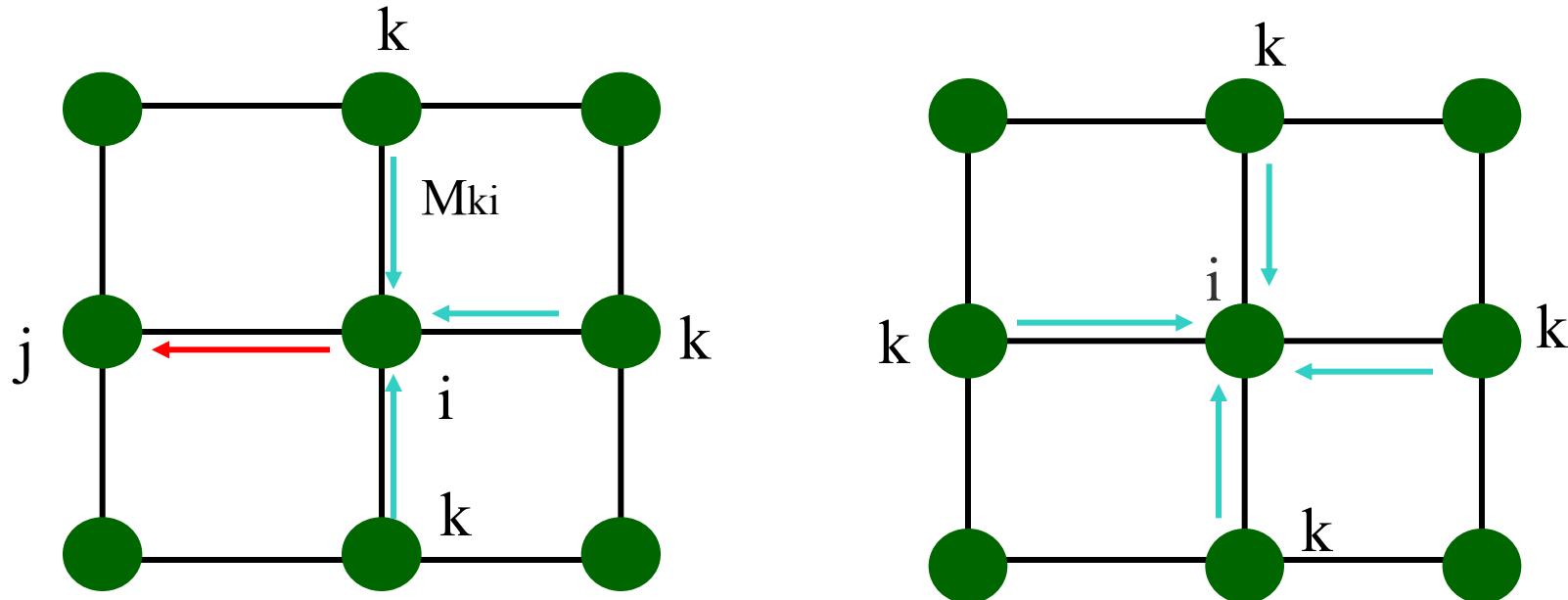


What if the graph is loopy?





Belief Propagation on loopy graphs



- BP Message-update Rules

$$M_{i \rightarrow j}(x_j) \propto \sum_{x_i} \psi_{ij}(x_i, x_j) \psi_i(x_i) \prod_k M_{k \rightarrow i}(x_i)$$

↑ ↑
 Compatibilities (interactions) external evidence

$$b_i(x_i) \propto \psi_i(x_i) \prod_k M_k(x_k)$$

- May not converge or converge to a wrong solution





Loopy Belief Propagation

- ❑ If BP is used on graphs with loops, messages may circulate indefinitely
- ❑ But let's run it anyway and hope for the best ... ☺
- ❑ Empirically, a good approximation is still achievable
 - ❑ Stop after fixed # of iterations
 - ❑ Stop when no significant change in beliefs
 - ❑ If solution is not oscillatory but converges, it usually is a good approximation

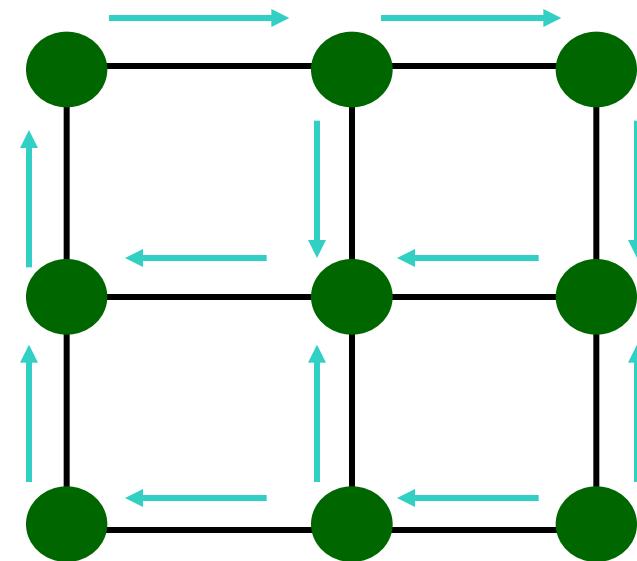
[Loopy-belief Propagation for Approximate Inference: An Empirical Study](#)
Kevin Murphy, Yair Weiss, and Michael Jordan.
UAI '99 (Uncertainty in AI).]





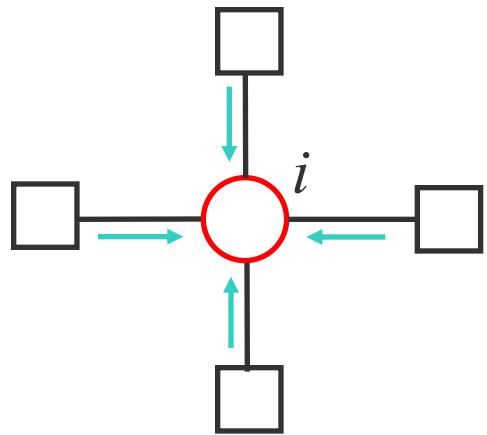
So what is going on?

- ❑ Is it a dirty hack that you bet your luck?



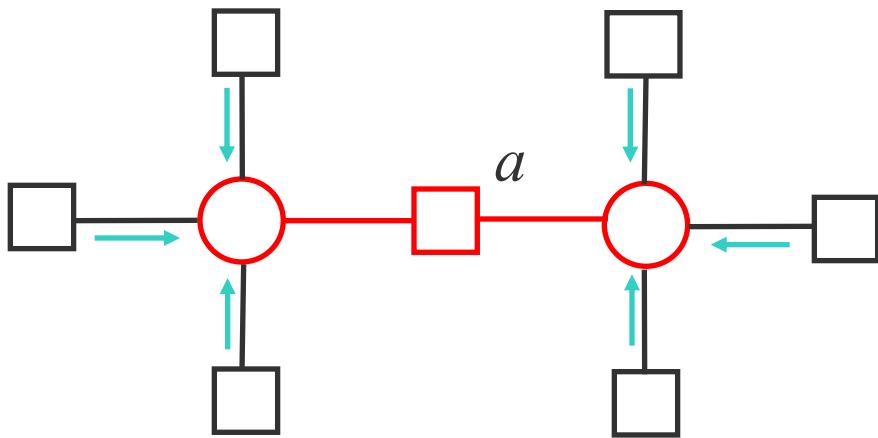


Beliefs and messages in “Factored Graphs”



$$b_i(x_i) \propto f_i(x_i) \prod_{a \in N(i)} m_{a \rightarrow i}(x_i)$$

↑
“beliefs” “messages”



$$m_{i \rightarrow a}(x_i) = \prod_{c \in N(i) \setminus a} m_{c \rightarrow i}(x_i)$$

$$b_a(X_a) \propto f_a(X_a) \prod_{i \in N(a)} m_{i \rightarrow a}(x_i)$$

$$m_{a \rightarrow i}(x_i) = \sum_{X_a \setminus x_i} f_a(X_a) \prod_{j \in N(a) \setminus i} m_{j \rightarrow a}(x_j)$$





Approximate Inference

- For the actual distribution P : $P(X) = 1/Z \prod_{f_a \in F} f_a(X_a)$
- We wish to find a distribution Q such that Q is a “good” approximation to P
- The quality of the approximation can be measured by:

$$\begin{aligned} KL(Q \| P) &= \sum_X Q(X) \log \left(\frac{Q(X)}{P(X)} \right) \\ &= \sum_X Q(X) \log Q(X) - \sum_X Q(X) \log P(X) \\ &= -H_Q(X) - E_Q \log P(X) \\ &= -H_Q(X) - \log 1/Z - \sum_{f_a \in F} E_Q \log f_a(X_a) = -H_Q(X) - \underbrace{\sum_{f_a \in F} E_Q \log f_a(X_a)}_{F(P, Q)} + \log Z \end{aligned}$$

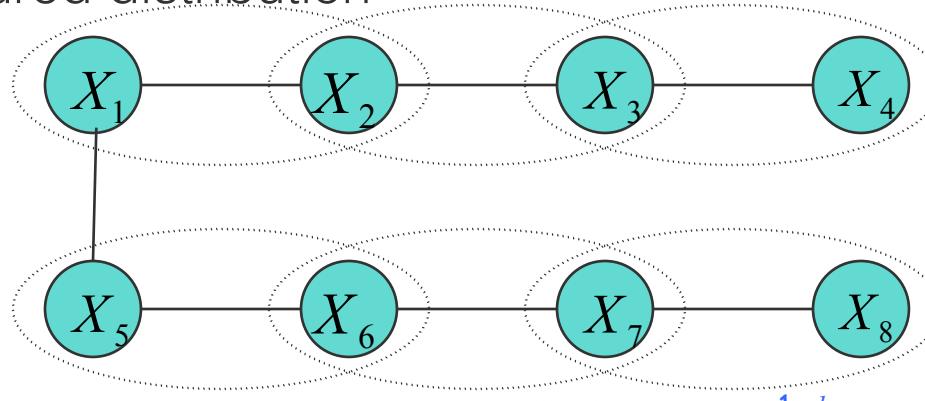
- $F(P, Q)$ is called the “Free energy” *





Tree Energy Functionals

- Consider a tree-structured distribution



- The probability can be written as: $b(\mathbf{x}) = \prod_a b_a(\mathbf{x}_a) \prod_i b_i(x_i)^{1-d_i}$

$$H_{tree} = -\sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln b_a(\mathbf{x}_a) + \sum_i (d_i - 1) \sum_{\mathbf{x}_i} b_i(\mathbf{x}_i) \ln b_i(\mathbf{x}_i)$$

$$F_{Tree} = \sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln \frac{b_a(\mathbf{x}_a)}{f_a(\mathbf{x}_a)} + \sum_i (1 - d_i) \sum_{\mathbf{x}_i} b_i(\mathbf{x}_i) \ln b_i(\mathbf{x}_i)$$

$$= F_{12} + F_{23} + \dots + F_{67} + F_{78} - F_1 - F_5 - F_2 - F_6 - F_3 - F_7$$

- involves summation over edges and vertices and is therefore easy to compute





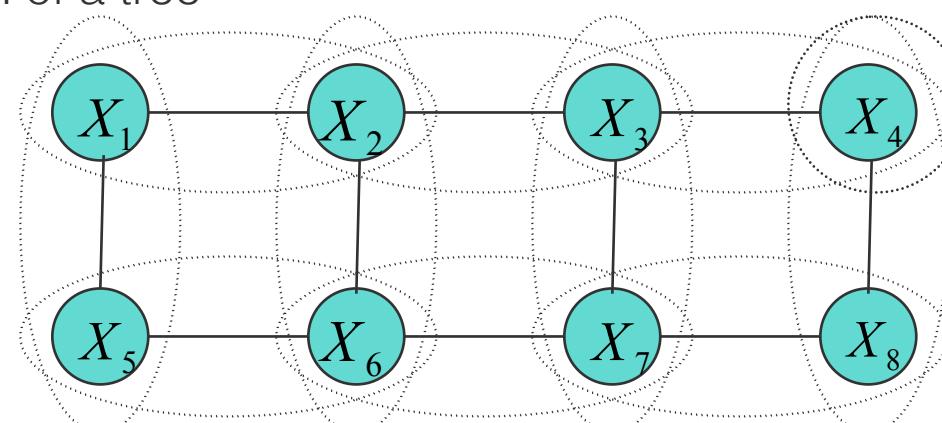
Bethe Approximation to Gibbs Free Energy

- For a general graph, choose $\hat{F}(P, Q) = F_{Bethe}$

$$H_{Bethe} = -\sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln b_a(\mathbf{x}_a) + \sum_i (d_i - 1) \sum_{\mathbf{x}_i} b_i(\mathbf{x}_i) \ln b_i(\mathbf{x}_i)$$

$$F_{Bethe} = \sum_a \sum_{\mathbf{x}_a} b_a(\mathbf{x}_a) \ln \frac{b_a(\mathbf{x}_a)}{f_a(\mathbf{x}_a)} + \sum_i (1 - d_i) \sum_{\mathbf{x}_i} b_i(\mathbf{x}_i) \ln b_i(\mathbf{x}_i) = -\langle f_a(\mathbf{x}_a) \rangle - H_{beta}$$

- Called the “Bethe approximation” after the physicist Hans Bethe
 - Equal to the exact Gibbs free energy when the factor graph is a tree
 - In general, H_{Bethe} is **not** the same as the H of a tree
-
- For a “loopy graph like this:



$$F_{beta} = F_{12} + F_{23} + \dots + F_{67} + F_{78} - F_1 - F_5 - 2F_2 - 2F_6 - \dots - F_8$$





