Scalable Machine Learning

9. Graphical Models

Alex Smola
Yahoo! Research and ANU

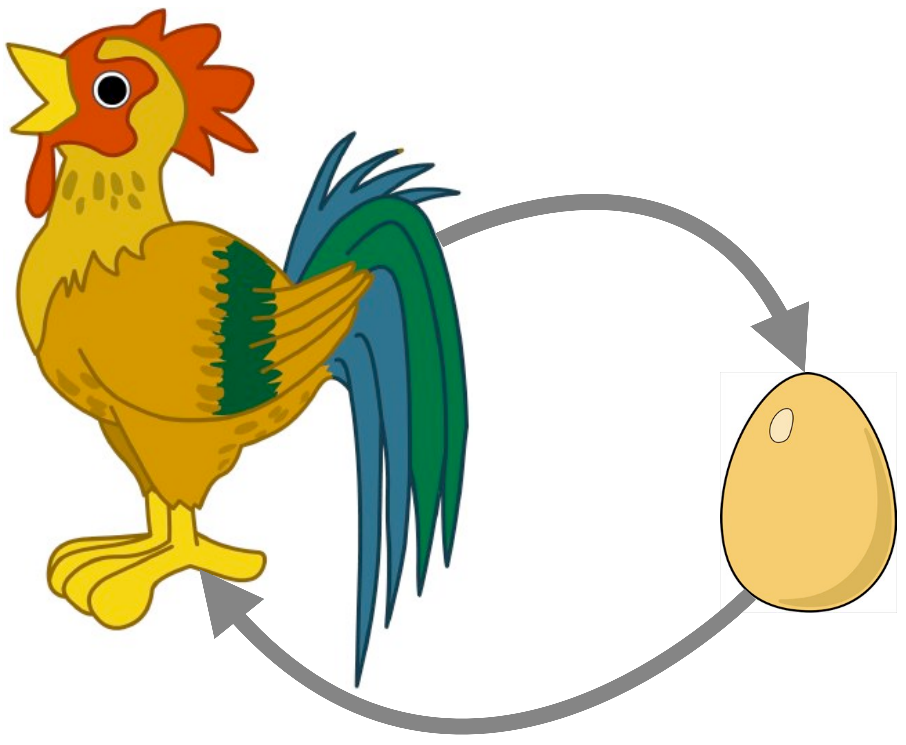
http://alex.smola.org/teaching/berkeley2012
Stat 260 SP 12

Significant content courtesy of Yehuda Koren

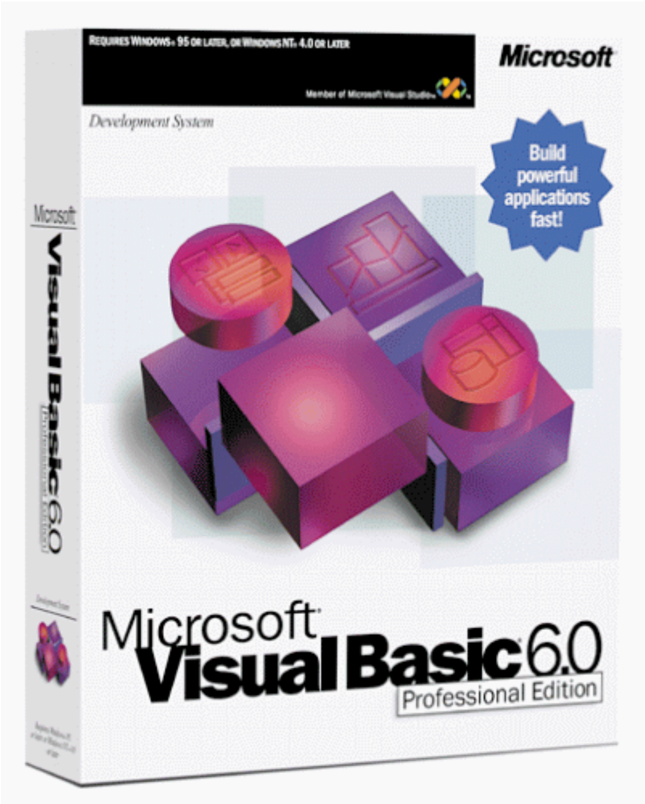
Outline

- Directed Graphical Models
  - Dependence
  - Inference for fully observed models
  - Incomplete information / variational and sampling inference
- Undirected Graphical Models
  - Hammersley Clifford decomposition
  - Conditional independence
  - Junction trees
- Dynamic Programming
  - Generalized Distributive Law
  - Naive Message Passing
- Inference techniques
  - Sampling (Gibbs and Monte Carlo)
  - Variational methods (EM, extensions, dual decomposition)

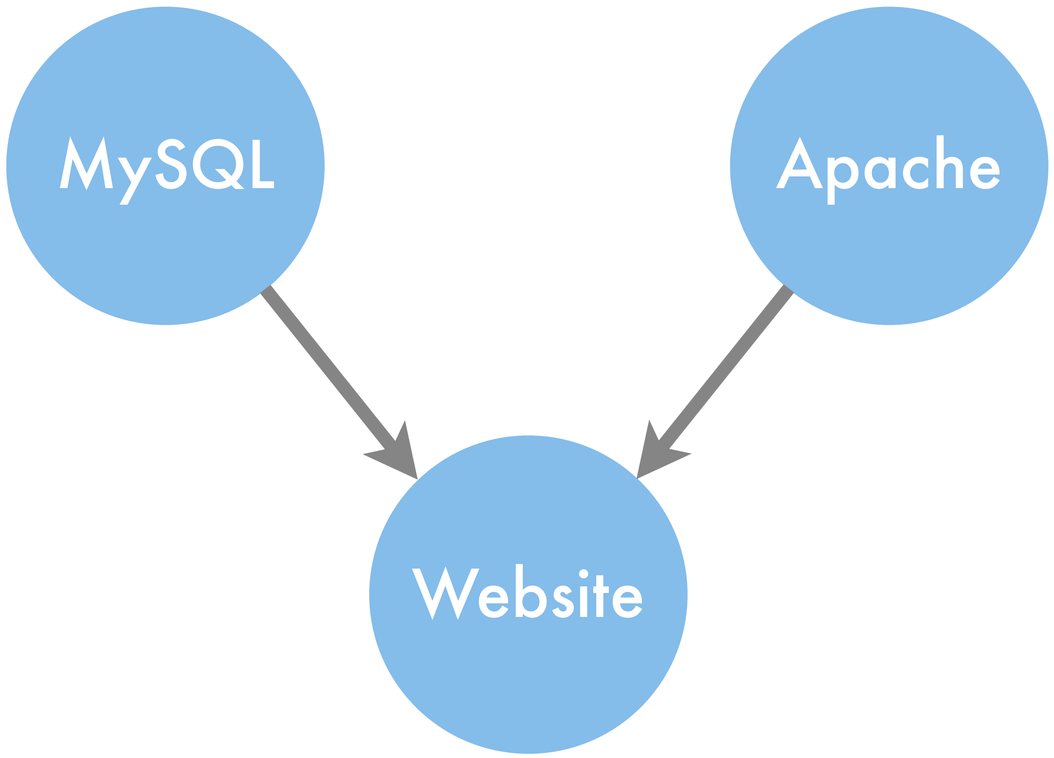
Directed Graphical Models



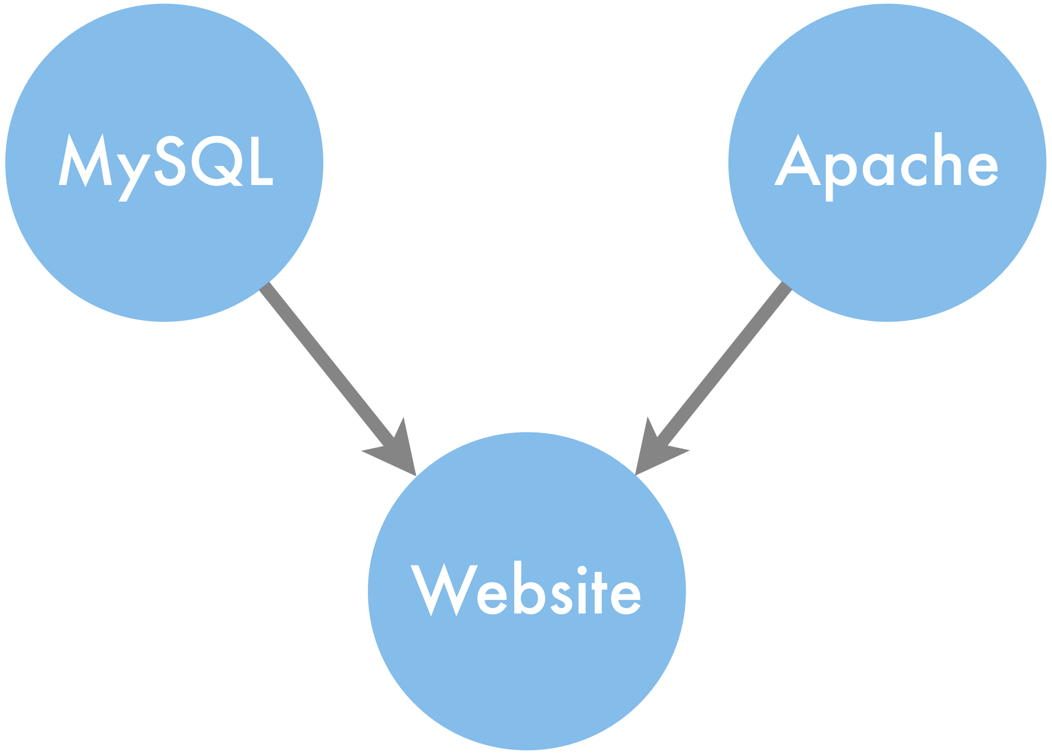
Basics


Microsoft
Development System
Build powerful applications fast!
Microsoft Visual Basic 6.0
Professional Edition

… some Web 2.0 service



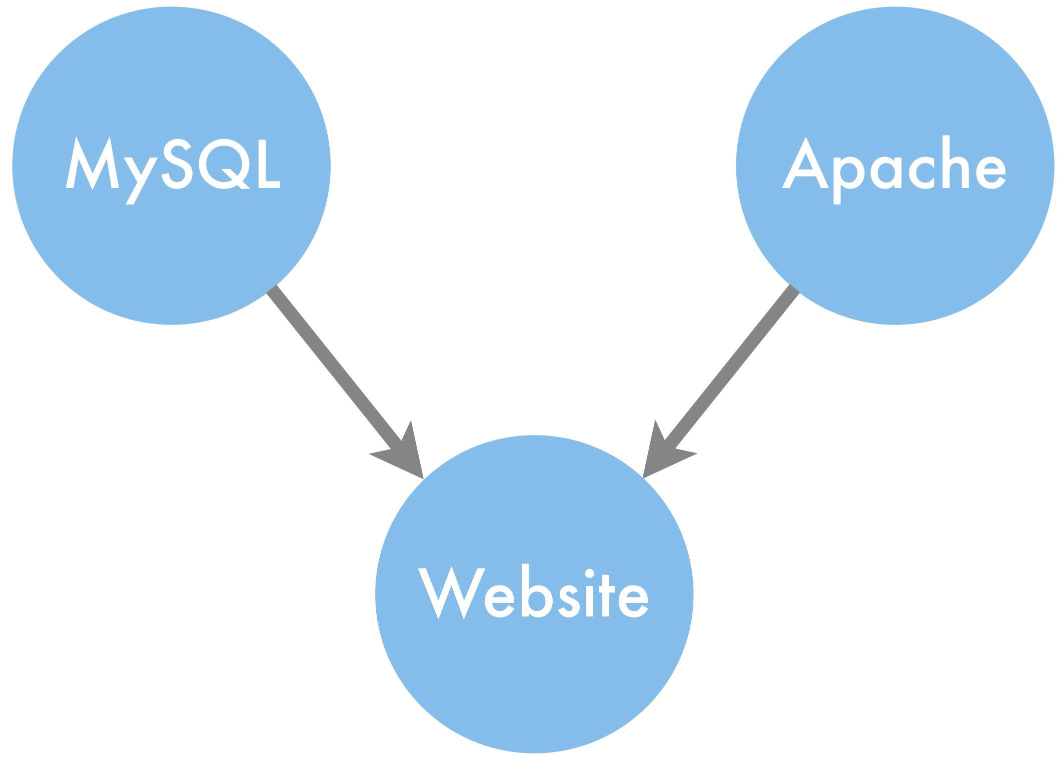
... some Web 2.0 service



- Joint distribution (assume a and m are independent)

$$p(m, a, w) = p(w|m, a)p(m)p(a)$$

... some Web 2.0 service



- Joint distribution (assume a and m are independent)

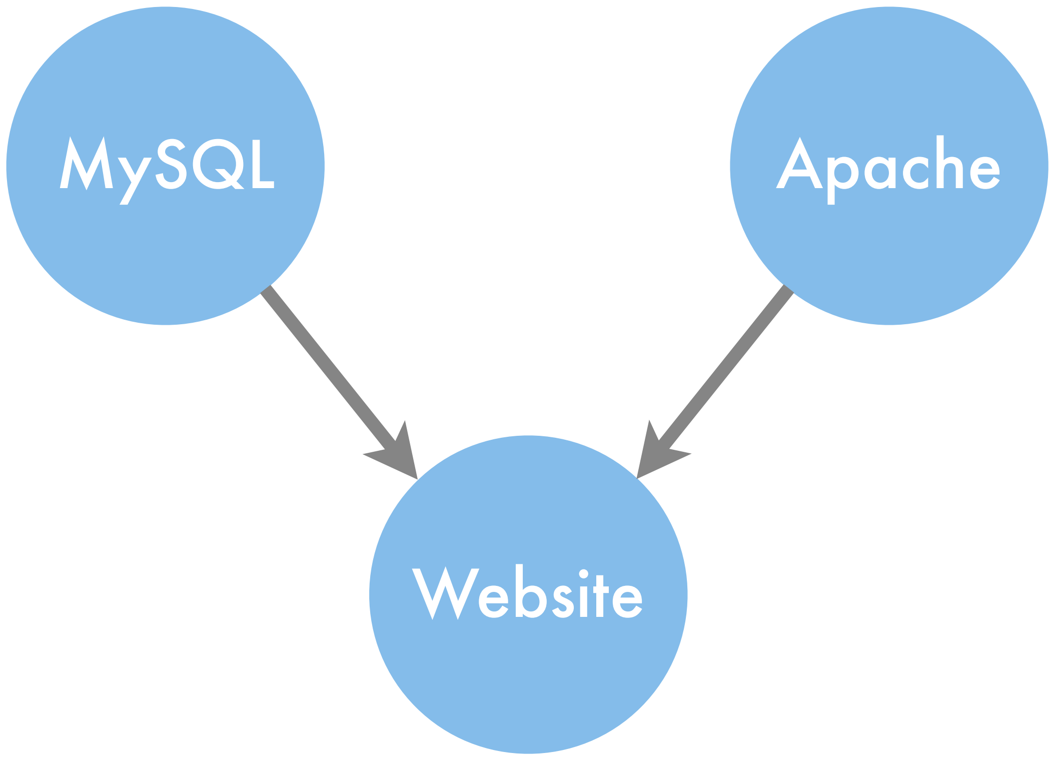
$$p(m, a, w) = p(w|m, a)p(m)p(a)$$

- Explaining away

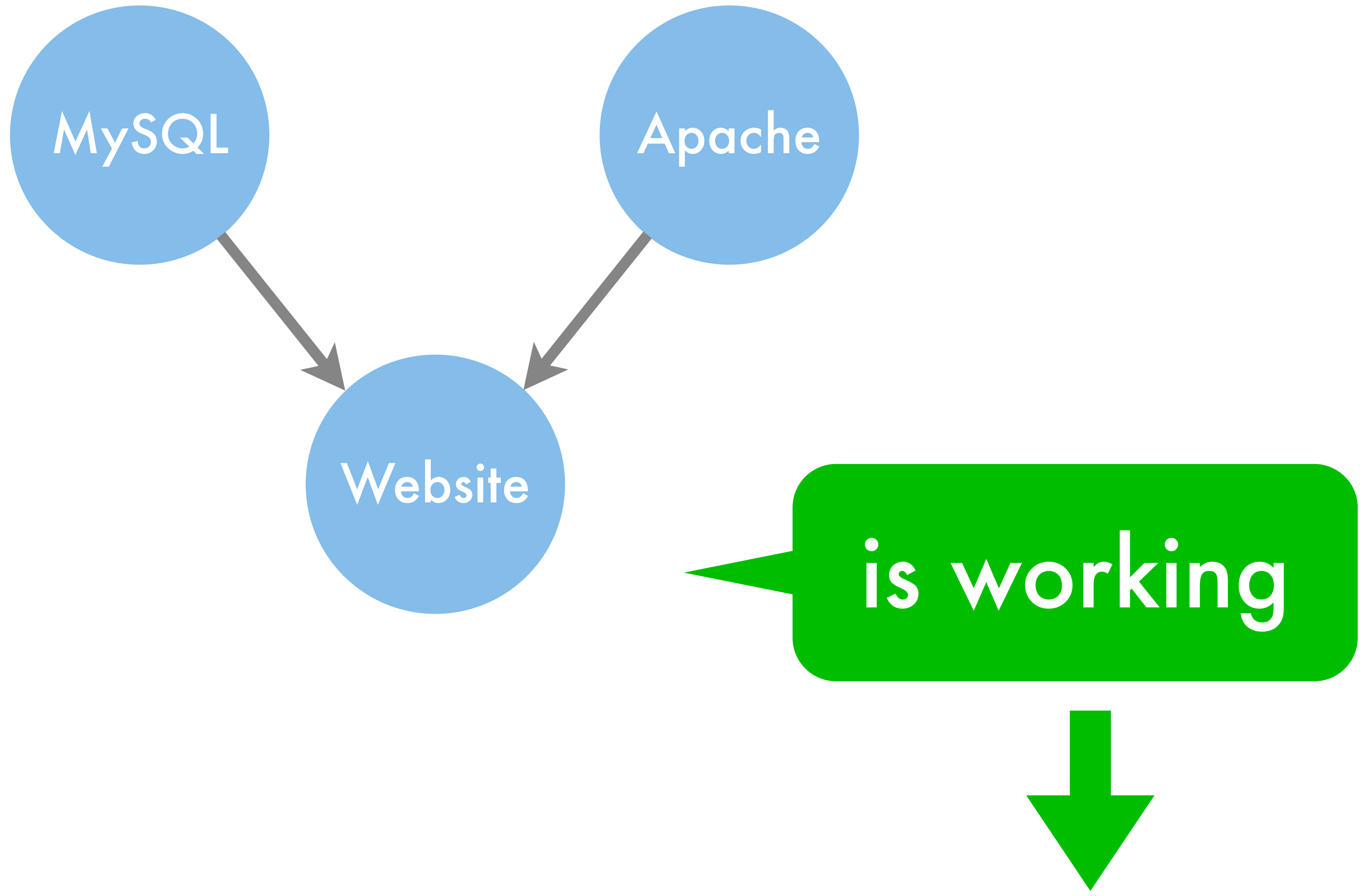
$$p(m, a|w) = \frac{p(w|m, a)p(m)p(a)}{\sum_{m', a'} p(w|m', a')p(m')p(a')}$$

a and m are dependent conditioned on w

… some Web 2.0 service

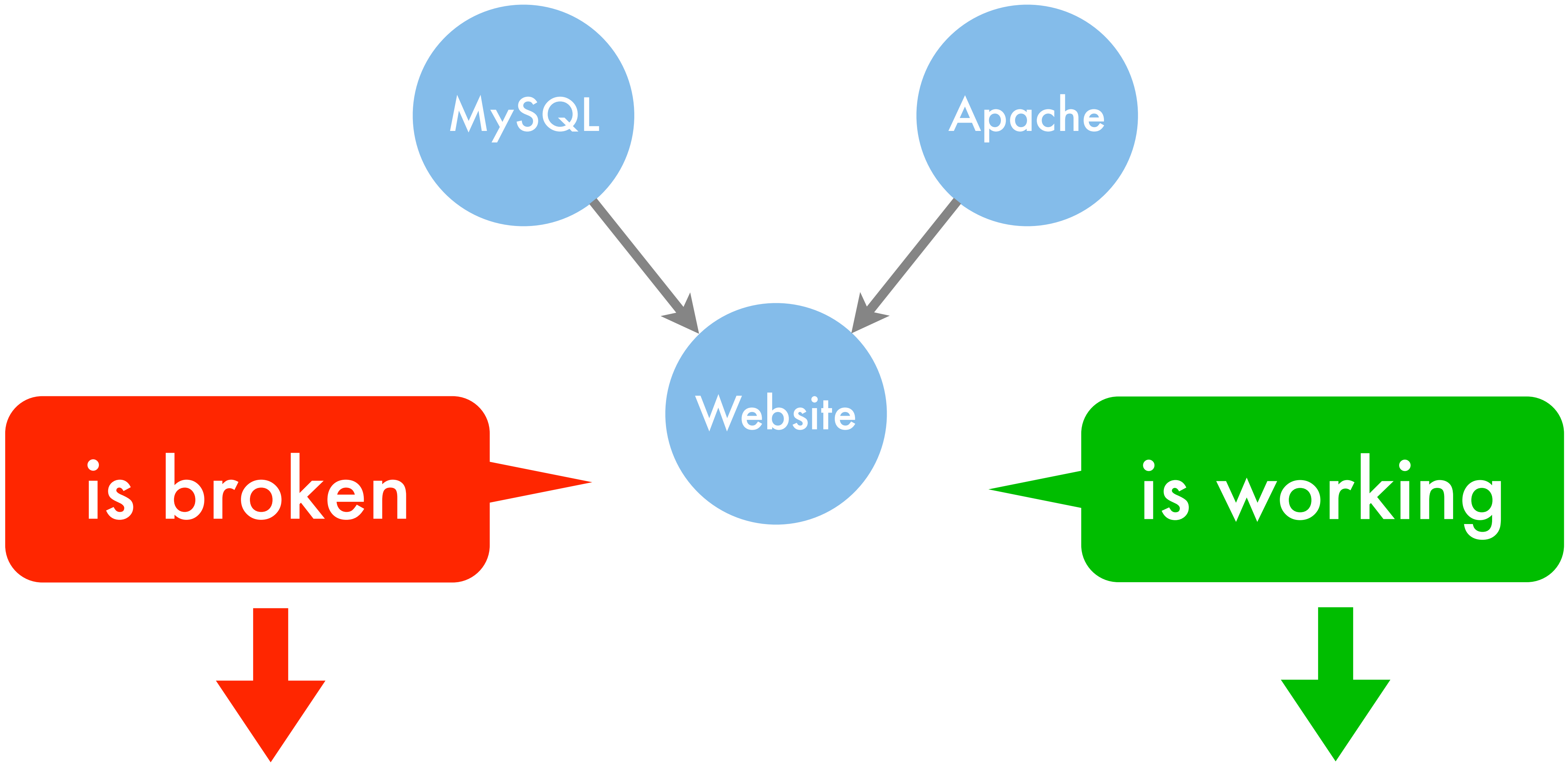


… some Web 2.0 service



MySQL is working
Apache is working

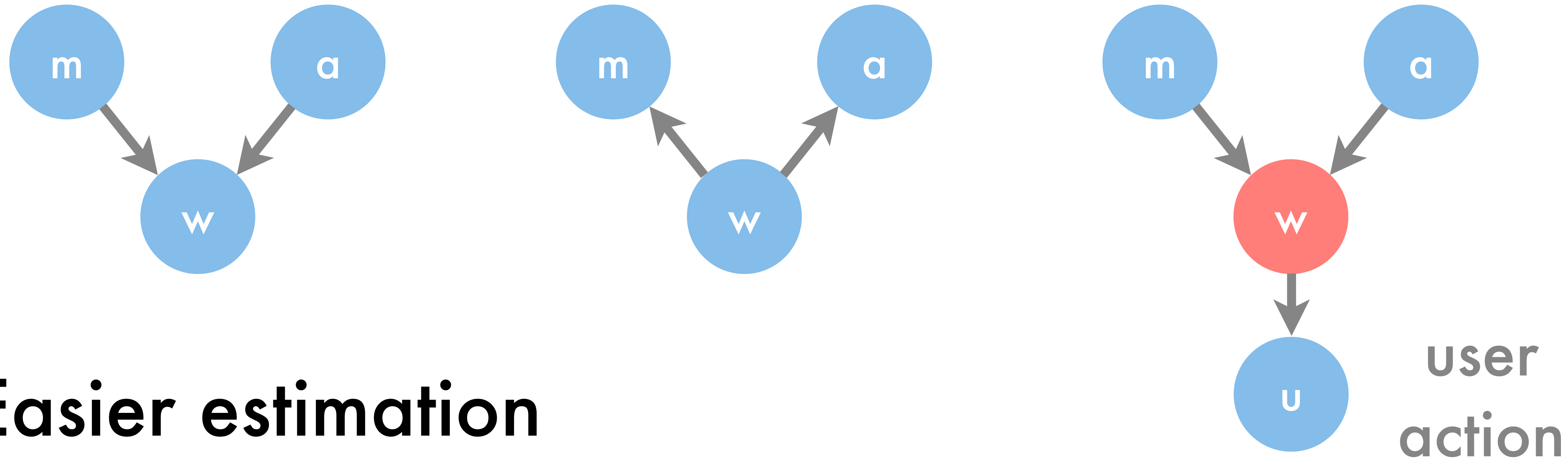
… some Web 2.0 service



At least one of the two services is broken (not independent)