Constrained Minimization of the Bethe Free Energy

$$L = F_{Bethe} + \sum_i \gamma_i \left\{ \sum_{x_i} b_i(x_i) - 1 \right\}$$

$$+ \sum_a \sum_{i \in N(a)} \sum_{x_i} \lambda_{ai}(x_i) \left\{ \sum_{X_a \setminus x_i} b_a(X_a) - b_i(x_i) \right\}$$

$$\frac{\partial L}{\partial b_i(x_i)} = 0 \quad \longrightarrow \quad b_i(x_i) \propto \exp \left(\frac{1}{d_i - 1} \sum_{a \in N(i)} \lambda_{ai}(x_i) \right)$$

$$\frac{\partial L}{\partial b_a(X_a)} = 0 \quad \longrightarrow \quad b_a(X_a) \propto \exp \left(-E_a(X_a) + \sum_{i \in N(a)} \lambda_{ai}(x_i) \right)$$



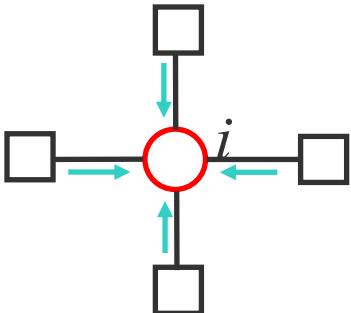


Bethe = BP on FG

- We had:

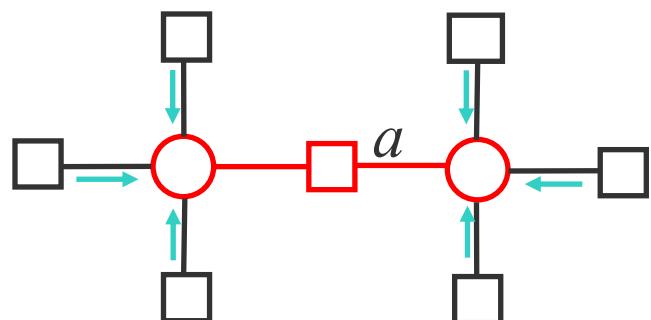
$$b_i(x_i) \propto \exp\left(\frac{1}{d_i - 1} \sum_{a \in N(i)} \lambda_{ai}(x_i)\right) \quad b_a(X_a) \propto \exp\left(-\log f_a(X_a) + \sum_{i \in N(a)} \lambda_{ai}(x_i)\right)$$

- Identify $\lambda_{ai}(x_i) = \log(m_{i \rightarrow a}(x_i)) = \log \prod_{b \in N(i) \setminus a} m_{b \rightarrow i}(x_i)$
- to obtain BP equations:



$$b_i(x_i) \propto f_i(x_i) \prod_{a \in N(i)} m_{a \rightarrow i}(x_i)$$

↑
“beliefs” ↑
“messages”



$$b_a(X_a) \propto f_a(X_a) \prod_{i \in N(a)} \prod_{c \in N(i) \setminus a} m_{c \rightarrow i}(x_i)$$

The “belief” is the BP approximation of the marginal probability.





Summary

- ❑ We defined an objective function (F) for approximate inference
- ❑ However, we found that optimizing this function was hard
- ❑ We first approximated objective function F to simpler F_{bethe}
 - ❑ Minima of F_{bethe} turned out to be fixed points of BP
- ❑ Then we extended this to more complicated approximations
 - ❑ The resulting algorithms come under a family called Generalized Belief Propagation
- ❑ Next class, we will cover other methods of approximations



Probabilistic Graphical Models

Approximate Inference and Topic Models:
Mean Field and Loopy Belief Prop

Eric Xing

Lecture 11b, February 20, 2019

Reading: see class homepage





Probabilistic Topic Models

- ❑ Humans cannot afford to deal with (e.g., search, browse, or measure similarity) a huge number of text documents
- ❑ We need computers to help out ...





How to get started for a new modeling task?

Here are some important elements to consider before you start:

- Task:
 - Embedding? Classification? Clustering? Topic extraction? ...
- Data representation:
 - Input and output (e.g., continuous, binary, counts, ...)
- Model:
 - BN? MRF? Regression? SVM?
- Inference:
 - Exact inference? MCMC? Variational?
- Learning:
 - MLE? MCLE? Max margin?
- Evaluation:
 - Visualization? Human interpretability? Perplexity? Predictive accuracy?

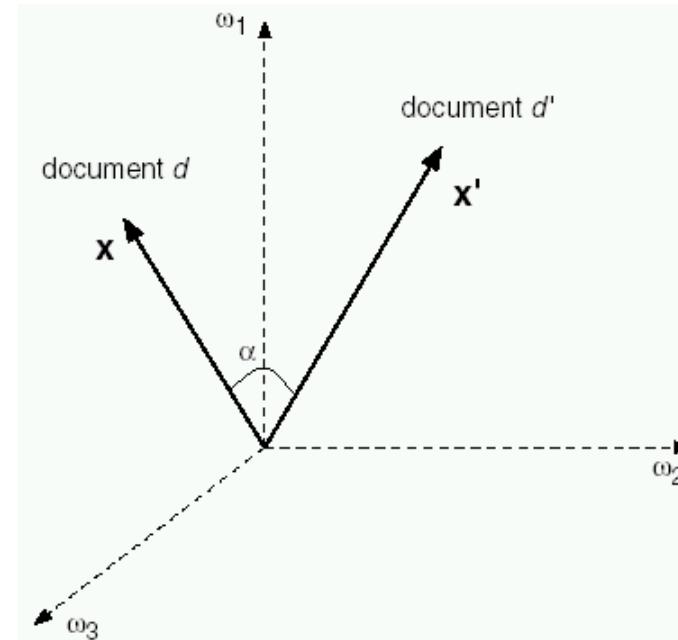
It is better to consider one element at a time!





Tasks: document embedding

- Say, we want to have a mapping ..., so that



- Compare simil
- Classify contents
- Cluster/group/categorizing
- Distill semantics and perspectives
- ..





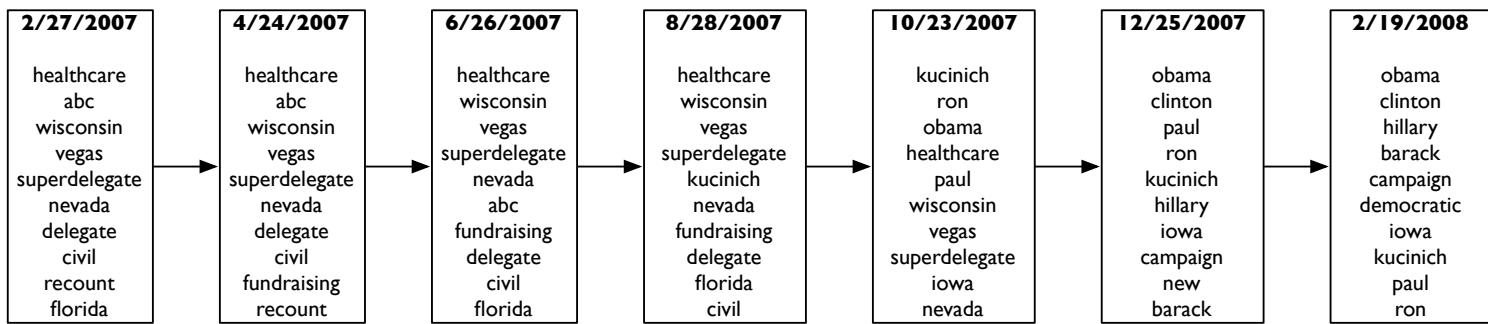
Summarizing the data using topics

Bayesian modeling	Visual cortex	Education	Market
Bayesian model	cortex	students	market
inference	cortical areas	education	economic
models	visual	learning	financial
probability	area	educational	economics
probabilistic	primary	teaching	markets
Markov prior	connections	school	returns
hidden approach	ventral	student	price
	cerebral	skills	stock
	sensory	teacher	value
		academic	investment





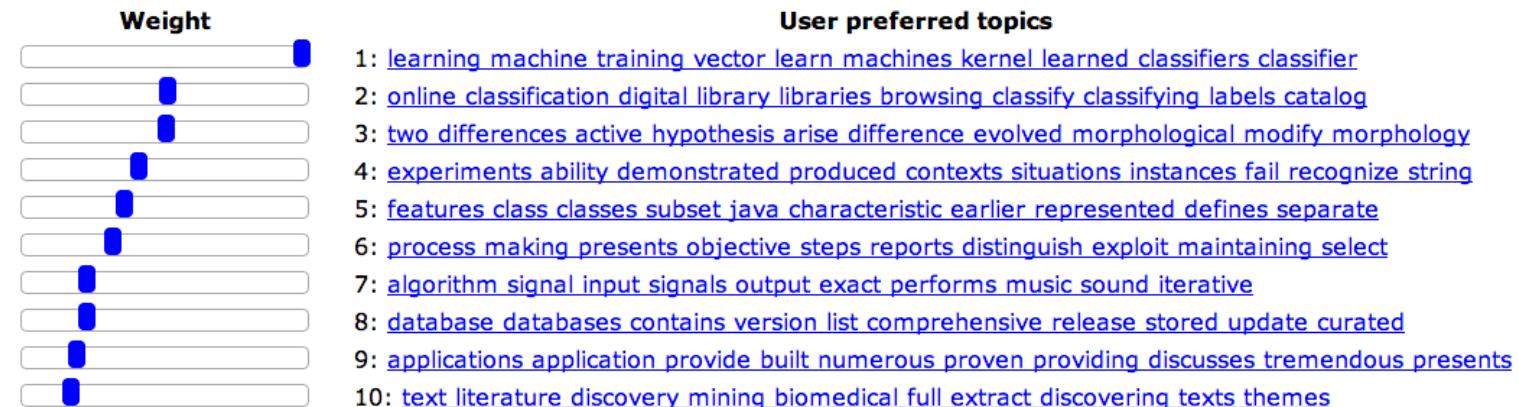
See how data changes over time





User interest modeling using topics

User interest profile (adjustable with sliders---Changing these changes recommendations.)



<http://cogito-demos.ml.cmu.edu/cgi-bin/recommendation.cgi>



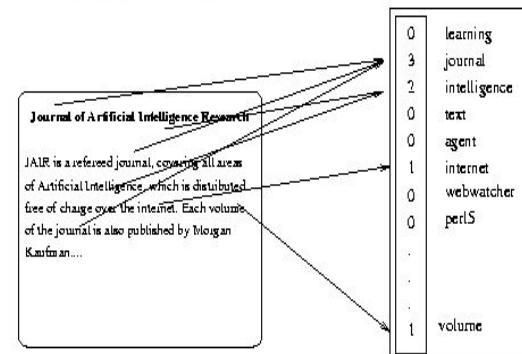


Representation: Bag of Words Representation

□ Data:

As for the Arabian and Palestinian voices that are against the current negotiations and the so-called peace process, they are not against peace per se, but rather for their well-founded predictions that Israel would NOT give an inch of the West bank (and most probably the same for Golan Heights) back to the Arabs. An 18 months of "negotiations" in Madrid, and Washington proved these predictions. Now many will jump on me saying why are you blaming israelis for no-result negotiations. I would say why would the Arabs stall the negotiations, what do they have to loose ?

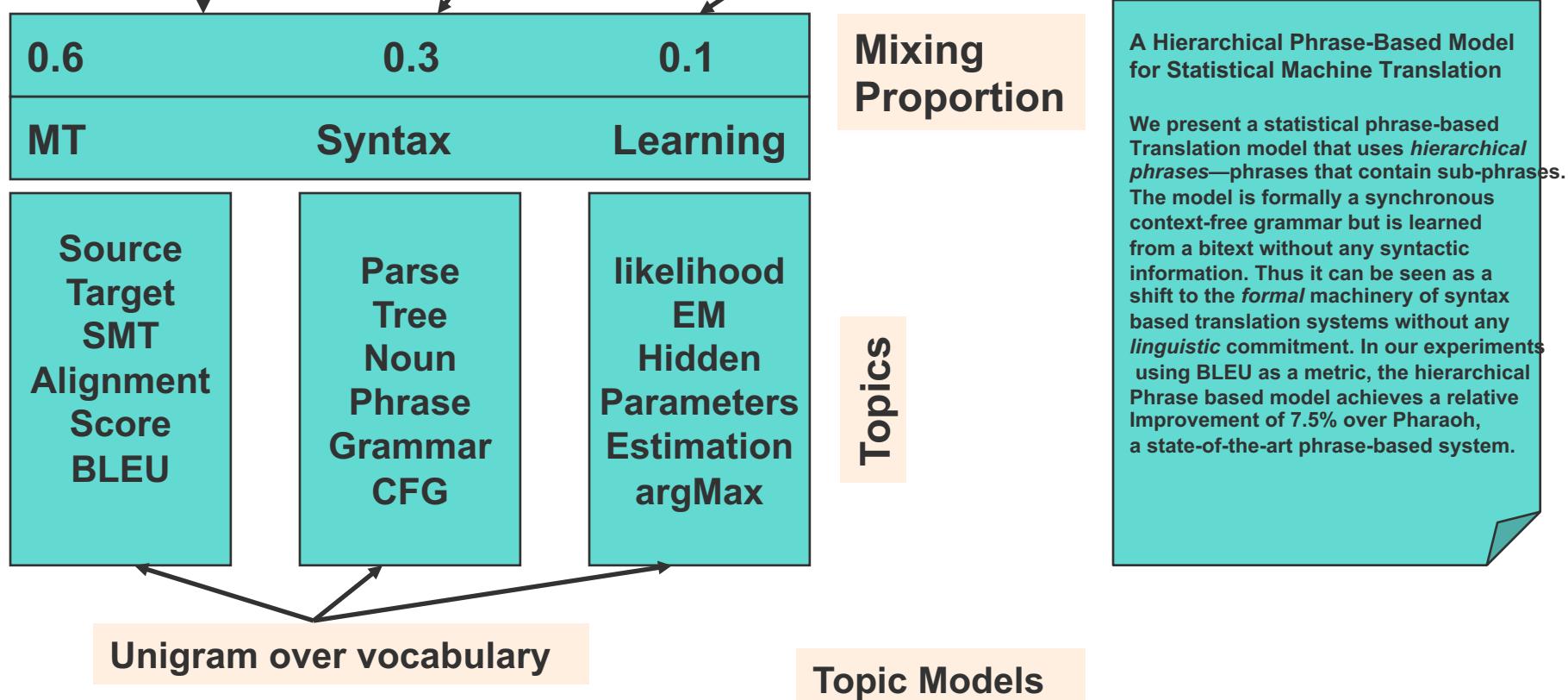
- Each document is a vector in the word space
- Ignore the order of words in a document. Only count matters!
- A high-dimensional and sparse representation
 - Not efficient text processing tasks, e.g., search, document classification, or similarity measure $(|V| \gg D)$
 - Not effective for browsing





How to Model Semantic?