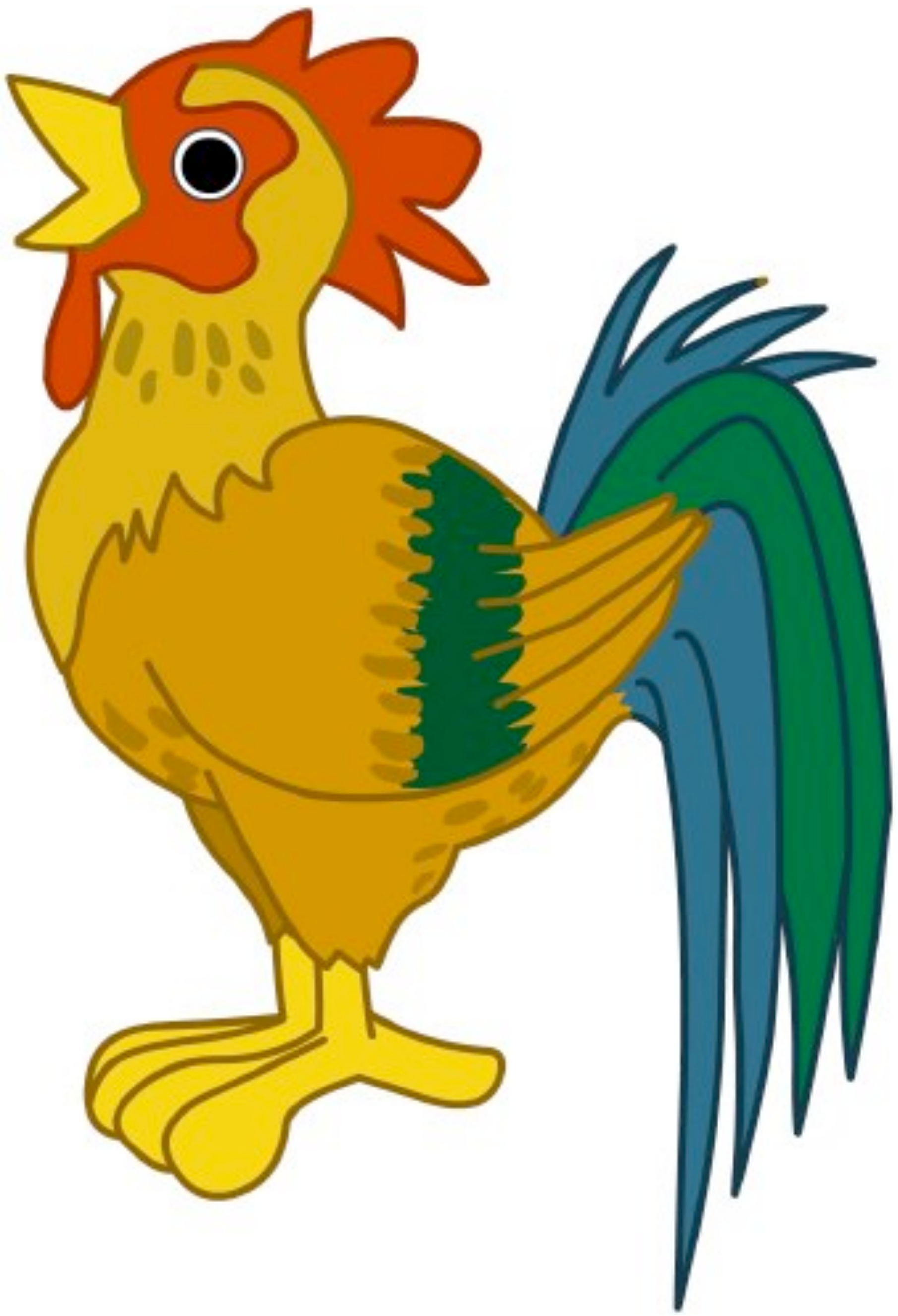
MySQL is working
Apache is working

Directed graphical model

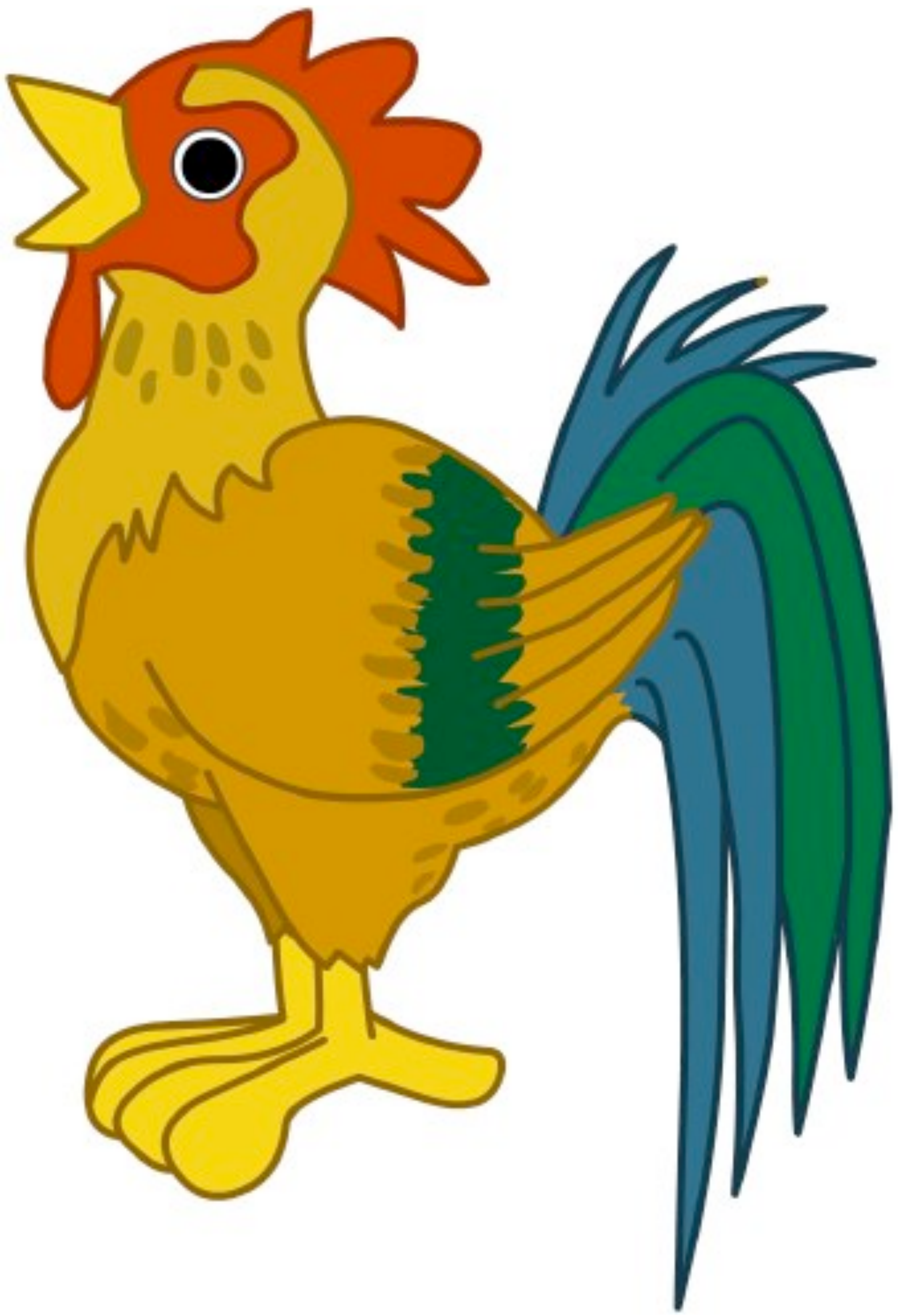


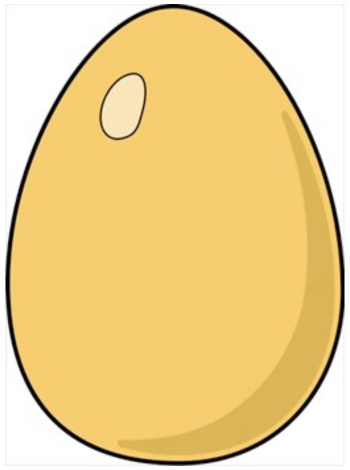
- Easier estimation
  - 15 parameters for full joint distribution
  - 1+1+4+1 for factorizing distribution
- Causal relations
- Inference for unobserved variables

No loops allowed

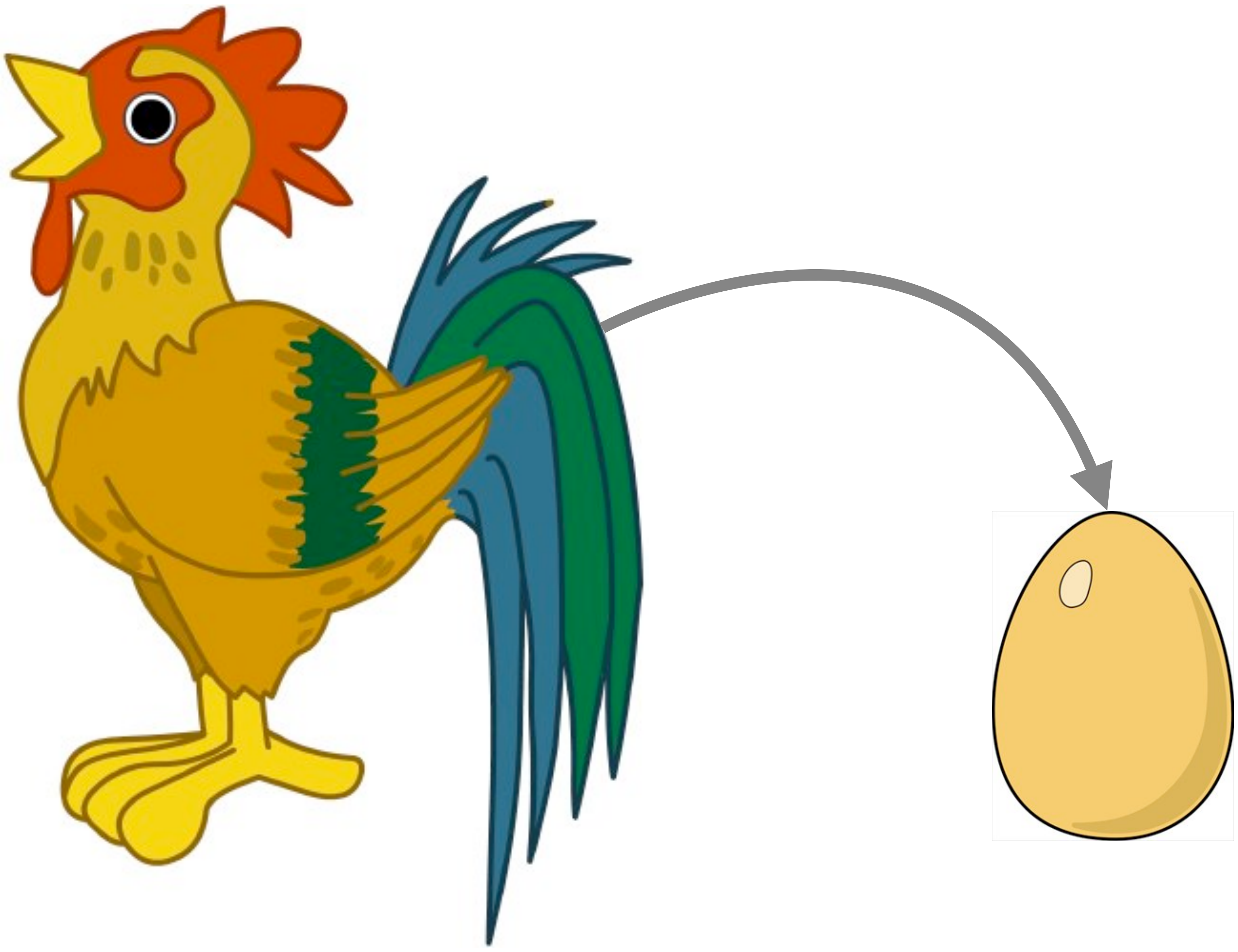


No loops allowed

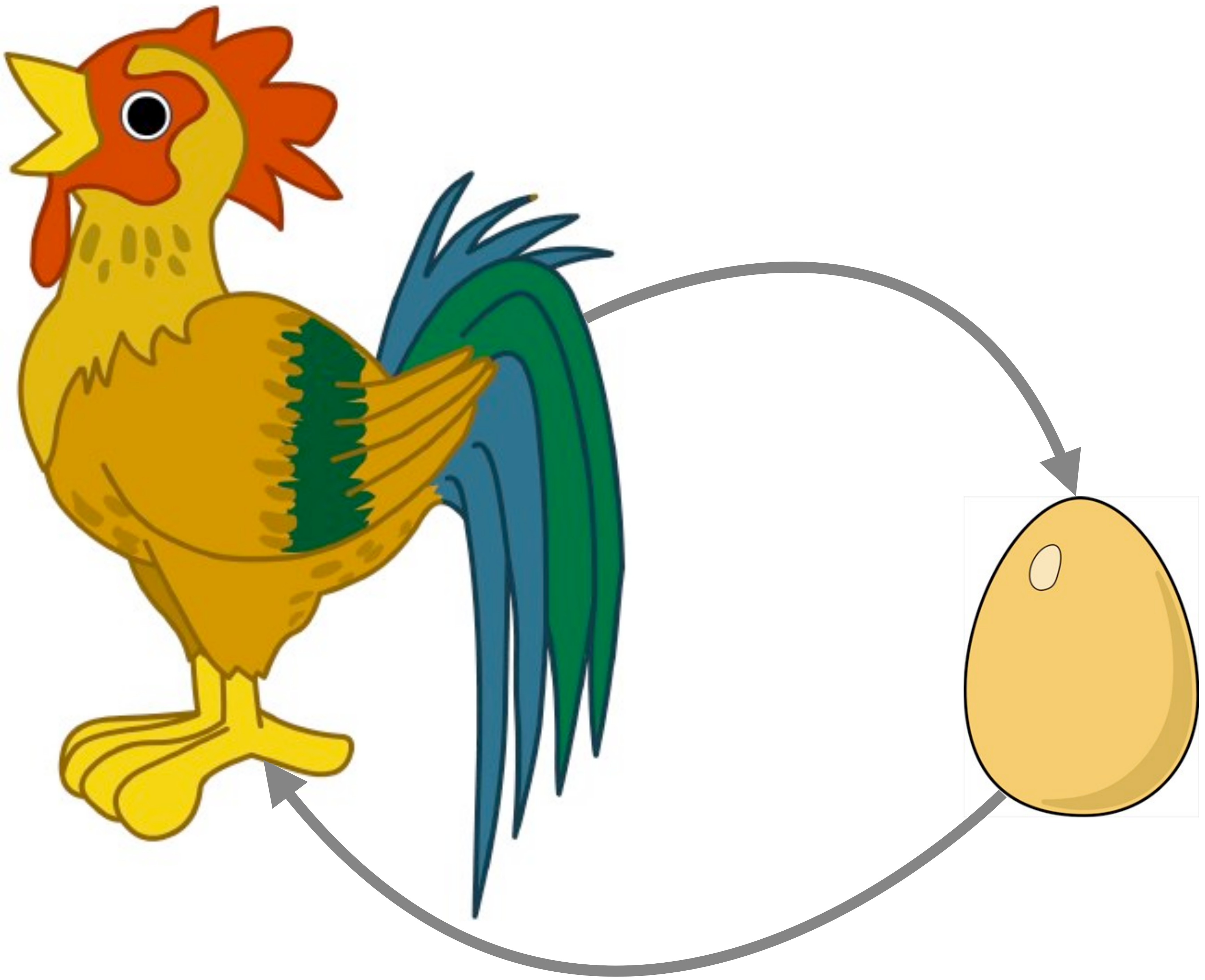




No loops allowed

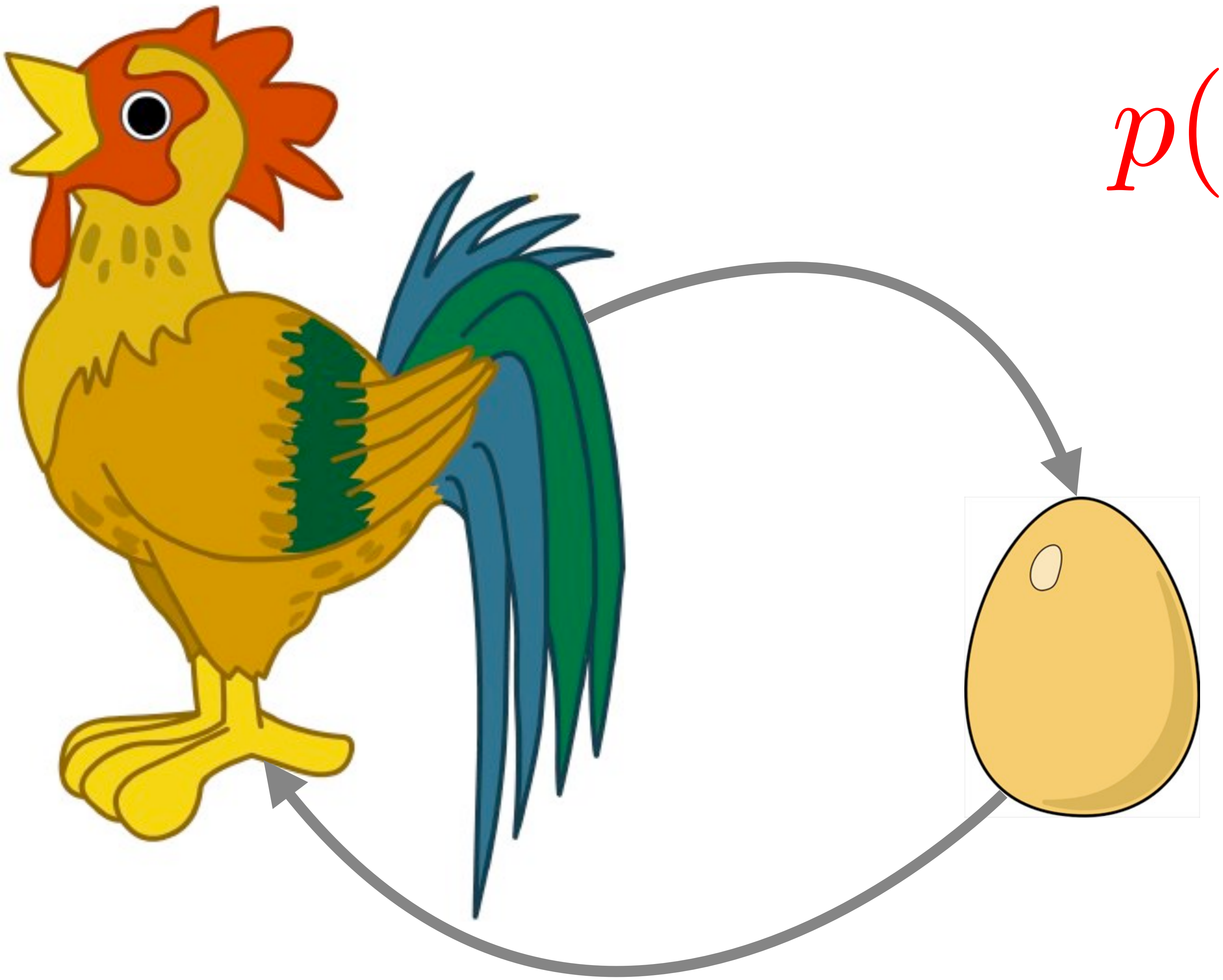


No loops allowed



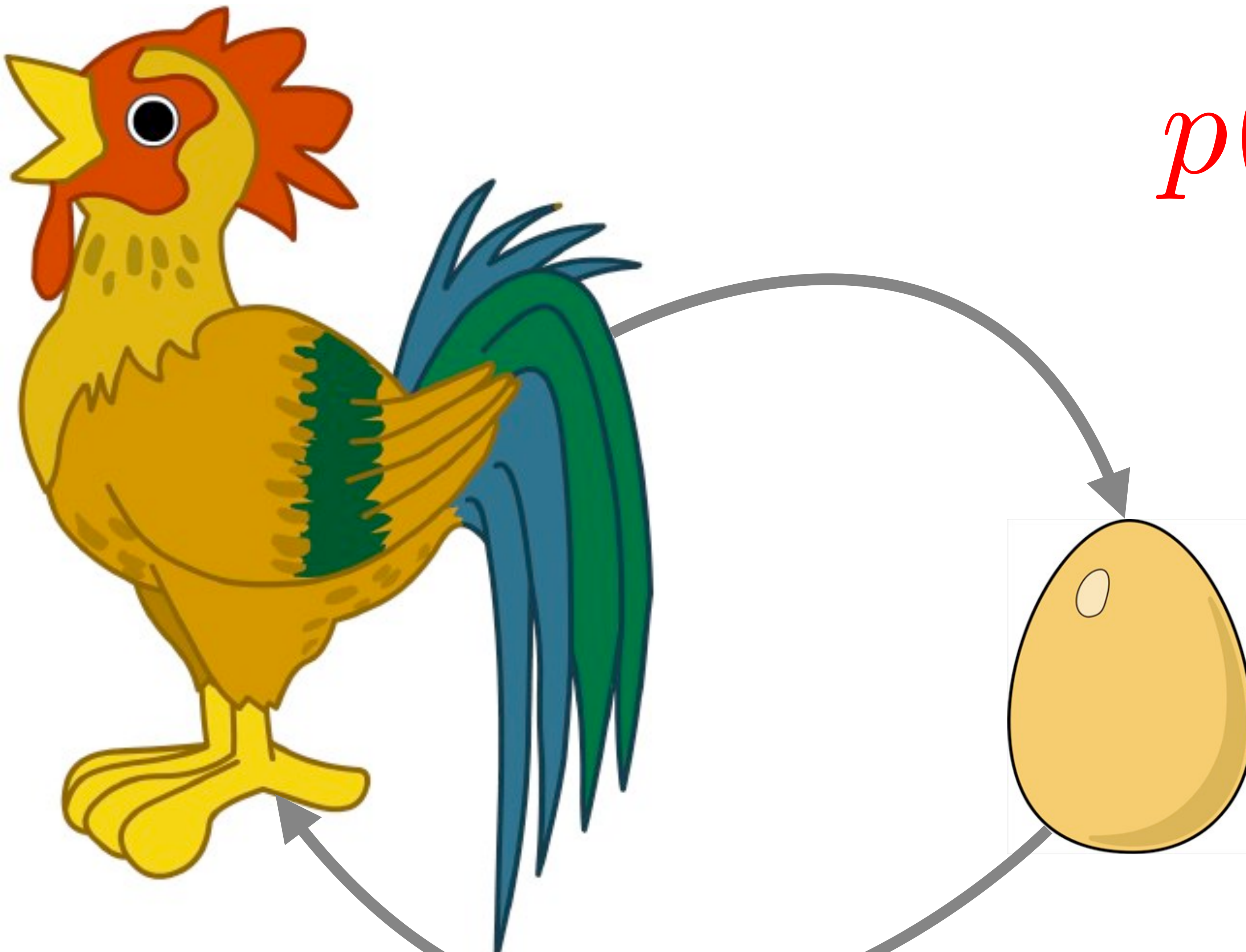
No loops allowed

$$p(c|e)p(e|c)$$



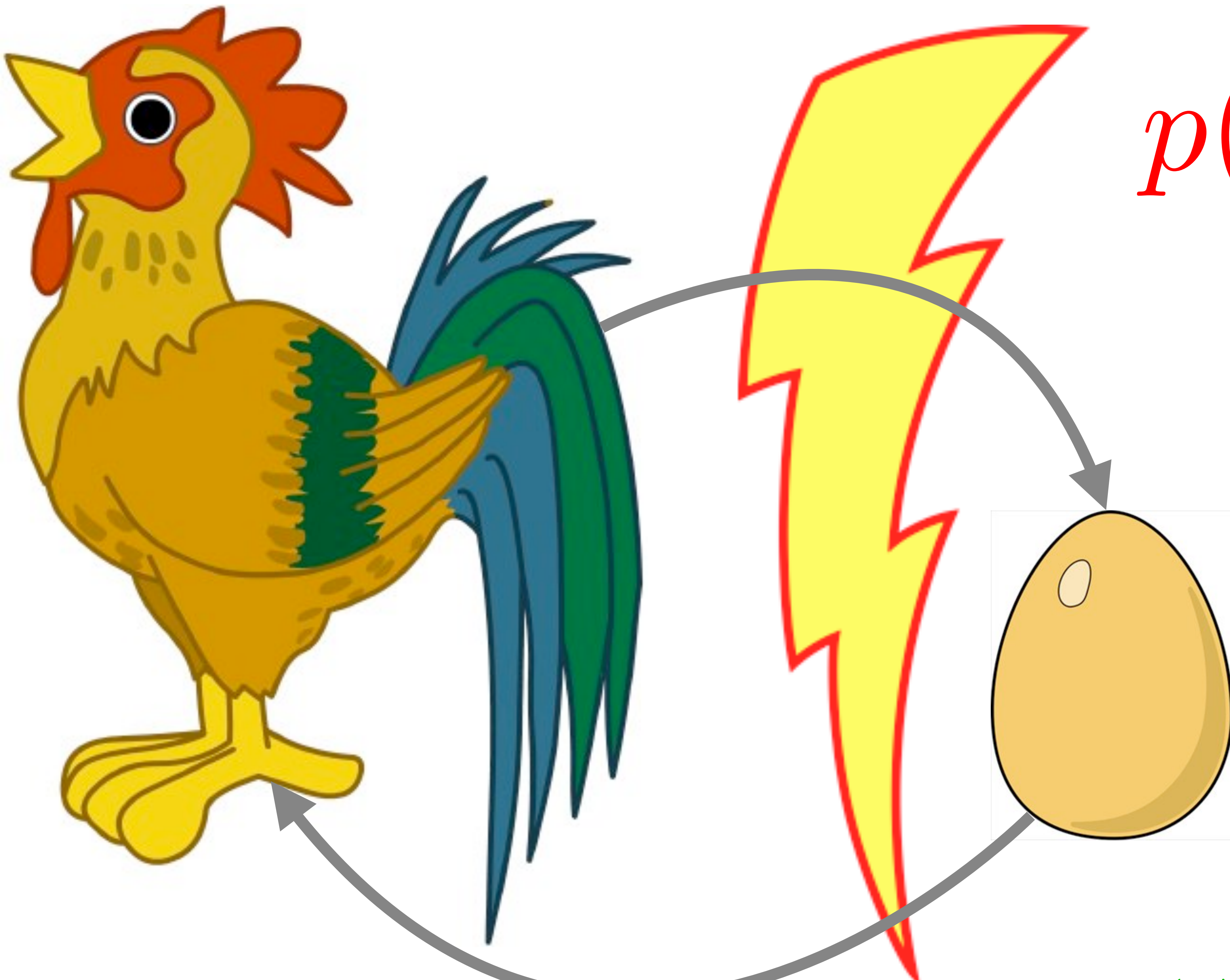
No loops allowed

$$p(c|e)p(e|c)$$



$$p(c|e)p(e) \text{ or } p(e|c)p(c)$$

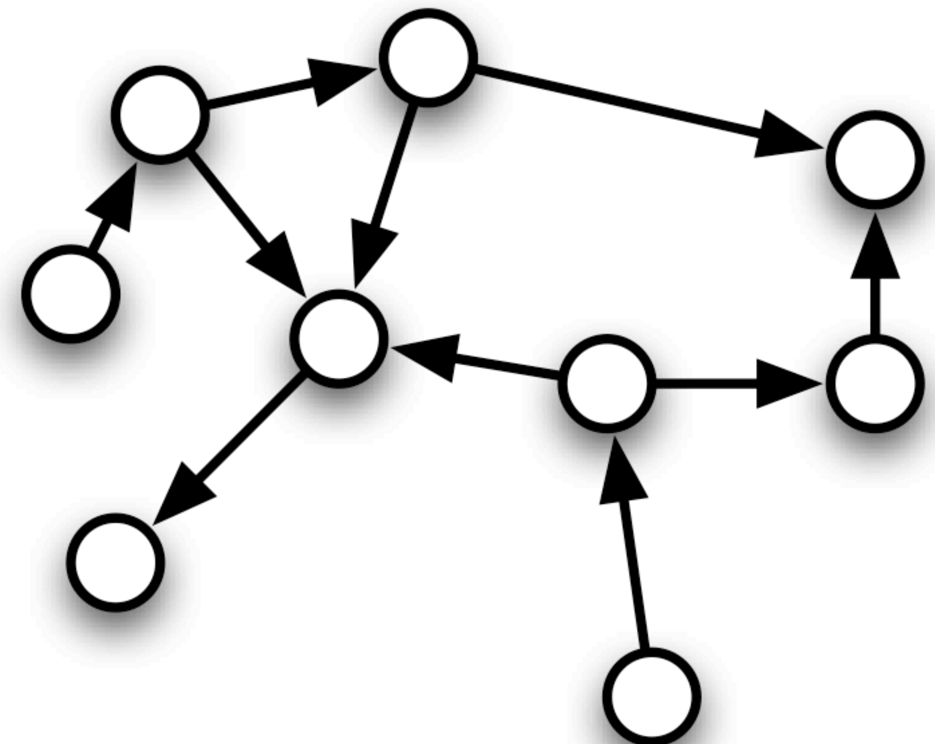
No loops allowed



$$p(c|e)p(e|c)$$

$$p(c|e)p(e) \text{ or } p(e|c)p(c)$$

Directed Graphical Model



- Joint probability distribution

$$p(x) = \prod_i p(x_i | x_{\text{parents(i)}})$$

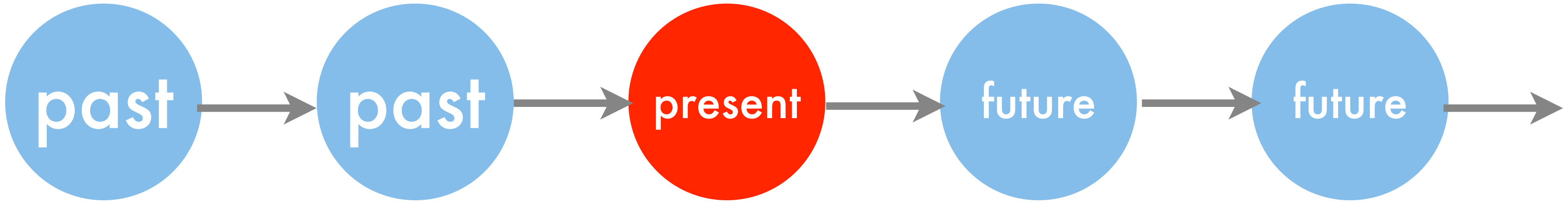
- Parameter estimation
  - If x is fully observed the likelihood breaks up

$$\log p(x|\theta) = \sum_i \log p(x_i | x_{\text{parents(i)}}, \theta)$$

  - If x is partially observed things get interesting
    maximization, EM, variational, sampling ...