- Q: What is it about?
- A: Mainly MT, with syntax, some learning





Why this is Useful?

- Q: What is it about?
- A: Mainly MT, with syntax, some learning



- **Q: give me similar document?**
 - Structured way of browsing the collection
- **Other tasks**
 - Dimensionality reduction
 - TF-IDF vs. topic mixing proportion
 - Classification, clustering, and more ...

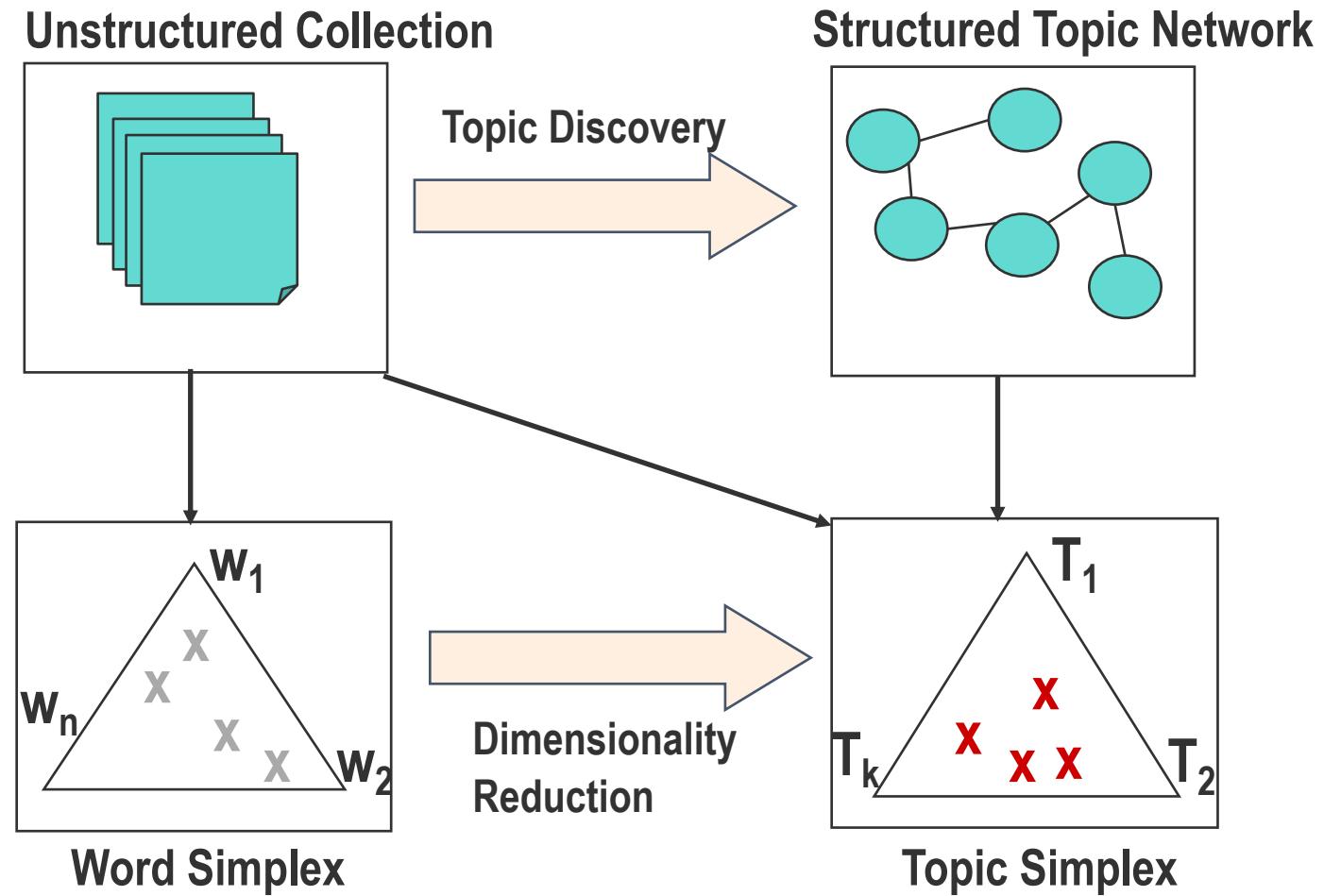
A Hierarchical Phrase-Based Model for Statistical Machine Translation

We present a statistical phrase-based Translation model that uses *hierarchical phrases*—phrases that contain sub-phrases. The model is formally a synchronous context-free grammar but is learned from a bitext without any syntactic information. Thus it can be seen as a shift to the *formal* machinery of syntax based translation systems without any *linguistic* commitment. In our experiments using BLEU as a metric, the hierarchical Phrase based model achieves a relative Improvement of 7.5% over Pharaoh, a state-of-the-art phrase-based system.



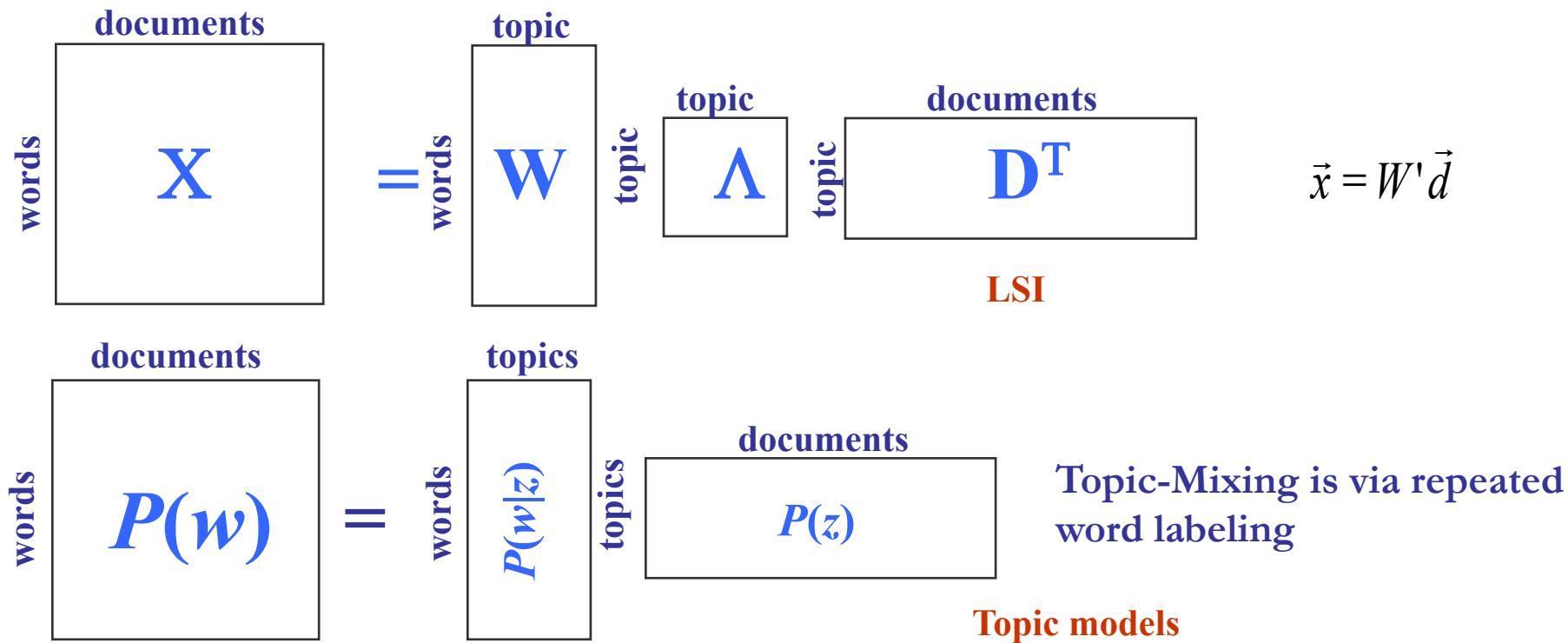


Topic Models: The Big Picture





LSI versus Topic Model (probabilistic LSI)





Words in Contexts

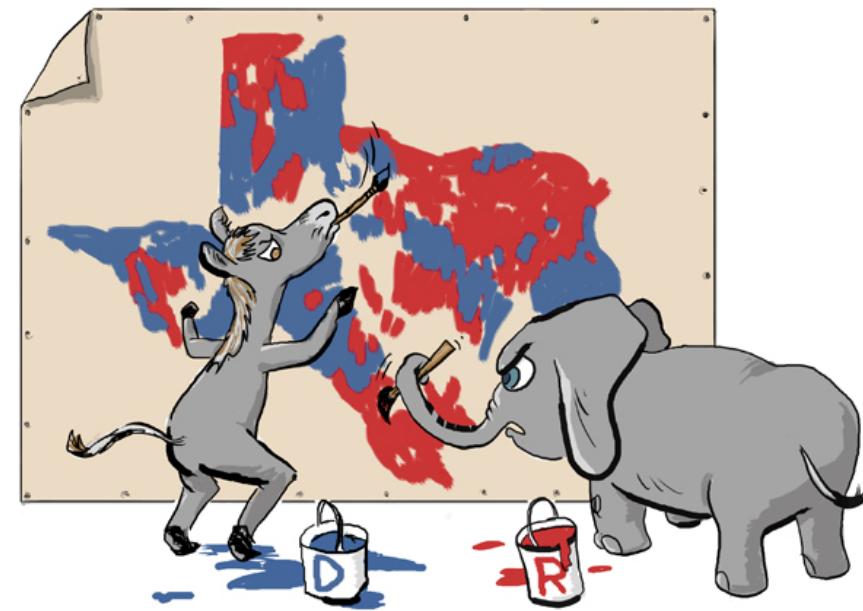
- “It was a nice **shot**. ”





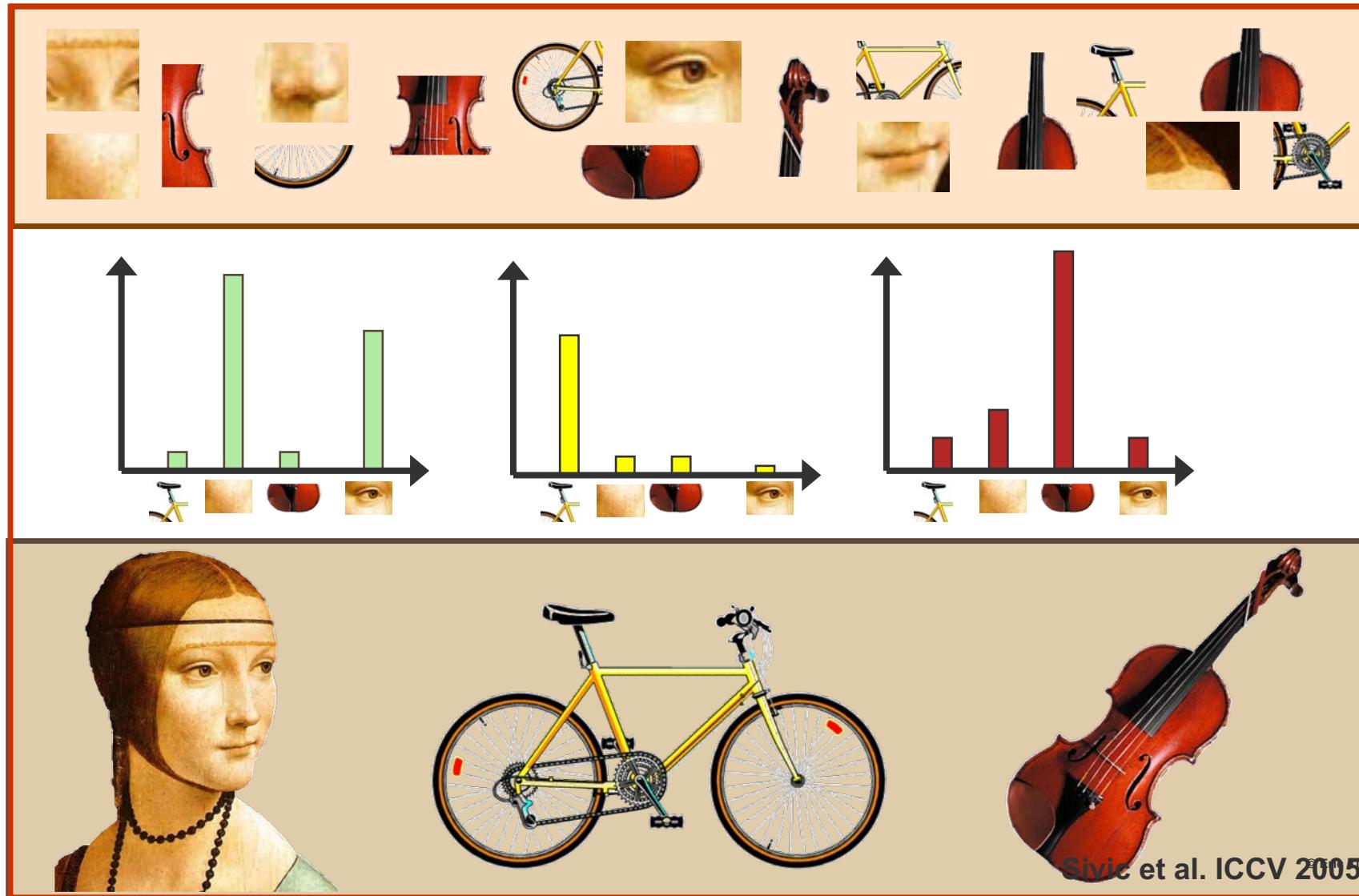
Words in Contexts (con'd)

- The opposition Labor Party fared even worse, with a predicted 35 **seats**, seven less than last election.