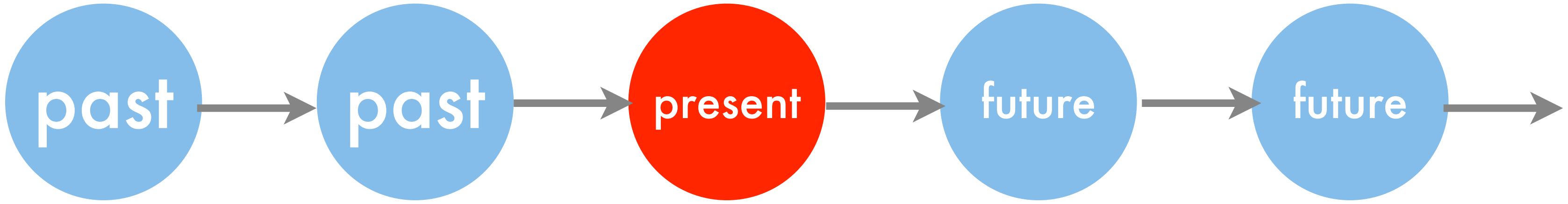
Chains

Markov Chain

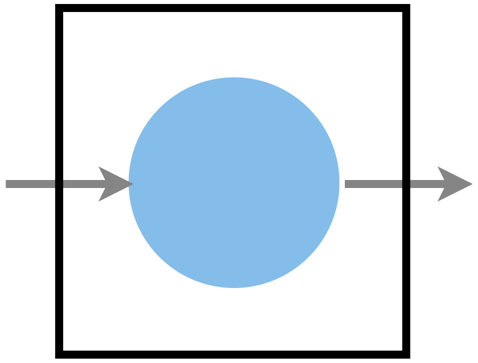


Chains

Markov Chain

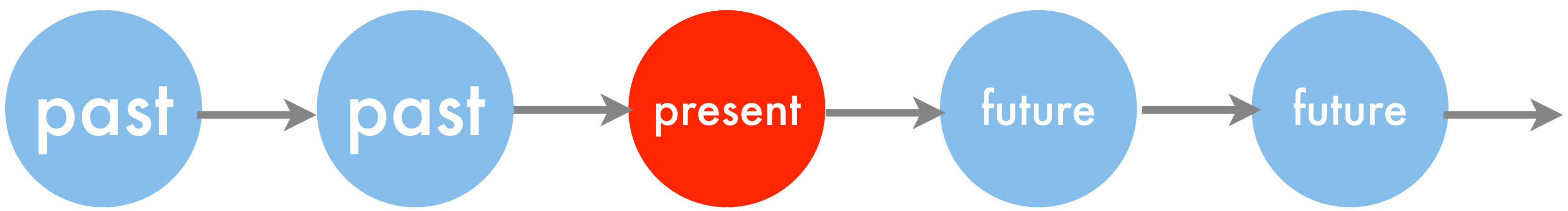


Plate

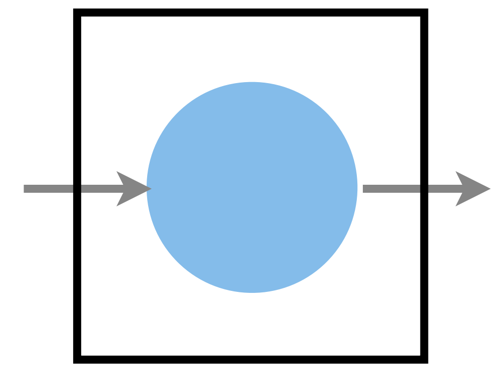


Chains

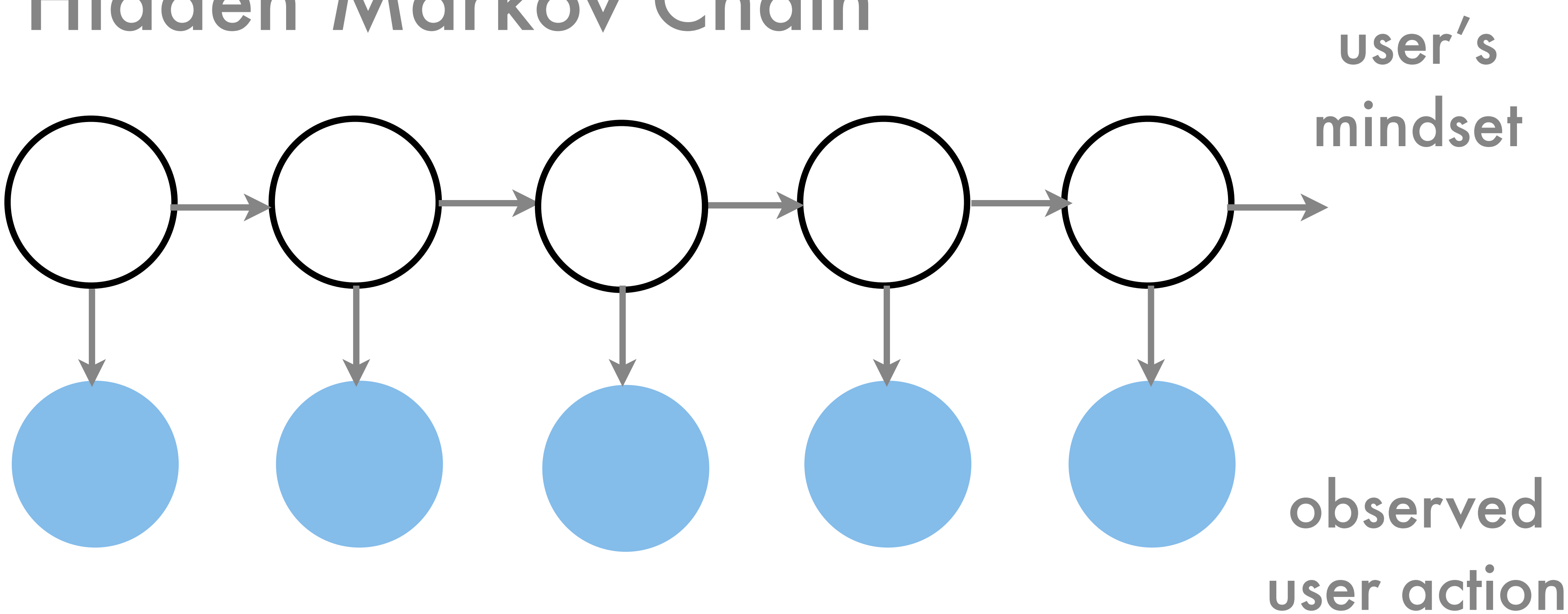
Markov Chain



Plate

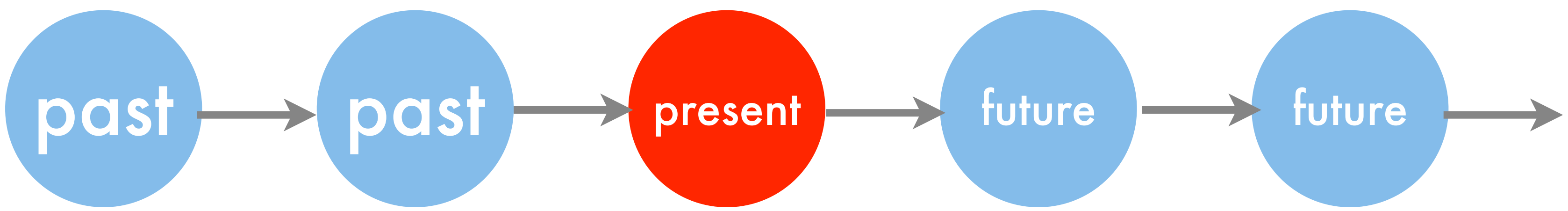


Hidden Markov Chain

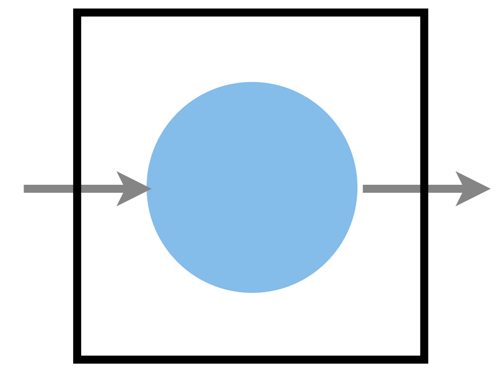


Chains

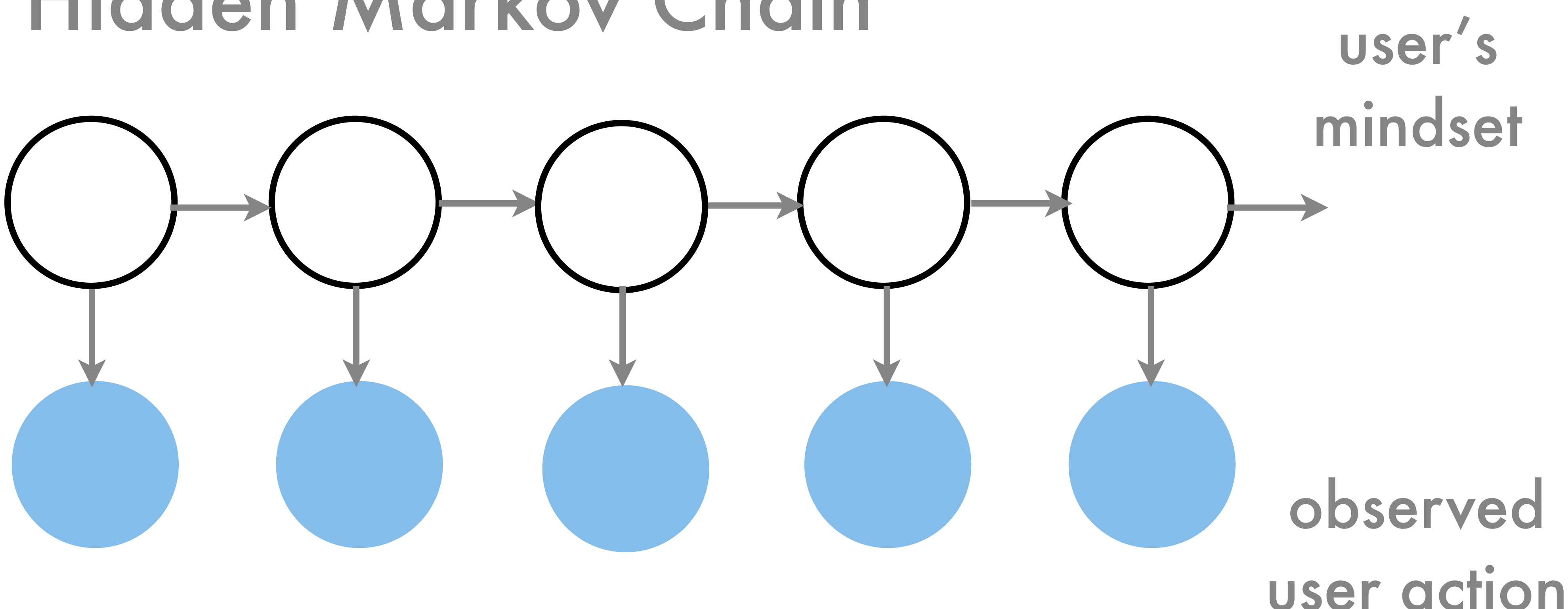
Markov Chain



Plate



Hidden Markov Chain

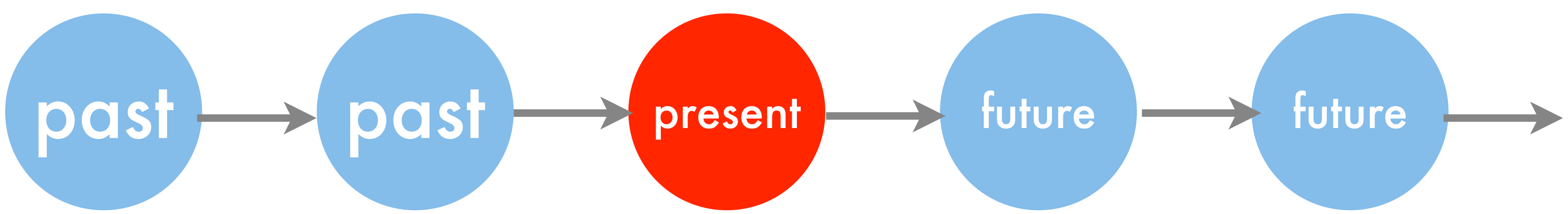


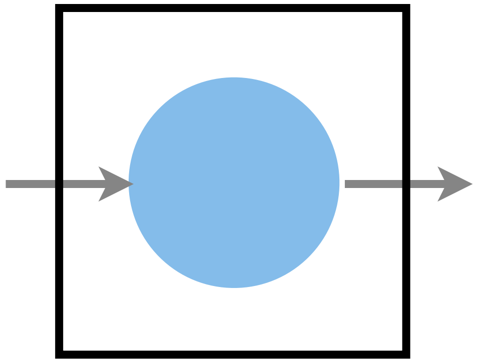
user model for traversal through search results

Chains

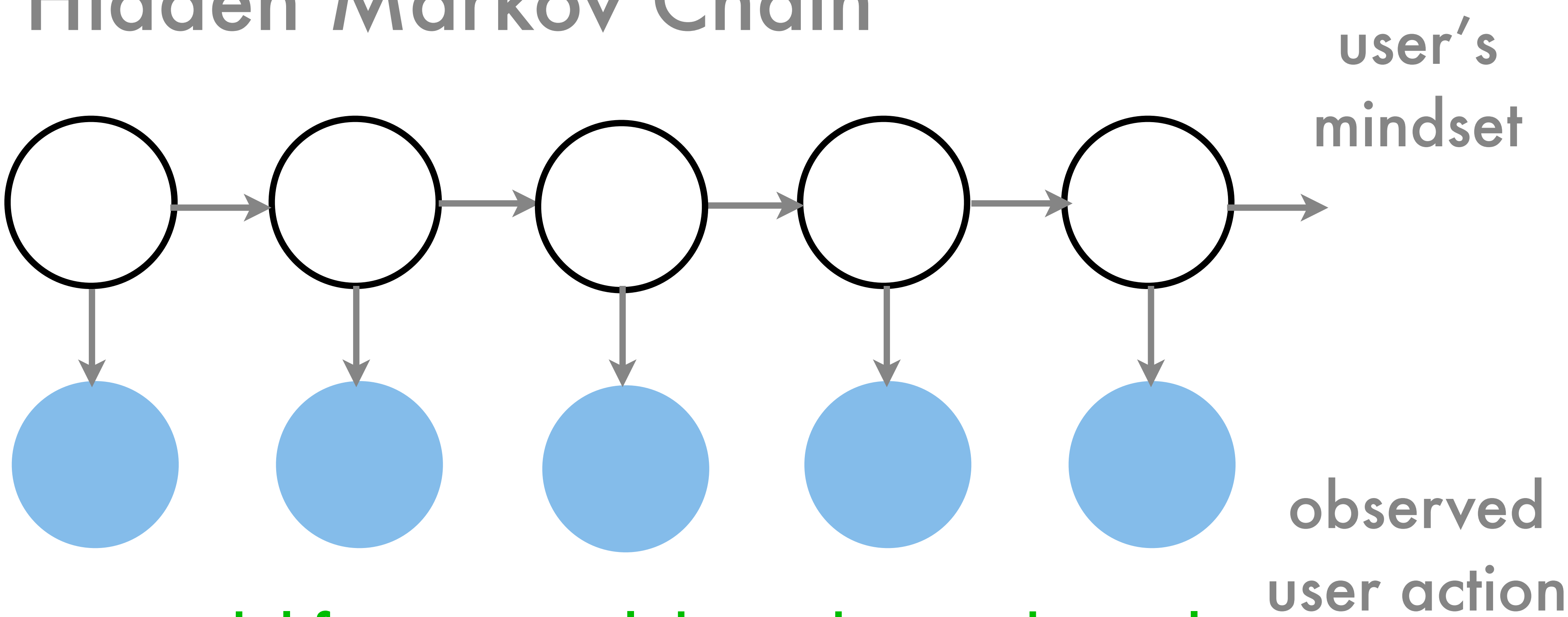
Markov Chain

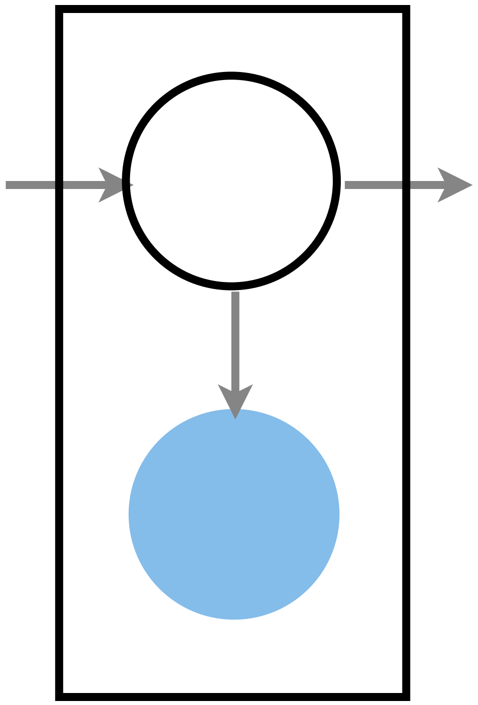
Plate





Hidden Markov Chain



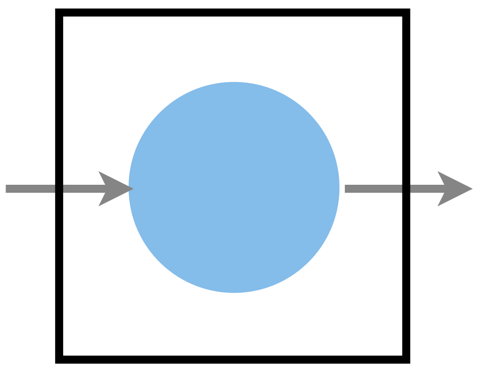


user model for traversal through search results

Chains

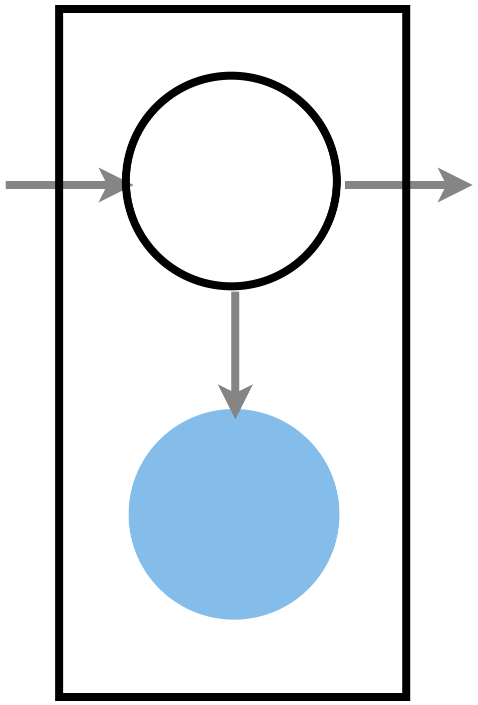
Markov Chain

Plate



$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$

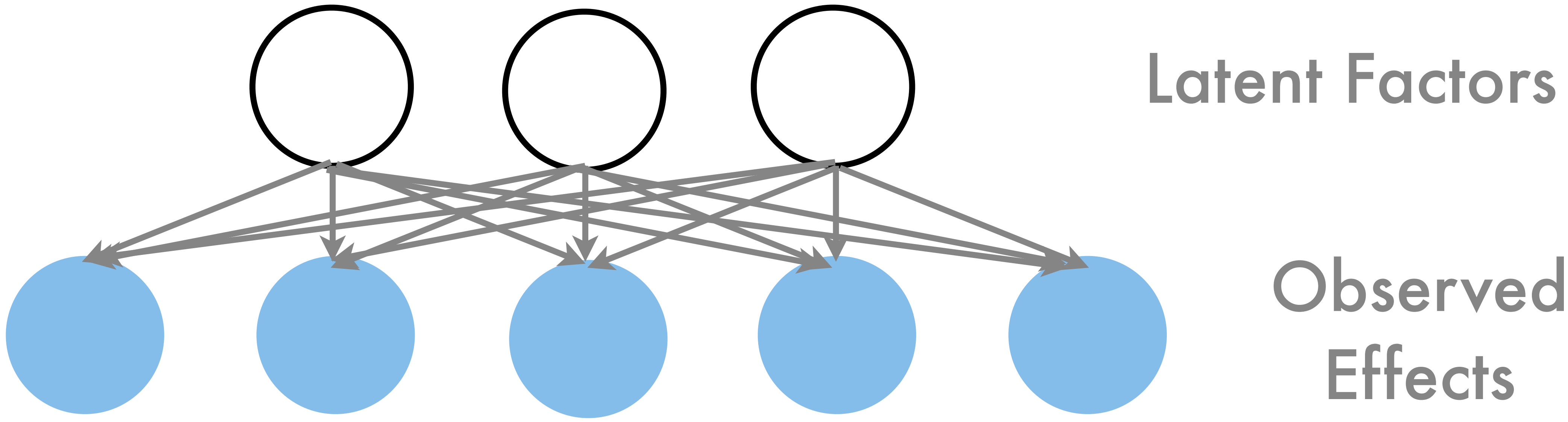
Hidden Markov Chain



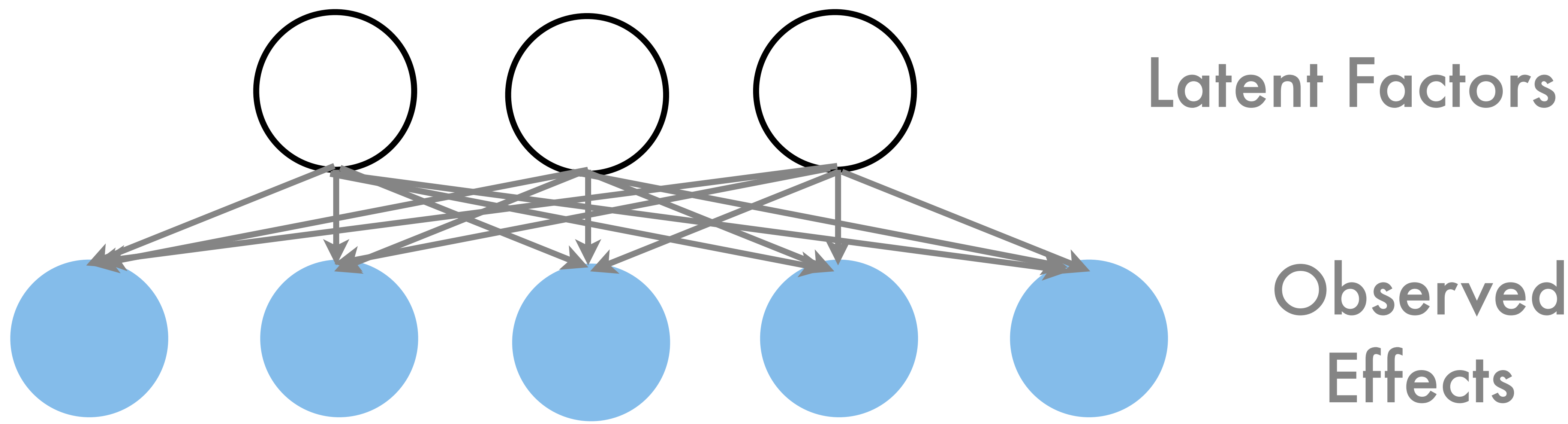
$$p(x,y;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)\prod_{i=1}^{n} p(y_i|x_i)$$

user model for traversal through search results

Factor Graphs

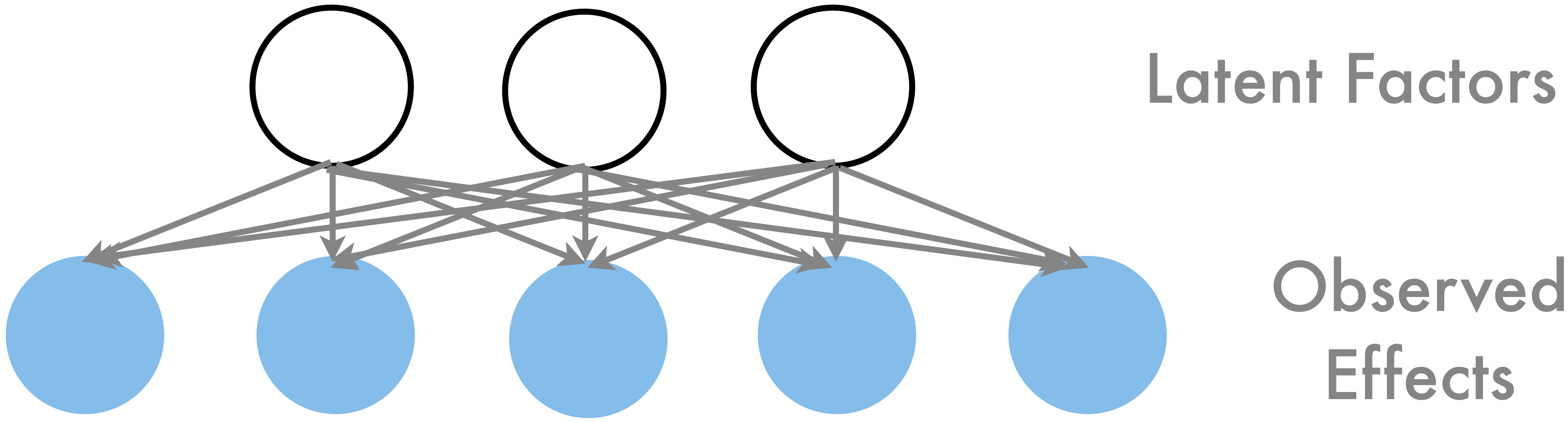


Factor Graphs



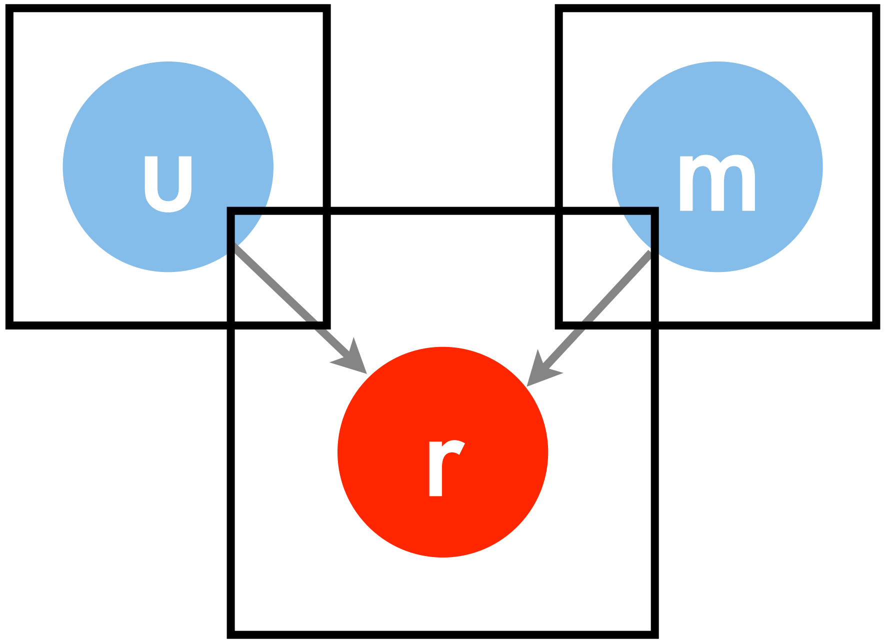
- Observed effects
  Click behavior, queries, watched news, emails

Factor Graphs

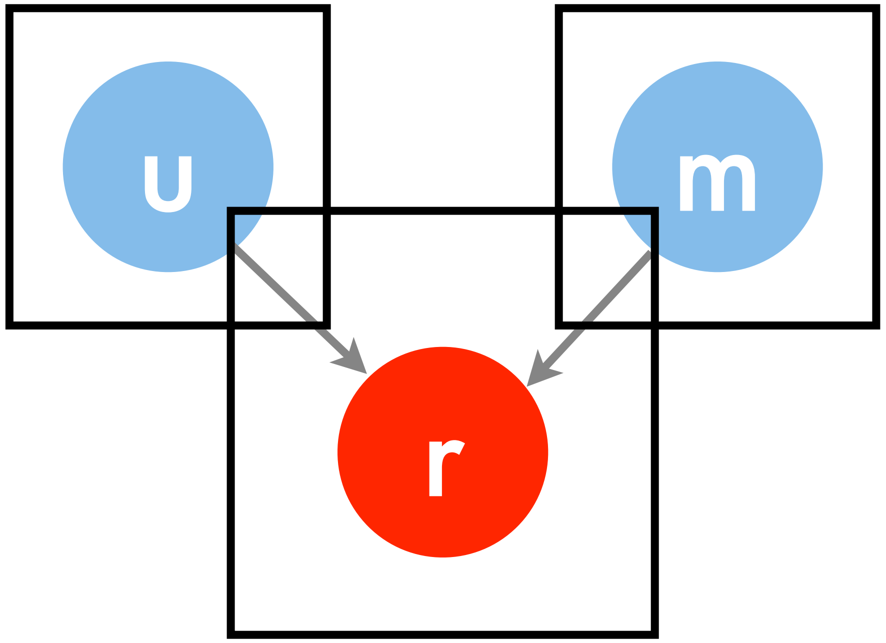


- Observed effects
  Click behavior, queries, watched news, emails
- Latent factors
  User profile, news content, hot keywords, social connectivity graph, events

Recommender Systems

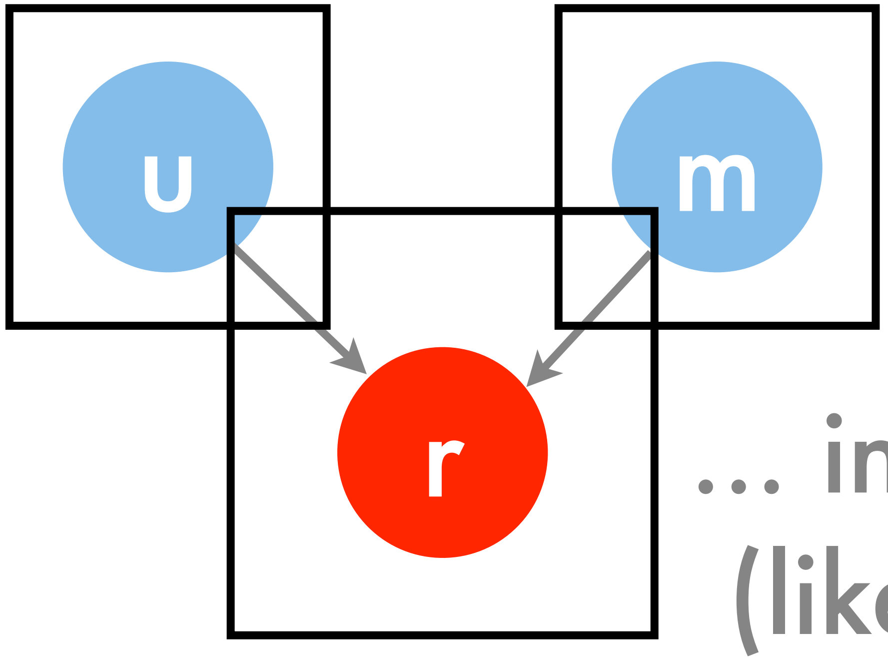


Recommender Systems



- Users u
- Movies m
- Ratings r (but only for a subset of users)

Recommender Systems

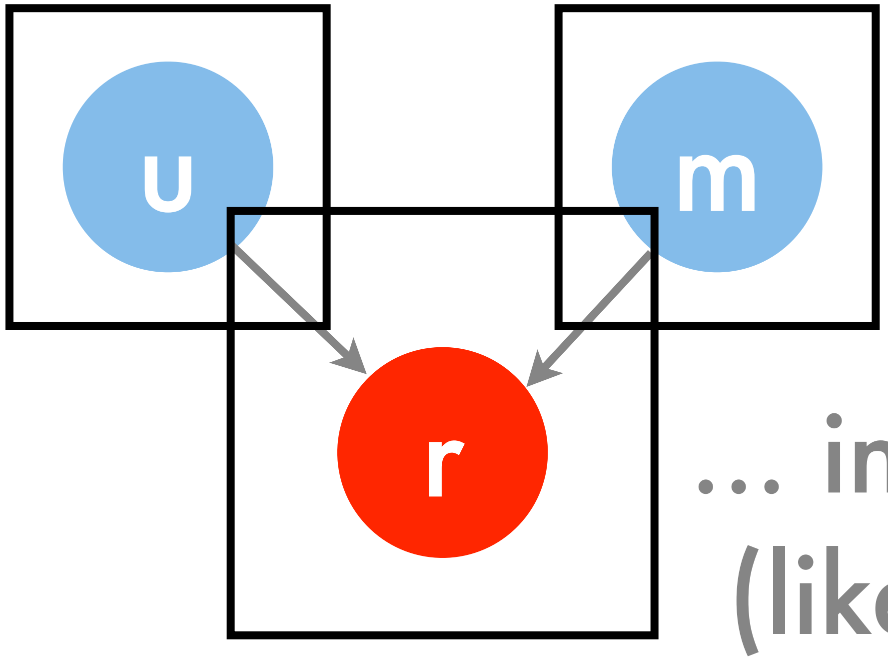


... intersecting plates ...
(like nested for loops)

- Users u
- Movies m
- Ratings r (but only for a subset of users)

Recommender Systems

news,
SearchMonkey
answers
social
ranking
OMG
personals



... intersecting plates ...
(like nested for loops)

- Users u
- Movies m
- Ratings r (but only for a subset of users)