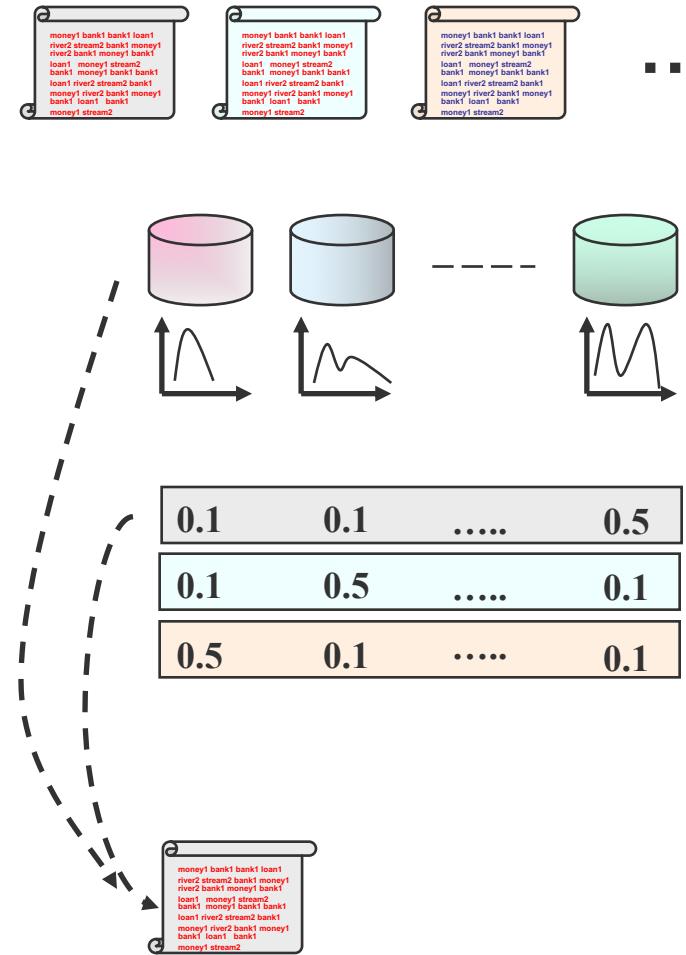
"Words" in Contexts (con'd)





Admixture Models

- Objects are **bags** of elements
- Mixtures are **distributions** over elements
- Objects have **mixing vector** θ
 - Represents each mixtures' contributions
- Object is **generated** as follows:
 - Pick a mixture component from θ
 - Pick an **element** from that component





Topic Models

Generating a document

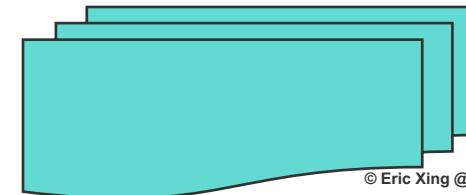
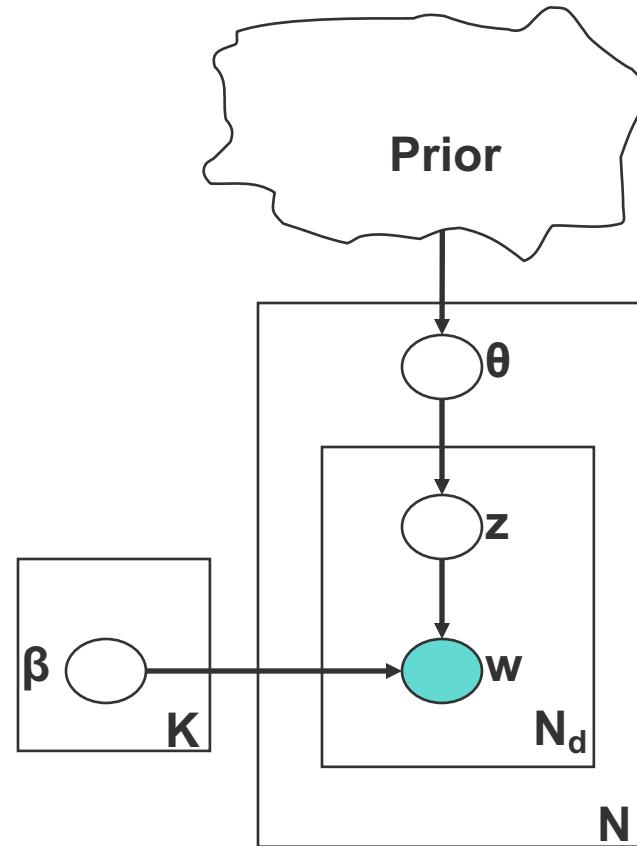
– Draw θ from the prior

For each word n

– Draw z_n from $multinomial(\theta)$

– Draw $w_n | z_n, \{\beta_{1:k}\}$ from $multinomial(\beta_{z_n})$

Which prior to use?

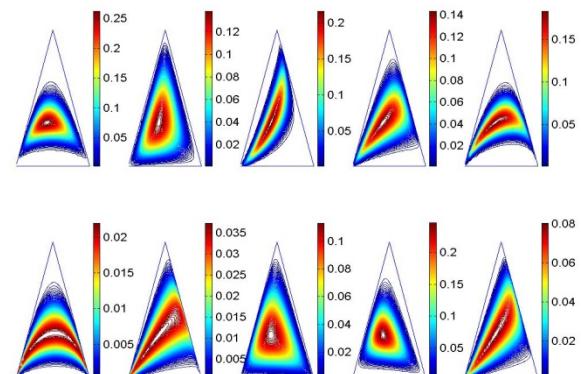
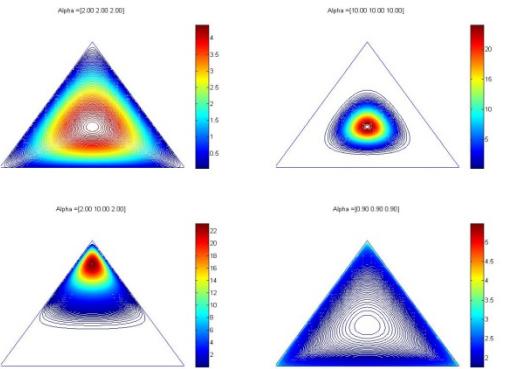




Choices of Priors

- Dirichlet (LDA) (Blei et al. 2003)
 - Conjugate prior means efficient inference
 - Can only capture variations in each topic's intensity independently

- Logistic Normal (CTM=LoNTAM) (Blei & Lafferty 2005, Ahmed & Xing 2006)
 - Capture the intuition that some topics are highly correlated and can rise up in intensity together
 - Not a conjugate prior implies hard inference





Generative Semantic of LoNTAM

Generating a document

- Draw θ from the prior

For each word n

- Draw z_n from $multinomial(\theta)$

- Draw $w_n | z_n, \{\beta_{1:k}\}$ from $multinomial(\beta_{z_n})$

$$\theta \sim LN_K(\mu, \Sigma)$$

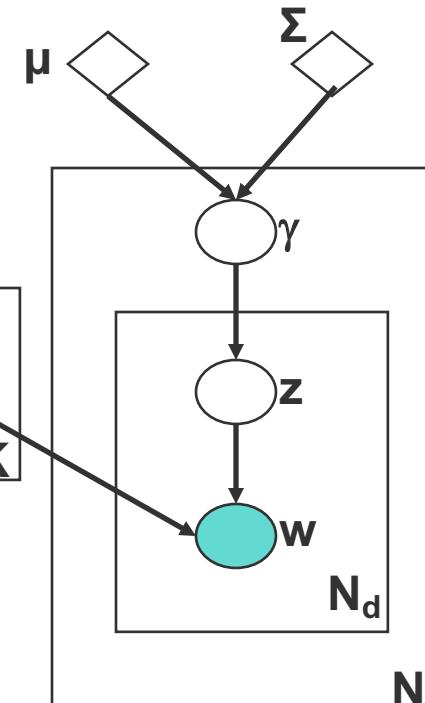
$$\gamma \sim N_{K-1}(\mu, \Sigma)$$

$$\gamma_K = 0$$

$$\theta_i = \exp\left\{ \gamma_i - \log\left(1 + \sum_{i=1}^{K-1} e^{\gamma_i} \right) \right\}$$

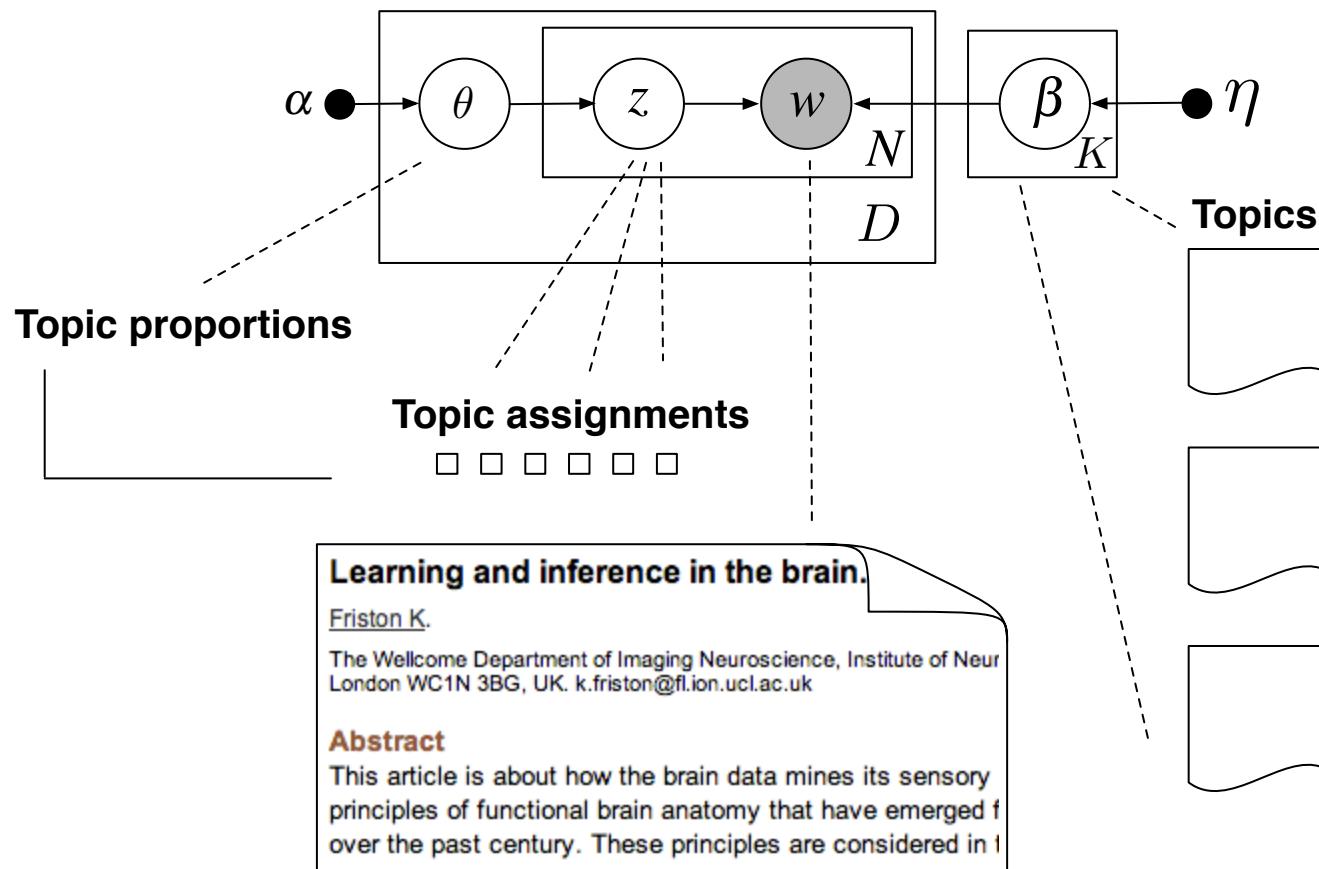
$$C(\gamma) = \log\left(1 + \sum_{i=1}^{K-1} e^{\gamma_i} \right)$$

- Problem**
- Log Partition Function
 - Normalization Constant



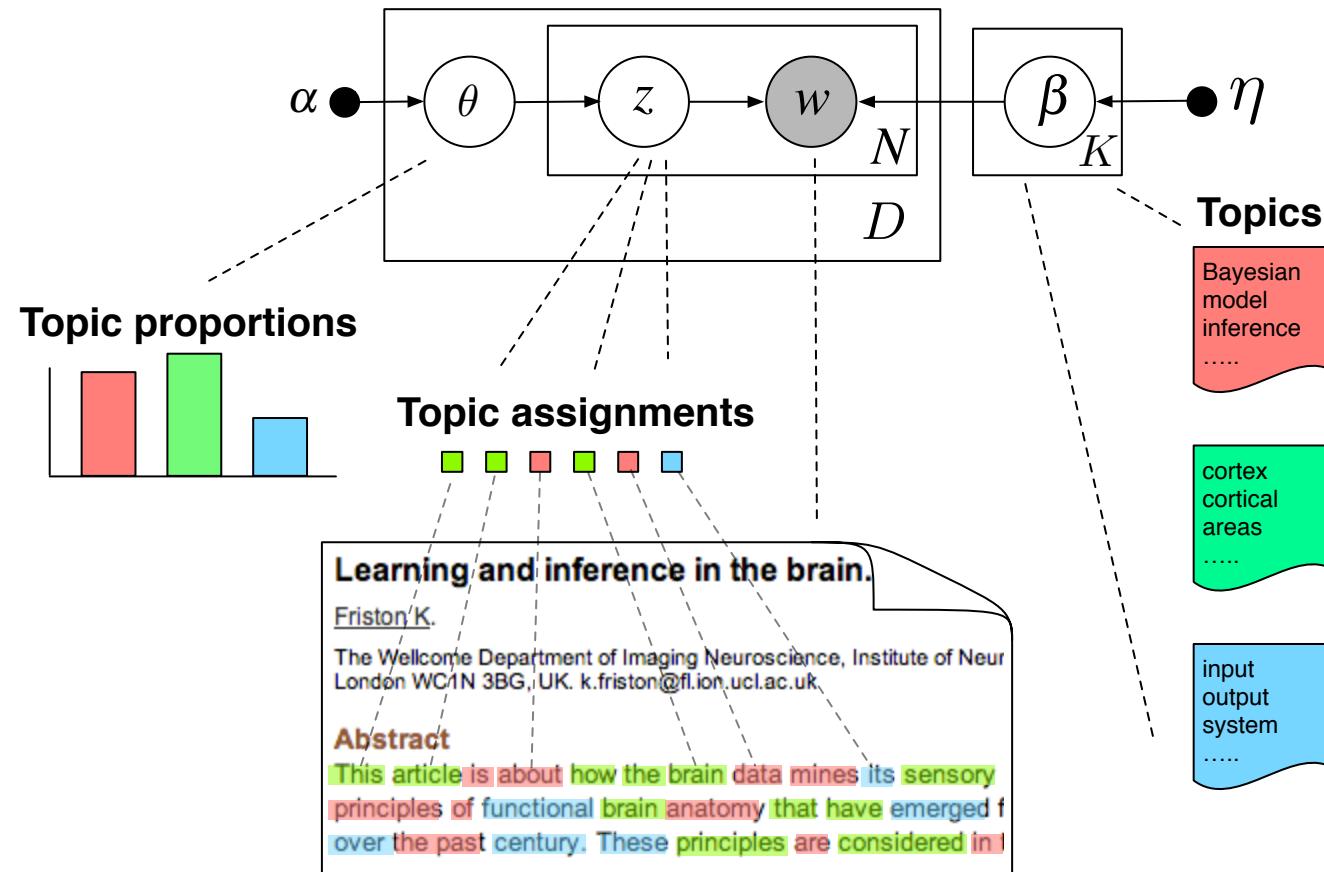


Posterior inference





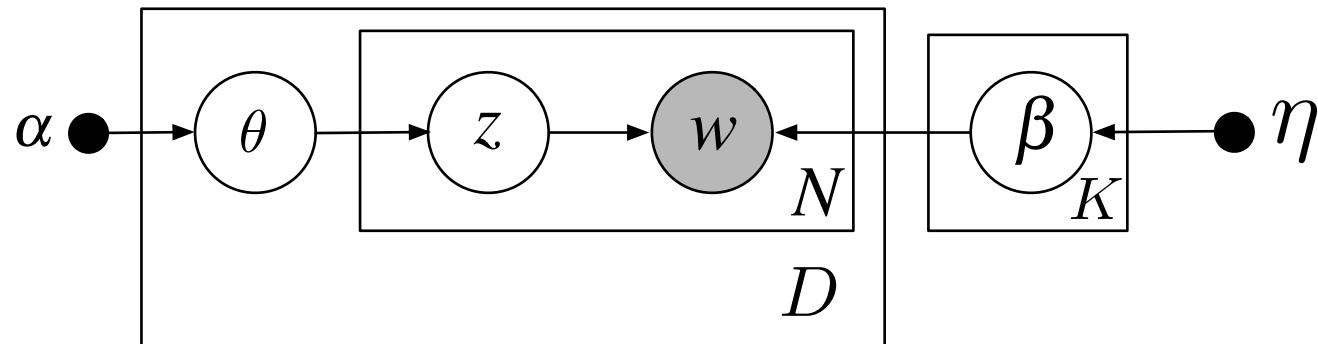
Posterior inference results





Joint likelihood of all variables

$$p(\beta, \theta, z, w) = \prod_{k=1}^K p(\beta_k | \eta) \prod_{d=1}^D p(\theta_d | \alpha) \prod_{n=1}^N p(z_{dn} | \theta_d) p(w_{dn} | z_{dn}, \beta)$$



We are interested in computing the posterior,
and the data likelihood!





Inference and Learning are both intractable

- ❑ A possible query:

$$p(\theta_n | D) = ?$$

$$p(z_{n,m} | D) = ?$$

- ❑ Close form solution?

$$p(\theta_n | D) = \frac{p(\theta_n, D)}{p(D)}$$
$$= \frac{\sum_{\{z_{n,m}\}} \int \left(\prod_n \left(\prod_m p(w_{n,m} | \beta_{z_n}) p(z_{n,m} | \theta_n) \right) p(\theta_n | \alpha) \right) p(\beta | \eta) d\theta_{-i} d\beta}{p(D)}$$

$$p(D) = \sum_{\{z_{n,m}\}} \int \cdots \int \left(\prod_n \left(\prod_m p(x_{n,m} | \beta_{z_n}) p(z_{n,m} | \theta_n) \right) p(\theta_n | \alpha) \right) p(\beta | \eta) d\theta_1 \cdots d\theta_N d\beta$$

- ❑ Sum in the denominator over T^n terms, and integrate over n k -dimensional topic vectors
- ❑ Learning: What to learn? What is the objective function?





Approximate Inference

- Variational Inference
 - Mean field approximation (Blei et al.)
 - Expectation propagation (Minka et al.)
 - Variational 2nd-order Taylor approximation (Xing)

- Markov Chain Monte Carlo
 - Gibbs sampling (Griffiths et al)





Variational Inference

- Consider a generative model $p_{\theta}(\mathbf{x}|\mathbf{z})$, and prior $p(\mathbf{z})$
 - Joint distribution: $p_{\theta}(\mathbf{x}, \mathbf{z}) = p_{\theta}(\mathbf{x}|\mathbf{z})p(\mathbf{z})$
- Assume **variational distribution** $q_{\phi}(\mathbf{z}|\mathbf{x})$
- Objective: Maximize **lower bound** for log likelihood

$$\begin{aligned}& \log p(\mathbf{x}) \\&= KL\left(q_{\phi}(\mathbf{z}|\mathbf{x}) \parallel p_{\theta}(\mathbf{z}|\mathbf{x})\right) + \int_{\mathbf{z}} q_{\phi}(\mathbf{z}|\mathbf{x}) \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q_{\phi}(\mathbf{z}|\mathbf{x})} \\&\geq \int_{\mathbf{z}} q_{\phi}(\mathbf{z}|\mathbf{x}) \log \frac{p_{\theta}(\mathbf{x}, \mathbf{z})}{q_{\phi}(\mathbf{z}|\mathbf{x})} \\&:= \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x})\end{aligned}$$

- Equivalently, minimize **free energy**

$$F(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = -\log p(\mathbf{x}) + KL(q_{\phi}(\mathbf{z}|\mathbf{x}) \parallel p_{\theta}(\mathbf{z}|\mathbf{x}))$$





Variational Inference

Maximize the variational lower bound:

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(z|x)}[\log p_{\boldsymbol{\theta}}(x|z)] + KL(q_{\boldsymbol{\phi}}(z|x) || p(z))$$

- **E-step:** maximize \mathcal{L} w.r.t. $\boldsymbol{\phi}$, with $\boldsymbol{\theta}$ fixed

$$\max_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x})$$

- If closed form solutions exist:

$$q_{\boldsymbol{\phi}}^*(z|x) \propto \exp[\log p_{\boldsymbol{\theta}}(x, z)]$$

- **M-step:** maximize \mathcal{L} w.r.t. $\boldsymbol{\theta}$, with $\boldsymbol{\phi}$ fixed

$$\max_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x})$$





Mean-field assumption (in topic models)

- ❑ True posterior

$$p(\beta, \theta, z | \mathbf{w}) = \frac{p(\beta, \theta, z, \mathbf{w})}{p(\mathbf{w})}$$

- ❑ Break the dependency using the **fully factorized** distribution

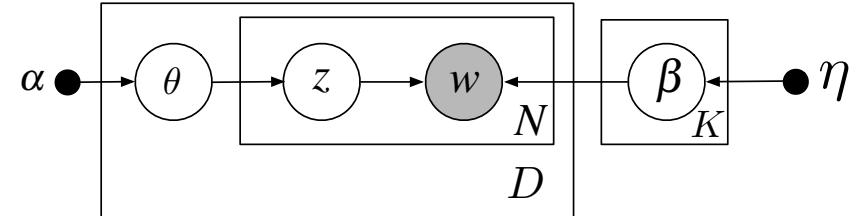
$$q(\beta, \theta, z) = \prod_k q(\beta_k) \prod_d q(\theta_d) \prod_n q(z_{dn})$$

- ❑ Mean-field family usually does NOT include the true posterior.





Mean Field Approximation

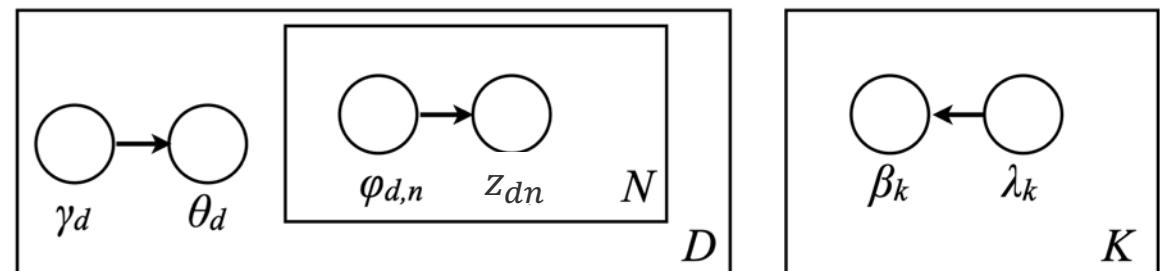


- Parametric form for each marginal factor in $q(\beta, z, \theta | \lambda, \phi, \gamma)$:

$$q(\beta_k | \lambda_k) = \text{Dirichlet}(\beta_k | \lambda_k)$$

$$q(\theta_d | \gamma_d) = \text{Dirichlet}(\theta_d | \gamma_d)$$

$$q(z_{dn} | \phi_{dn}) = \text{Multinomial}(z_{dn} | \phi_{dn})$$



- Learning parameters of the variational distribution (E-step):

$$\gamma^*, \lambda^*, \phi^* = \arg \min_{\gamma, \lambda, \phi} \text{KL}(q(\beta, \theta, \mathbf{z} | \gamma, \phi) \| p(\beta, \theta, \mathbf{z} | \mathbf{w}, \alpha, \eta))$$

- For LDA, we can compute the optimal MF approximation in closed form.





Update each marginal

- Update:

$$q(\theta_d) \propto \exp \left\{ \mathbb{E}_{\prod_n q(z_{dn})} \left[\log p(\theta_d | \alpha) + \sum_n \log p(z_{dn} | \theta_d) \right] \right\}$$

- Where in LDA:

$$p(\theta_d | \alpha) \propto \exp \left\{ \sum_{k=1}^K (\alpha_k - 1) \log \theta_{dk} \right\} \text{ -- Dirichlet}$$

$$p(z_{dn} | \theta_d) = \exp \left\{ \sum_{k=1}^K 1[z_{dn} = k] \log \theta_{dk} \right\} \text{ -- Multinomial}$$

- And we obtain:

$$q(\theta_d) \propto \exp \left\{ \sum_{k=1}^K \left(\sum_{n=1}^N q(z_{dn} = k) + \alpha_k - 1 \right) \log \theta_{dk} \right\}$$

This is also a Dirichlet — the same as its prior!