Challenges

engineering

machine learning

Challenges

- How to design models
  - Common (engineering) sense
  - Computational tractability

engineering

machine learning

Challenges

- How to design models
  - Common (engineering) sense
  - Computational tractability
- Dependency analysis
  - Bayes ball (not in this lecture)

engineering

machine learning

Challenges

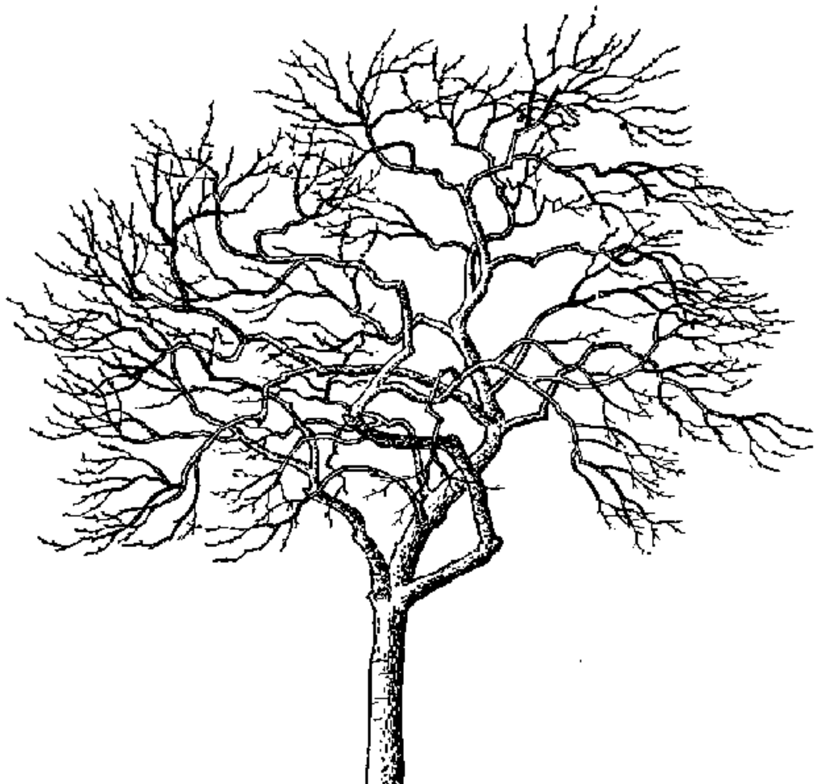
- How to design models
  - Common (engineering) sense
  - Computational tractability
- Dependency analysis
  - Bayes ball (not in this lecture)
- Inference
  - Easy for fully observed situations
  - Many algorithms if not fully observed
  - Dynamic programming / message passing

engineering

machine learning

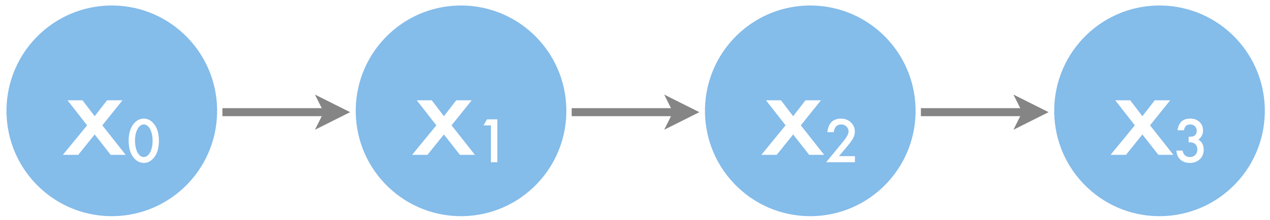
Dynamic Programming

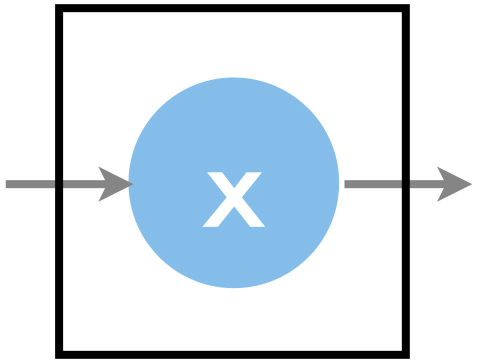
Chains and Trees



Chains

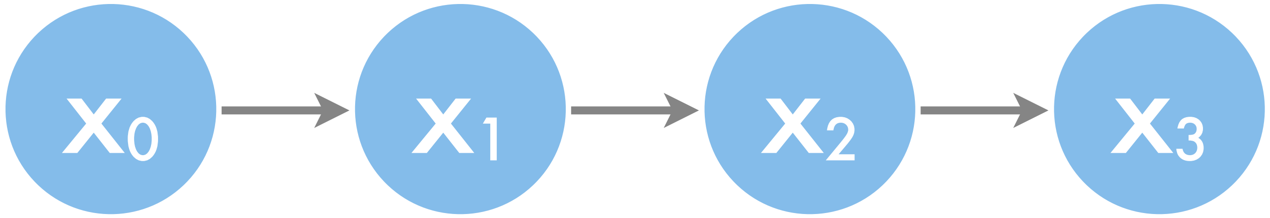
$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$



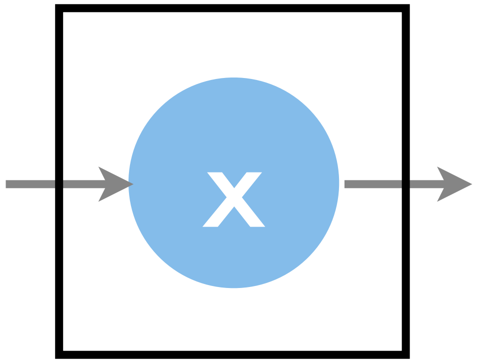


Chains

$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$

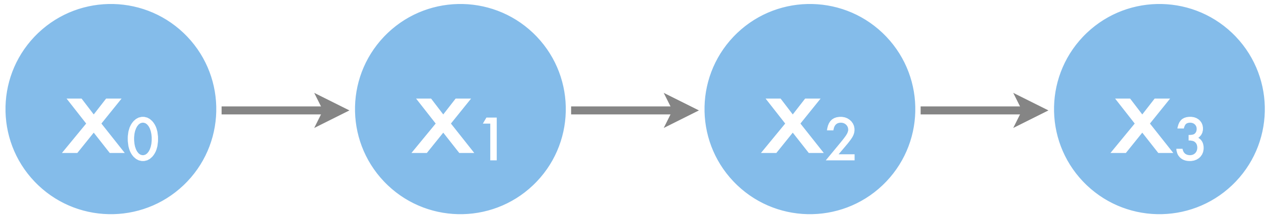


$$p(x_i) = \sum_{x_0,\dots x_{i-1},x_{i+1}\dots x_n} \underbrace{p(x_0)}_{:=l_0(x_0)} \prod_{j=1}^{n} p(x_j|x_{j-1})$$

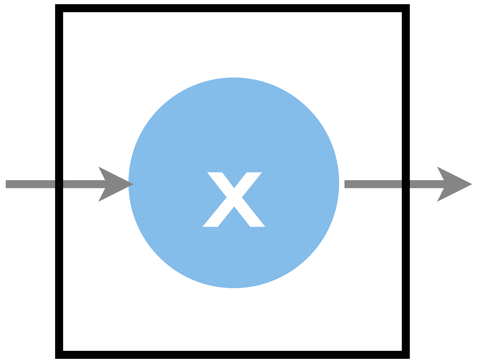


Chains

$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$

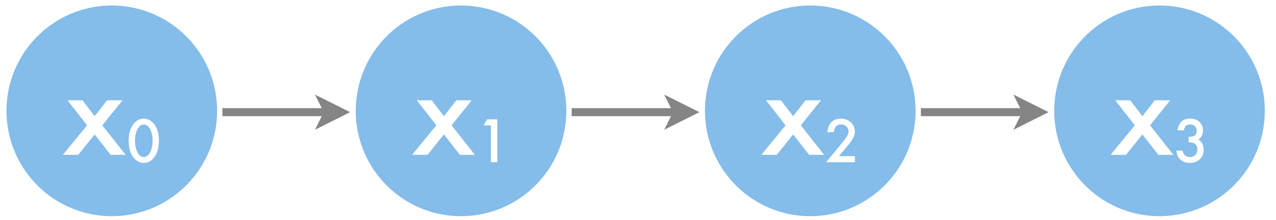


$$\begin{aligned} p(x_i) &= \sum_{x_0,\ldots x_{i-1},x_{i+1}\ldots x_n} \underbrace{p(x_0)}_{:=l_0(x_0)} \prod_{j=1}^{n} p(x_j|x_{j-1}) \\ &= \sum_{x_1,\ldots x_{i-1},x_{i+1}\ldots x_n} \underbrace{\sum_{x_0}[l_0(x_0)p(x_1|x_0)]}_{:=l_1(x_1)} \prod_{j=2}^{n} p(x_j|x_{j-1}) \end{aligned}$$

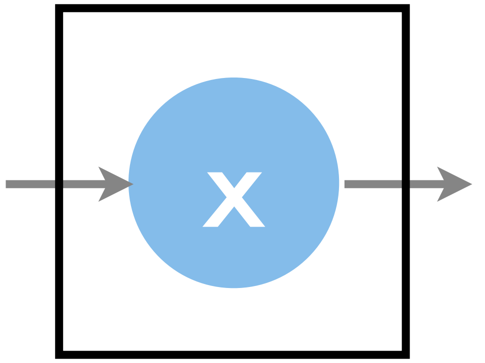


Chains

$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$

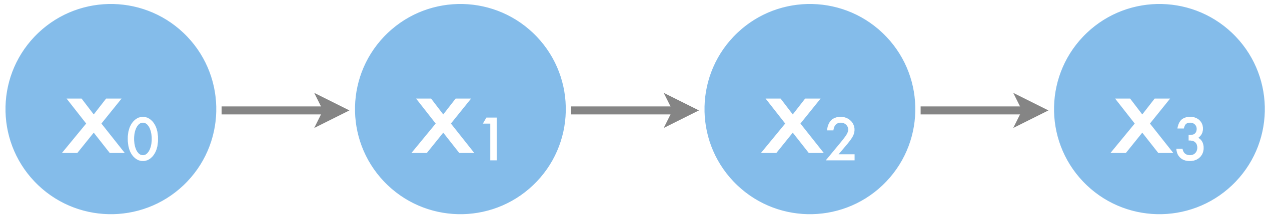


$$\begin{aligned} p(x_i) &= \sum_{x_0,\ldots x_{i-1},x_{i+1}\ldots x_n} \underbrace{p(x_0)}_{:=l_0(x_0)} \prod_{j=1}^{n} p(x_j|x_{j-1}) \\ &= \sum_{x_1,\ldots x_{i-1},x_{i+1}\ldots x_n} \underbrace{\sum_{x_0} [l_0(x_0)p(x_1|x_0)]}_{:=l_1(x_1)} \prod_{j=2}^{n} p(x_j|x_{j-1}) \\ &= \sum_{x_2,\ldots x_{i-1},x_{i+1}\ldots x_n} \underbrace{\sum_{x_1} [l_1(x_1)p(x_2|x_1)]}_{:=l_2(x_2)} \prod_{j=3}^{n} p(x_j|x_{j-1}) \end{aligned}$$



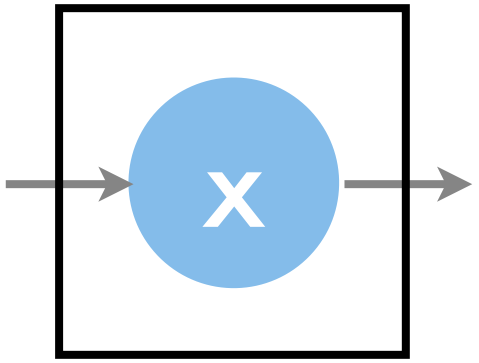
Chains

$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$



$$p(x_i) = l_i(x_i)\sum_{x_{i+1}\dots x_n}\prod_{j=i}^{n-1} p(x_{j+1}|x_j)$$

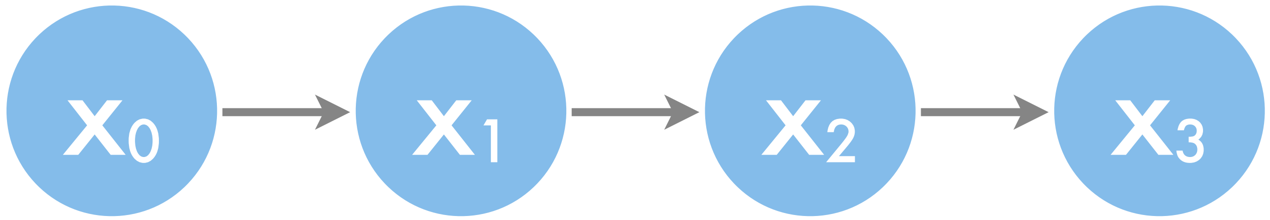
$$= l_i(x_i)\sum_{x_{i+1}\dots x_{n-1}}\prod_{j=i}^{n-2} p(x_{j+1}|x_j)\underbrace{\sum_{x_n} p(x_n|x_{n-1})}_{:=r_{n-1}(x_{n-1})}$$



$$= l_i(x_i)\sum_{x_{i+1}\dots x_{n-2}}\prod_{j=i}^{n-3} p(x_{j+1}|x_j)\underbrace{\sum_{x_{n-1}} p(x_{n-1}|x_{n-2})r_{n-1}(x_{n-1})}_{:=r_{n-2}(x_{n-2})}$$

Chains

$$p(x;\theta) = p(x_0;\theta)\prod_{i=1}^{n-1} p(x_{i+1}|x_i;\theta)$$



- **Forward recursion**

$$l_0(x_0) := p(x_0) \text{ and } l_i(x_i) := \sum_{x_{i-1}} l_{i-1}(x_{i-1})p(x_i|x_{i-1})$$

- **Backward recursion**

$$r_n(x_n) := 1 \text{ and } r_i(x_i) := \sum_{x_{i+1}} r_{i+1}(x_{i+1})p(x_{i+1}|x_i)$$

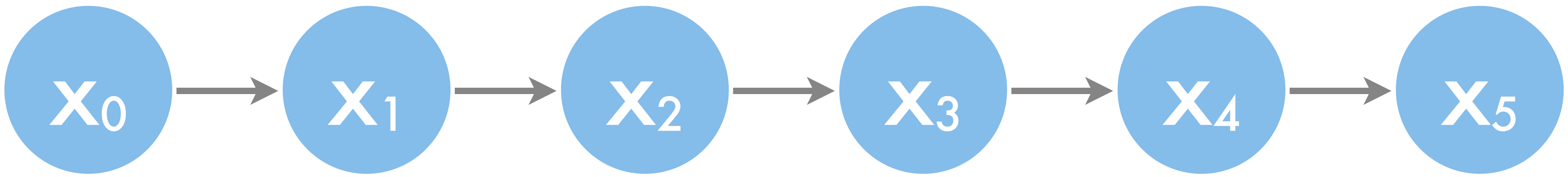
- **Marginalization & conditioning**

$$p(x_i) = l_i(x_i)r_i(x_i)$$

$$p(x_{-i}|x_i) = \frac{p(x)}{p(x_i)}$$

$$p(x_i, x_{i+1}) = l_i(x_i)p(x_{i+1}|x_i)r_i(x_{i+1})$$

Chains



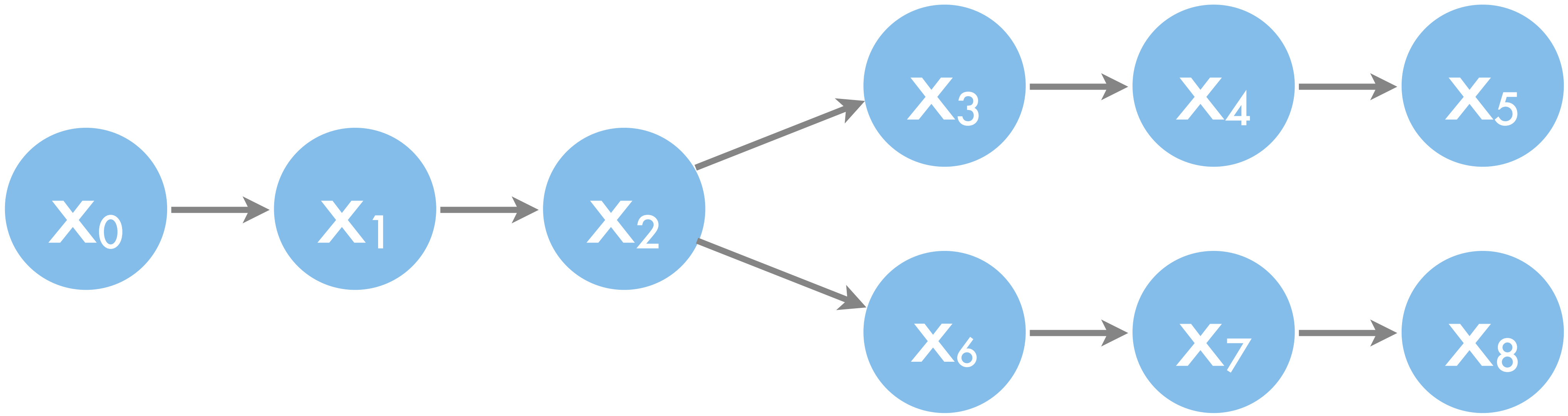
- Send forward messages starting from left node