Update each marginal

- Similarly to $q(\theta_d \mid \gamma_d)$, we obtain optimal parameters ϕ_{dn}^* for $q(z_{dn} \mid \phi_{dn})$:

$$q(z_{dn} = k \mid \phi_{dn}) = \phi_{dn}(k) = \beta_k(w_{dn}) \exp \left\{ \Psi(\gamma_d(k)) - \Psi\left(\sum_{j=1}^K \gamma_d(j)\right) \right\}$$

- And optimal parameters λ_k^* for $q(\beta_k \mid \lambda_k)$:

$$\lambda_k(j) = \eta(j) + \sum_{d=1}^D \sum_{n=1}^{N_d} \phi_{dn}^*(k) \mathbf{1}[w_{dn} = j]$$

- Iterating these equations to convergence yields the MF approximation to the posterior distribution.





Coordinate ascent algorithm for LDA

- 1: Initialize variational topics $q(\beta_k)$, $k = 1, \dots, K$.
- 2: **repeat**
- 3: **for** each document $d \in \{1, 2, \dots, D\}$ **do**
- 4: Initialize variational topic assignments $q(z_{dn})$, $n = 1, \dots, N$
- 5: **repeat**
- 6: Update variational topic proportions $q(\theta_d)$
- 7: Update variational topic assignments $q(z_{dn})$, $n = 1, \dots, N$
- 8: **until** Change of $q(\theta_d)$ is small enough
- 9: **end for**
- 10: Update variational topics $q(\beta_k)$, $k = 1, \dots, K$.
- 11: **until** Lower bound $L(q)$ converges





Conclusion

- ❑ GM-based topic models are cool
 - ❑ Flexible
 - ❑ Modular
 - ❑ Interactive
- ❑ There are many ways of implementing topic models
 - ❑ unsupervised
 - ❑ supervised
- ❑ Efficient Inference/learning algorithms
 - ❑ GMF, with Laplace approx. for non-conjugate dist.
 - ❑ MCMC
- ❑ Many applications
 - ❑ ...
 - ❑ Word-sense disambiguation
 - ❑ Image understanding
 - ❑ Network inference







Supplementary: More on strategies in VI

- Alternative approximation scheme
- How to evaluate: empirical (ground truth unknown) vs. simulation (ground truth known)
- Comparison (of what)
- Building blocks

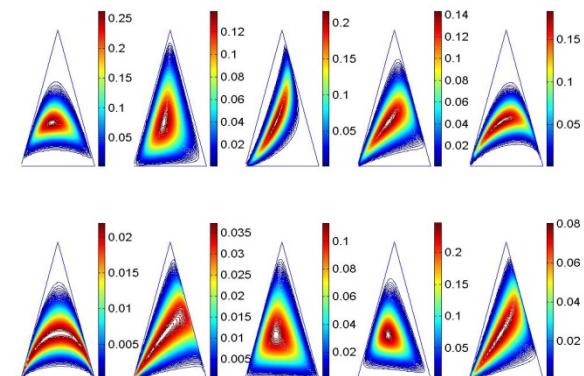
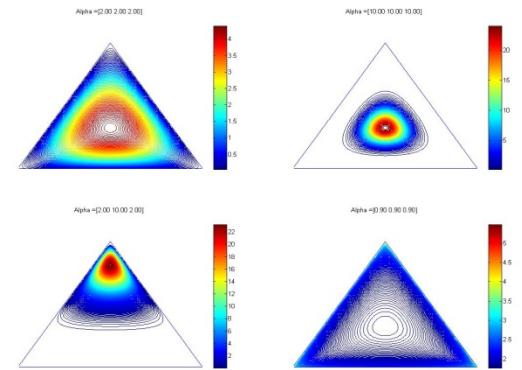




Recall Choices of Priors

- Dirichlet (LDA) (Blei et al. 2003)
 - Conjugate prior means efficient inference
 - Can only capture variations in each topic's intensity independently

- Logistic Normal (CTM=LoNTAM) (Blei & Lafferty 2005, Ahmed & Xing 2006)
 - Capture the intuition that some topics are highly correlated and can rise up in intensity together
 - Not a conjugate prior implies hard inference





Choice of $q()$ does matter



Σ^* is full matrix

Multivariate
Quadratic Approx.

Closed Form
Solution for μ^*, Σ^*

Log Partition Function

$$\log \left(1 + \sum_{i=1}^{K-1} e^{\gamma_i} \right)$$

Σ^* is assumed to be diagonal

Tangent Approx.

Numerical
Optimization to
fit μ^* , Diag(Σ^*)

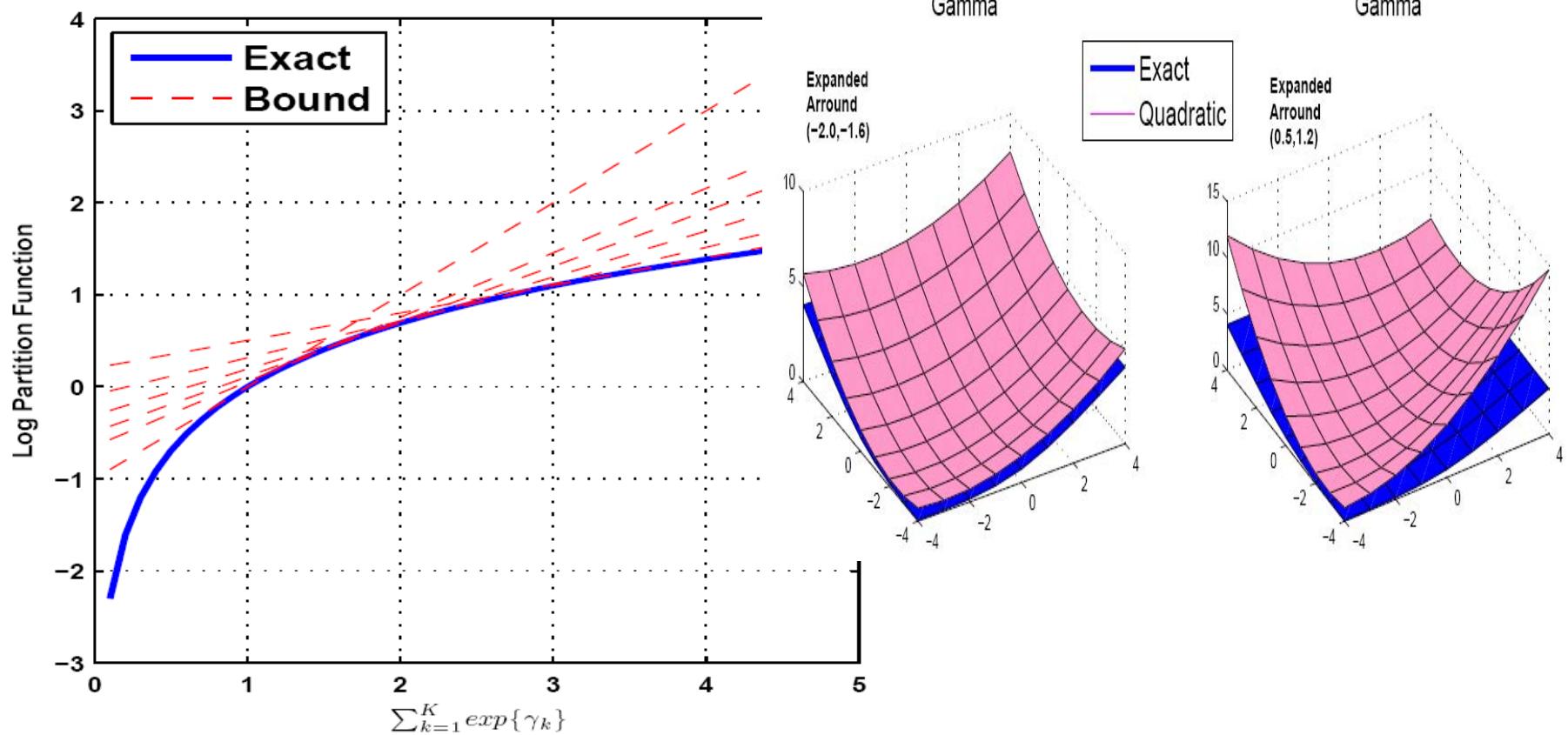
Ahmed&Xing

Blei&Lafferty





Tangent Approximation





How to evaluate?

- Empirical Visualization: e.g., topic discovery on New York Times

The 5 most frequent topics from the HDP on the *New York Times*.

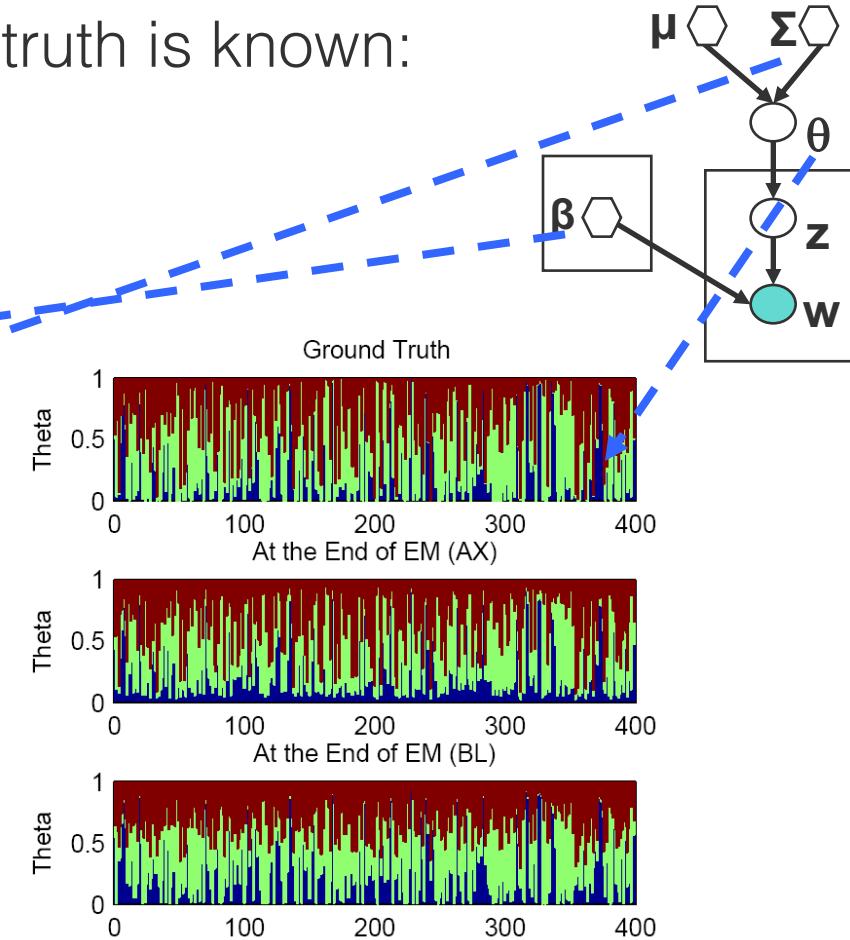
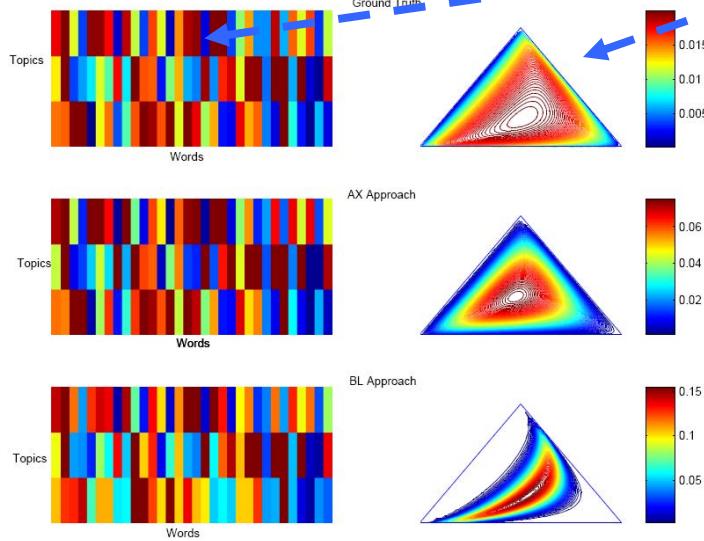
game season team coach play points games giants second players	life know school street man family says house children night	film movie show life television films director man story says	book life books novel story man author house war children	wine street hotel house room night place restaurant park garden
---	---	--	--	--





How to evaluate?

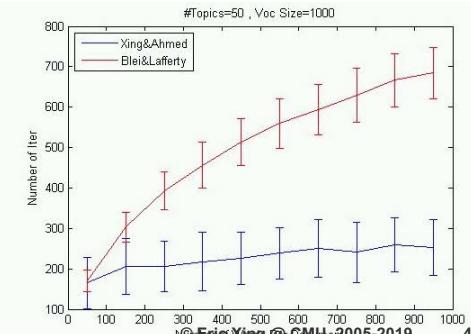
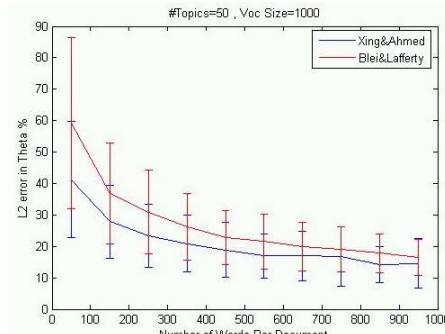
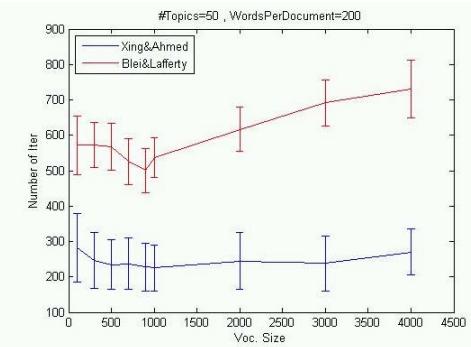
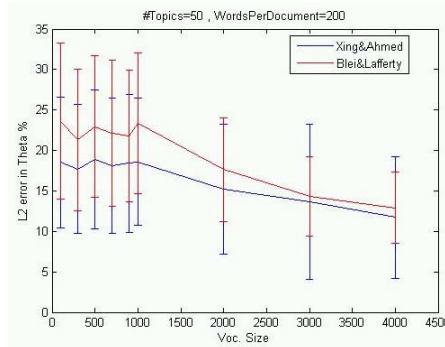
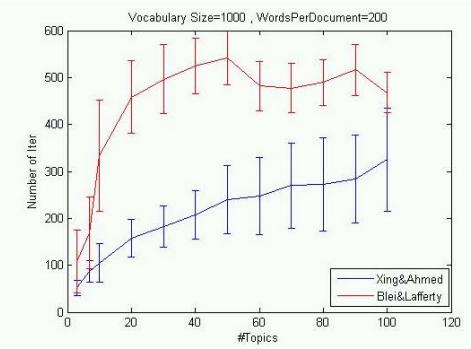
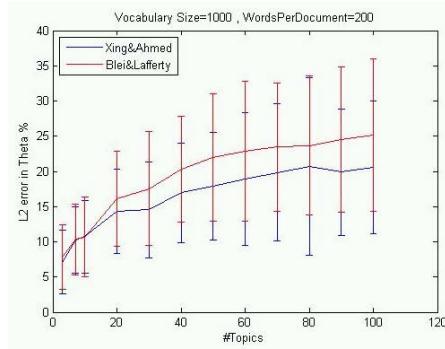
- Test on Synthetic Text where ground truth is known:





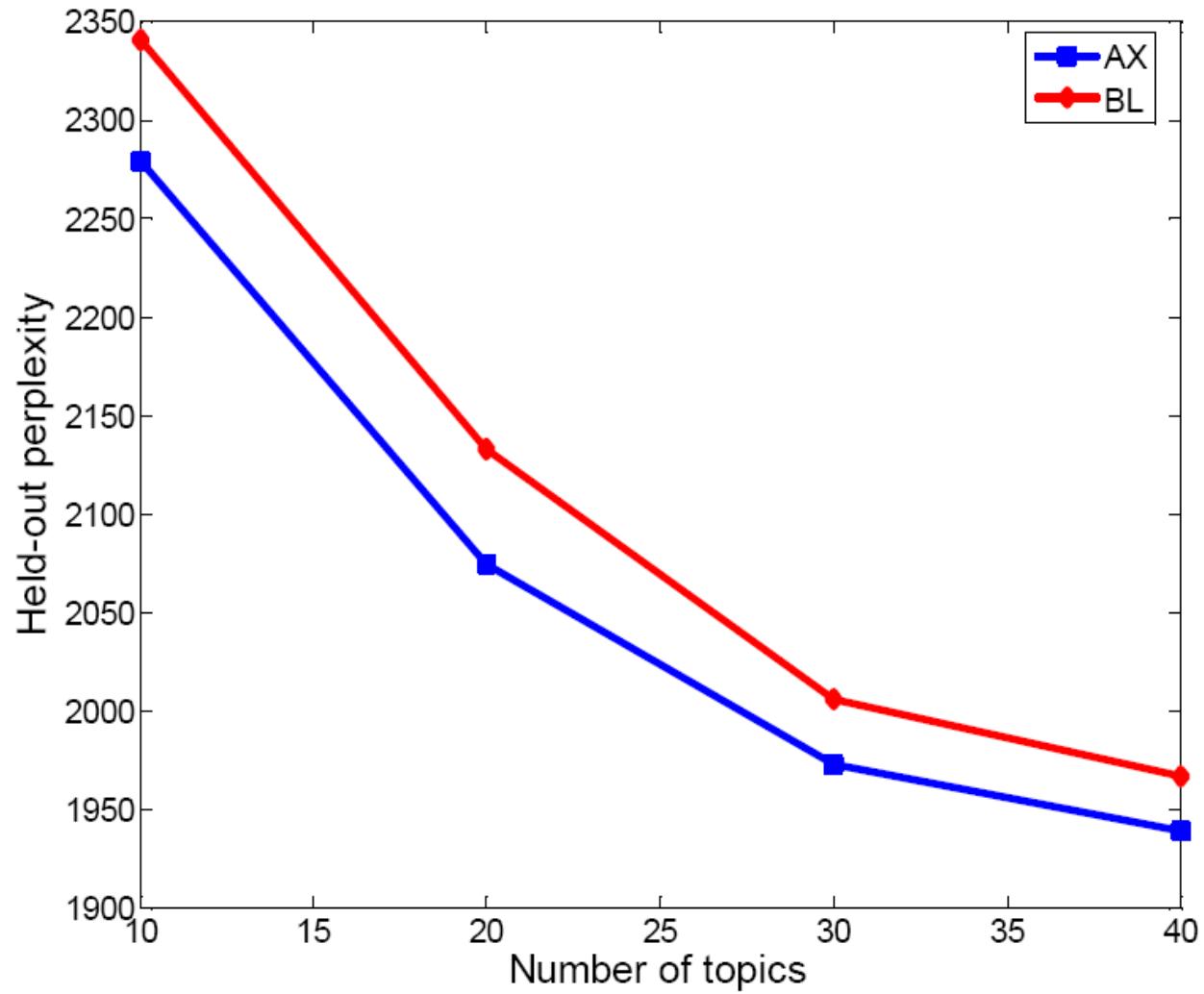
Comparison: accuracy and speed

- ❑ L2 error in topic vector est. and # of iterations
- ❑ Varying Num. of Topics
- ❑ Varying Voc. Size
- ❑ Varying Num. Words Per Document





Comparison: perplexity





Classification Result on PNAS collection

- ❑ PNAS abstracts from 1997-2002
 - ❑ 2500 documents
 - ❑ Average of 170 words per document
- ❑ Fitted 40-topics model using both approaches
- ❑ Use low dimensional representation to predict the abstract category
 - ❑ Use SVM classifier
 - ❑ 85% for training and 15% for testing

Classification Accuracy

Category	Doc	BL	AX
Genetics	21	61.9	61.9
Biochemistry	86	65.1	77.9
Immunology	24	70.8	66.6
Biophysics	15	53.3	66.6
Total	146	64.3	72.6

-Notable Difference
-Examine the low dimensional representations below

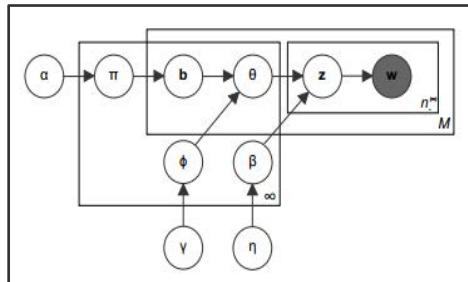




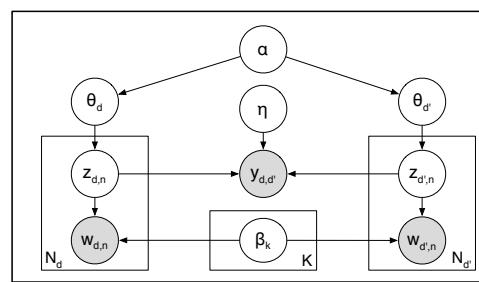
What makes topic models useful --- The Zoo of Topic Models!

- It is a building block of many models.

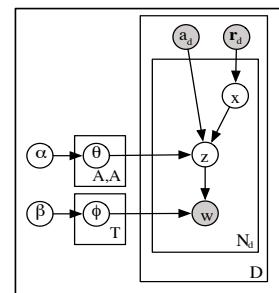
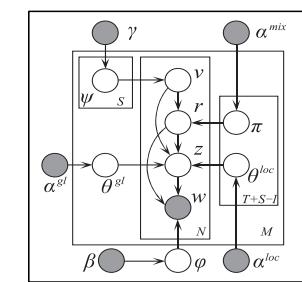
Williamson et al. 2010



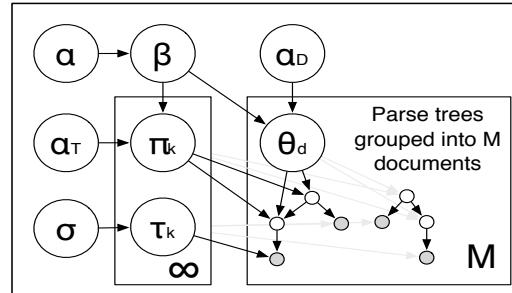
Chang & Blei, 2009



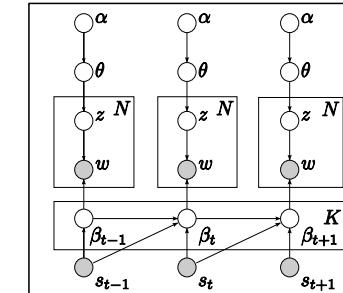
Titov & McDonald, 2008



McCallum et al. 2007



Boyd-Graber & Blei, 2008



Wang & Blei, 2008



More on Mean Field Approximation



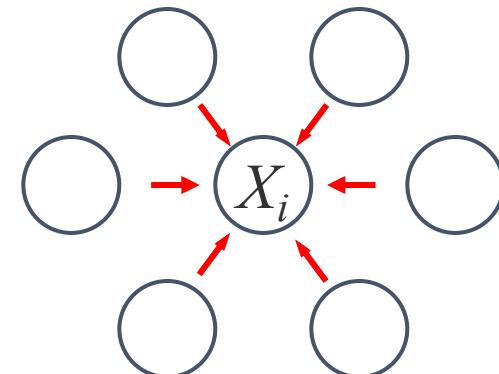


The naive mean field approximation

- Approximate $p(\mathbf{X})$ by fully factorized $q(\mathbf{X}) = \prod_i q_i(X_i)$
- For Boltzmann distribution $p(X) = \exp\{\sum_{i < j} q_{ij} X_i X_j + q_{io} X_i\}/Z$:

mean field equation:

$$\begin{aligned} q_i(X_i) &= \exp\left\{\theta_{i0} X_i + \sum_{j \in \mathcal{N}_i} \theta_{ij} X_i \langle X_j \rangle_{q_j} + A_i\right\} \\ &= p(X_i | \{\langle X_j \rangle_{q_j} : j \in \mathcal{N}_i\}) \end{aligned}$$



- $\langle X_j \rangle_{q_j}$ resembles a “message” sent from node j to i
- $\{\langle X_j \rangle_{q_j} : j \in \mathcal{N}_i\}$ forms the “mean field” applied to X_i from its neighborhood



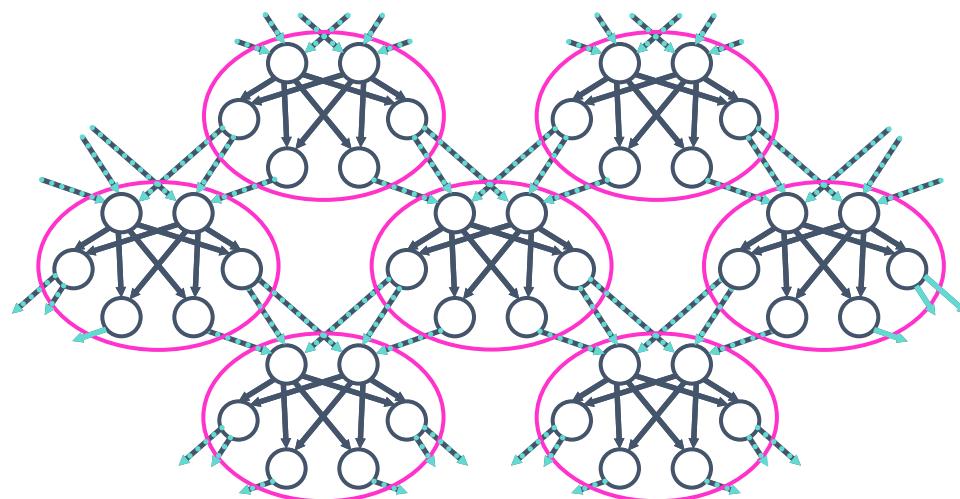


Cluster-based approx. to the Gibbs free energy

(Wiegerinck 2001,
Xing et al 03,04)

Exact: $G[p(X)]$ (*intractable*)

Clusters: $G[\{q_c(X_c)\}]$





Mean field approx. to Gibbs free energy

- Given a disjoint clustering, $\{C_1, \dots, C_J\}$, of all variables

- Let

$$q(\mathbf{X}) = \prod_i q_i(\mathbf{x}_{C_i}),$$

- Mean-field free energy

$$G_{\text{MF}} = \sum_i \sum_{\mathbf{x}_{C_i}} \prod_i q_i(\mathbf{x}_{C_i}) E(\mathbf{x}_{C_i}) + \sum_i \sum_{\mathbf{x}_{C_i}} q_i(\mathbf{x}_{C_i}) \ln q_i(\mathbf{x}_{C_i})$$

e.g., $G_{\text{MF}} = \sum_{i < j} \sum_{x_i x_j} q(x_i) q(x_j) \phi(x_i x_j) + \sum_i \sum_{x_i} q(x_i) \phi(x_i) + \sum_i \sum_{x_i} q(x_i) \ln q(x_i)$ (naïve mean field)

- Will **never** equal to the exact Gibbs free energy no matter what clustering is used, but it does **always** define a lower bound of the likelihood
- Optimize each $q_i(x_c)$'s.
 - Variational calculus ...
 - Do inference in each $q_i(x_c)$ using any tractable algorithm





The Generalized Mean Field theorem

Theorem: The optimum GMF approximation to the cluster marginal is isomorphic to the cluster posterior of the original distribution given internal evidence and its generalized mean fields:

$$q_i^*(\mathbf{X}_{H,C_i}) = p(\mathbf{X}_{H,C_i} \mid \mathbf{x}_{E,C_i}, \langle \mathbf{X}_{H,MB_i} \rangle_{q_{j \neq i}})$$

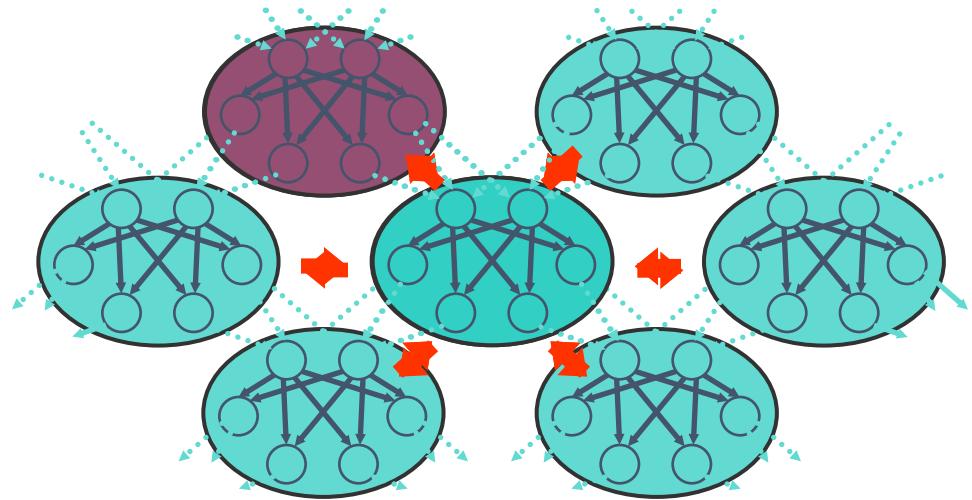
GMF algorithm: Iterate over each q_i





A generalized mean field algorithm

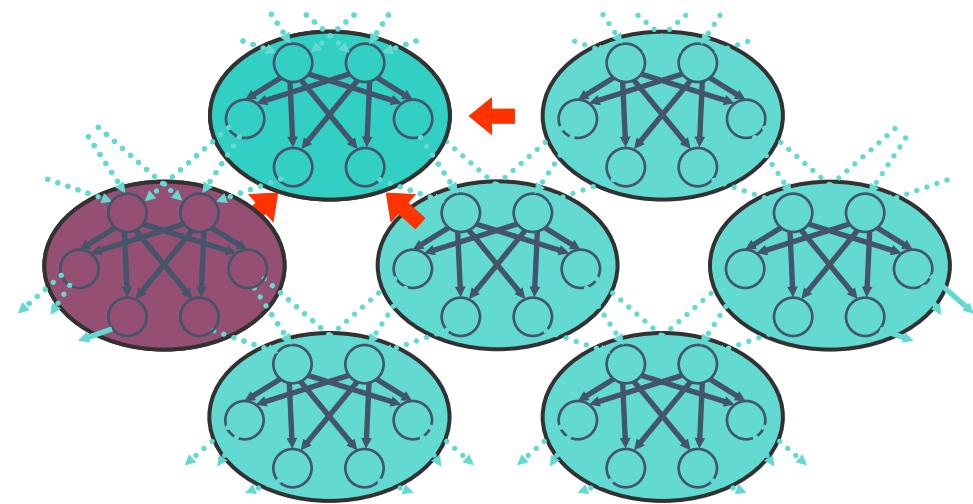
[xing et al. UAI 2003]





A generalized mean field algorithm

[Xing et al., UAI 2009]





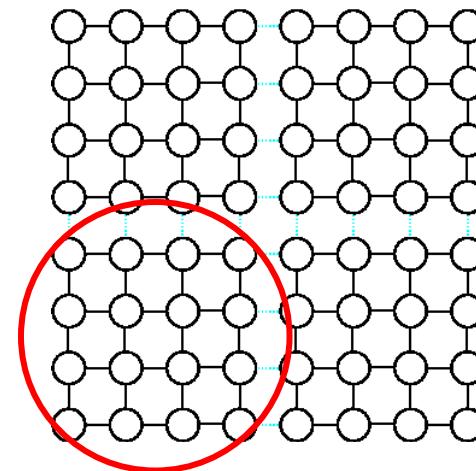
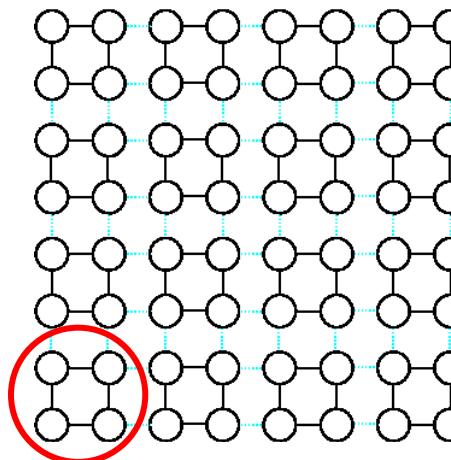
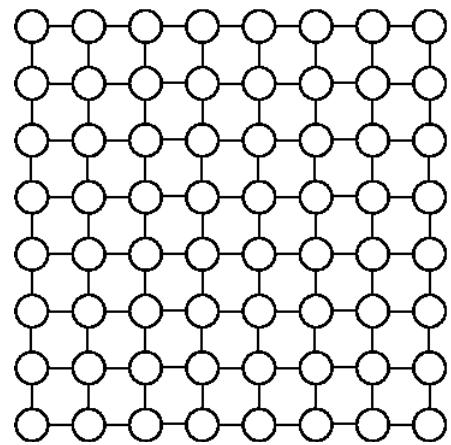
Convergence theorem

Theorem: The GMF algorithm is guaranteed to converge to a local optimum, and provides a lower bound for the likelihood of evidence (or partition function) the model.





Example 1: Generalized MF approximations to Ising models



Cluster marginal of a square block C_k :

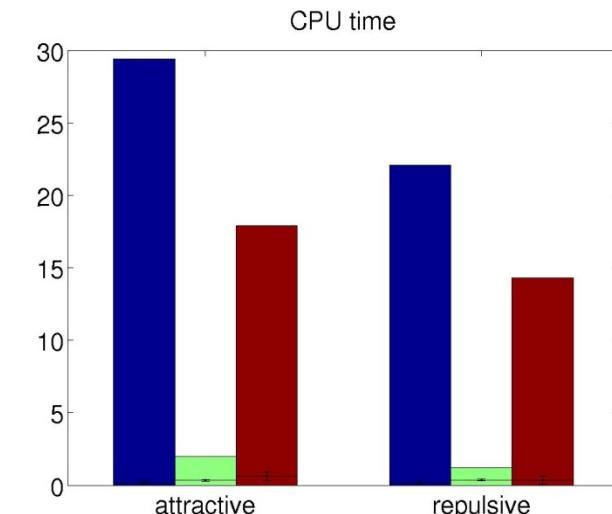
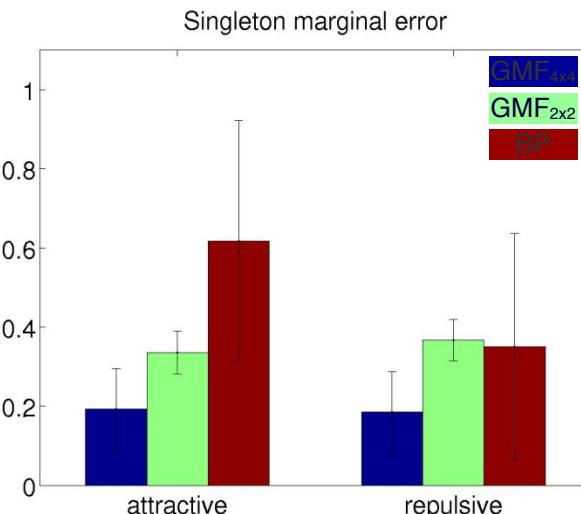
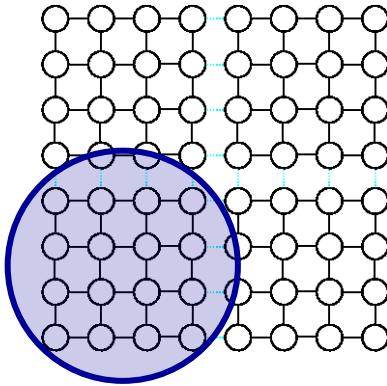
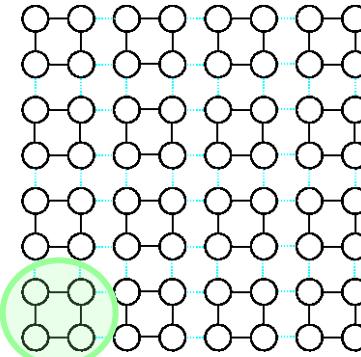
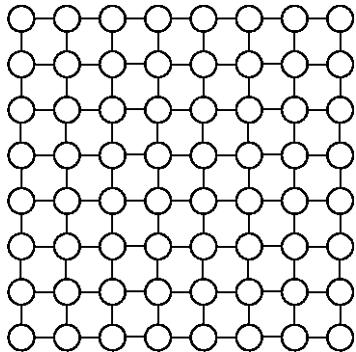
$$q(X_{C_k}) \propto \exp \left\{ \sum_{i,j \in C_k} \theta_{ij} X_i X_j + \sum_{i \in C_k} \theta_{i0} X_i + \sum_{\substack{i \in C_k, j \in MB_k, \\ k' \in MBC_k}} \theta_{ij} X_i \langle X_j \rangle_{q(X_{C_k'})} \right\}$$

Virtually a reparameterized Ising model of small size.





GMF approximation to Ising models

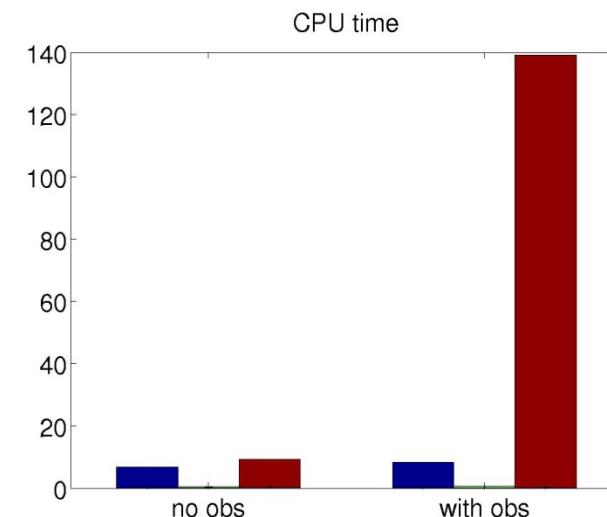
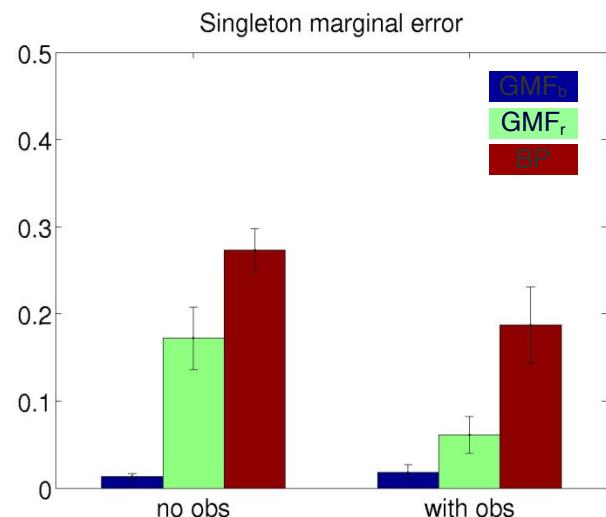
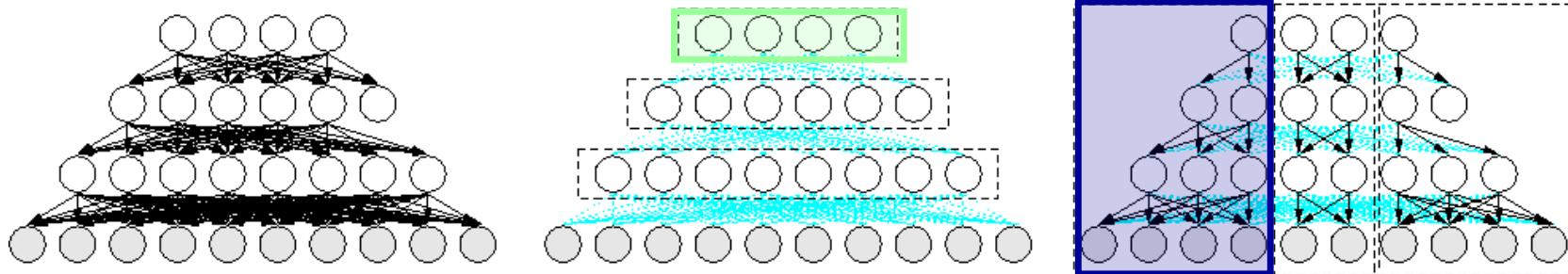


Attractive coupling: positively weighted
Repulsive coupling: negatively weighted





Example 2: Sigmoid belief network





Example 3: Factorial HMM