$$m_{i-1\to i}(x_i) = \sum_{x_{i-1}} m_{i-2\to i-1}(x_{i-1}) f(x_{i-1}, x_i)$$

- Send backward messages starting from right node

$$m_{i+1\to i}(x_i) = \sum_{x_{i+1}} m_{i+2\to i+1}(x_{i+1}) f(x_i, x_{i+1})$$

Trees



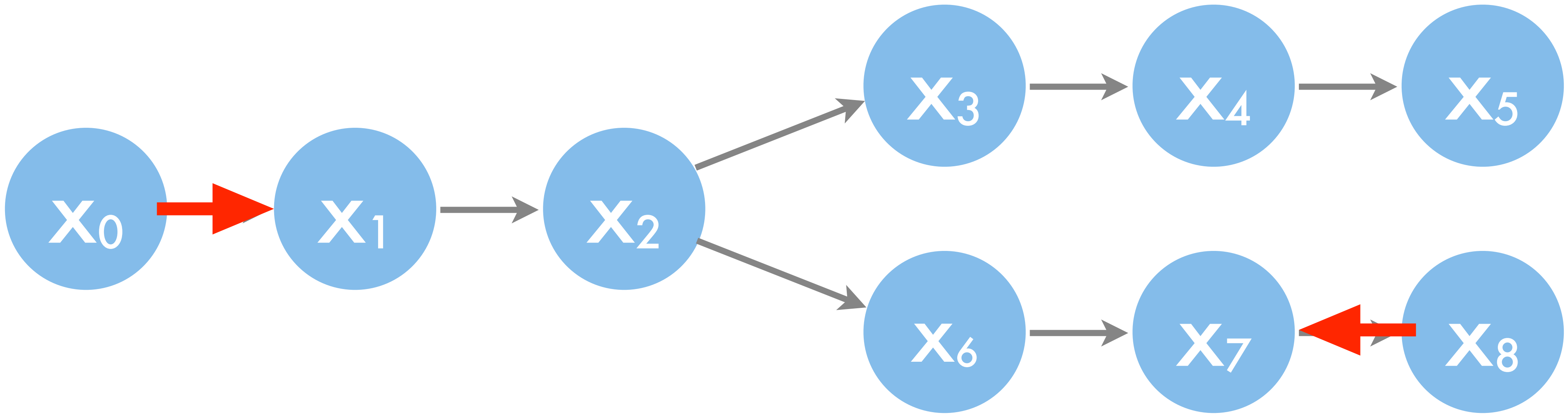
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to3}(x_3) = \sum_{x_2} m_{1\to2}(x_2) m_{6\to2}(x_2) f(x_2, x_3)$$

$$m_{2\to6}(x_6) = \sum_{x_2} m_{1\to2}(x_2) m_{3\to2}(x_2) f(x_2, x_6)$$

$$m_{2\to1}(x_1) = \sum_{x_2} m_{3\to2}(x_2) m_{6\to2}(x_2) f(x_1, x_2)$$

Trees



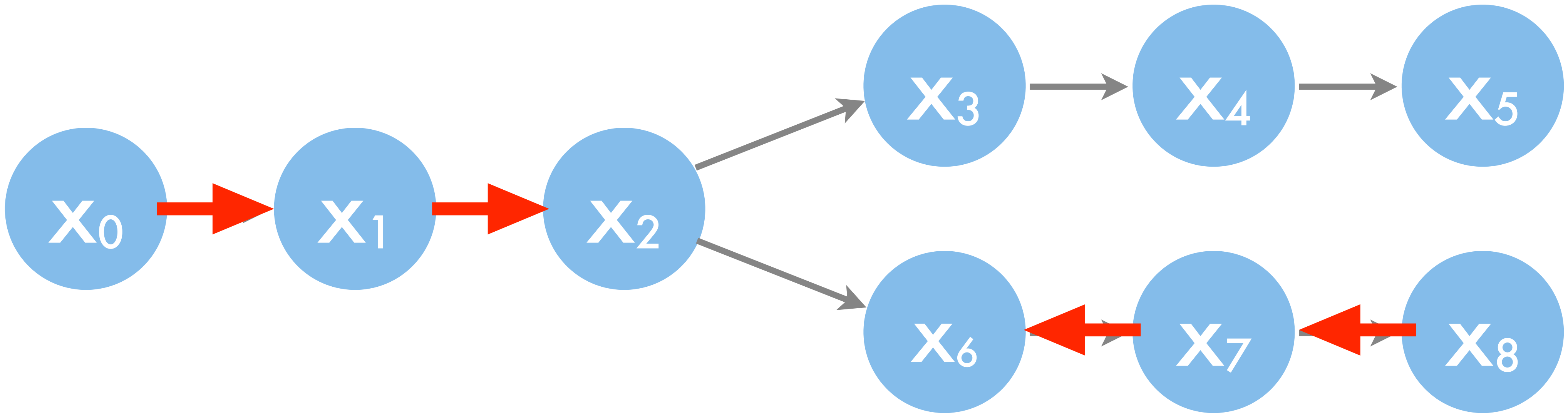
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to 3}(x_3) = \sum_{x_2} m_{1\to 2}(x_2) m_{6\to 2}(x_2) f(x_2, x_3)$$

$$m_{2\to 6}(x_6) = \sum_{x_2} m_{1\to 2}(x_2) m_{3\to 2}(x_2) f(x_2, x_6)$$

$$m_{2\to 1}(x_1) = \sum_{x_2} m_{3\to 2}(x_2) m_{6\to 2}(x_2) f(x_1, x_2)$$

Trees



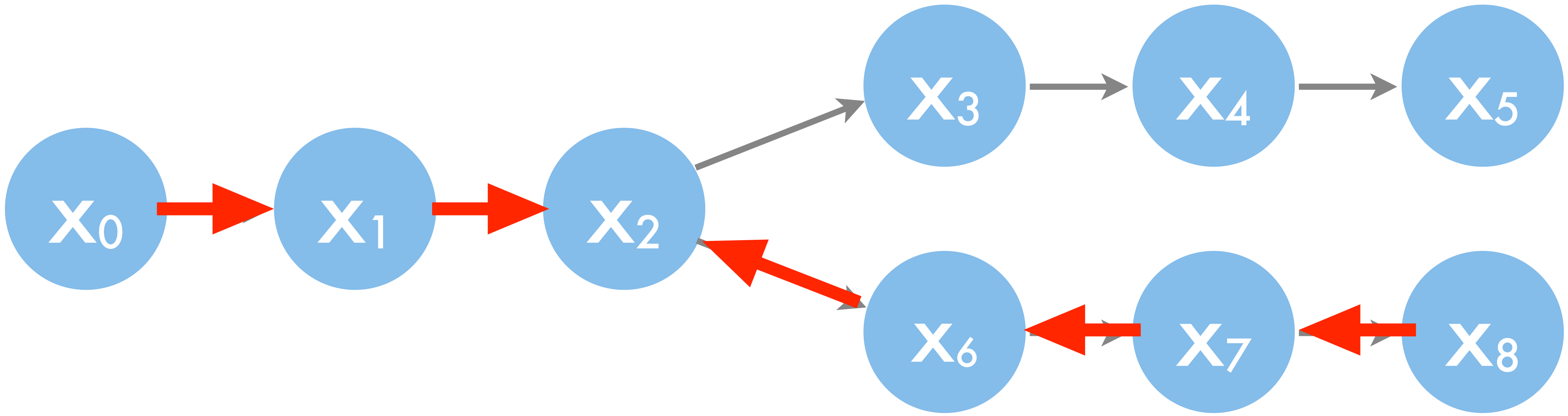
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to 3}(x_3) = \sum_{x_2} m_{1\to 2}(x_2) m_{6\to 2}(x_2) f(x_2, x_3)$$

$$m_{2\to 6}(x_6) = \sum_{x_2} m_{1\to 2}(x_2) m_{3\to 2}(x_2) f(x_2, x_6)$$

$$m_{2\to 1}(x_1) = \sum_{x_2} m_{3\to 2}(x_2) m_{6\to 2}(x_2) f(x_1, x_2)$$

Trees



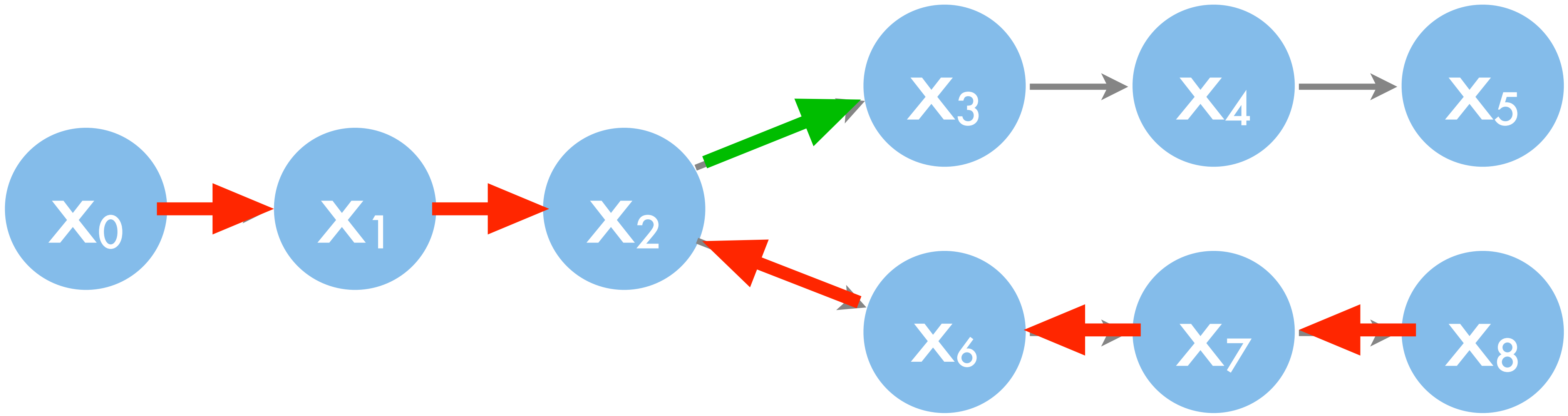
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to3}(x_3) = \sum_{x_2} m_{1\to2}(x_2) m_{6\to2}(x_2) f(x_2, x_3)$$

$$m_{2\to6}(x_6) = \sum_{x_2} m_{1\to2}(x_2) m_{3\to2}(x_2) f(x_2, x_6)$$

$$m_{2\to1}(x_1) = \sum_{x_2} m_{3\to2}(x_2) m_{6\to2}(x_2) f(x_1, x_2)$$

Trees



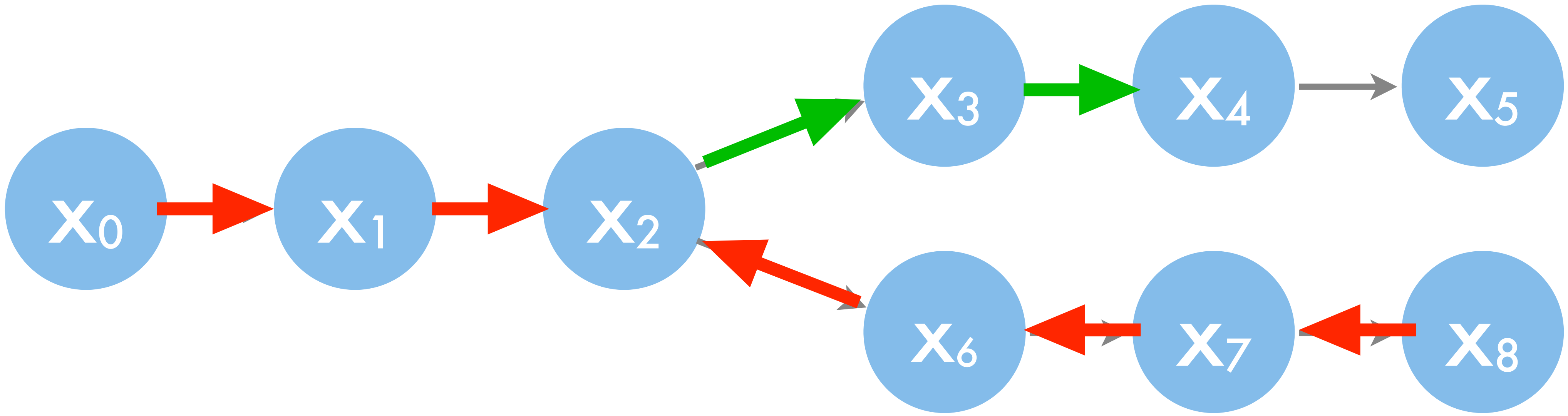
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to 3}(x_3) = \sum_{x_2} m_{1\to 2}(x_2) m_{6\to 2}(x_2) f(x_2, x_3)$$

$$m_{2\to 6}(x_6) = \sum_{x_2} m_{1\to 2}(x_2) m_{3\to 2}(x_2) f(x_2, x_6)$$

$$m_{2\to 1}(x_1) = \sum_{x_2} m_{3\to 2}(x_2) m_{6\to 2}(x_2) f(x_1, x_2)$$

Trees



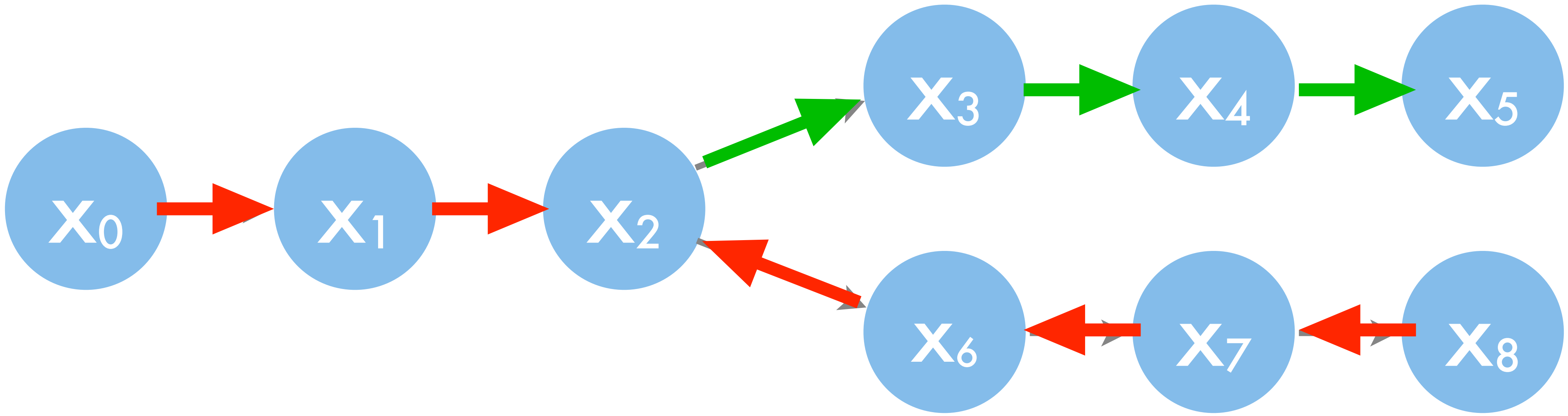
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to 3}(x_3) = \sum_{x_2} m_{1\to 2}(x_2) m_{6\to 2}(x_2) f(x_2, x_3)$$

$$m_{2\to 6}(x_6) = \sum_{x_2} m_{1\to 2}(x_2) m_{3\to 2}(x_2) f(x_2, x_6)$$

$$m_{2\to 1}(x_1) = \sum_{x_2} m_{3\to 2}(x_2) m_{6\to 2}(x_2) f(x_1, x_2)$$

Trees



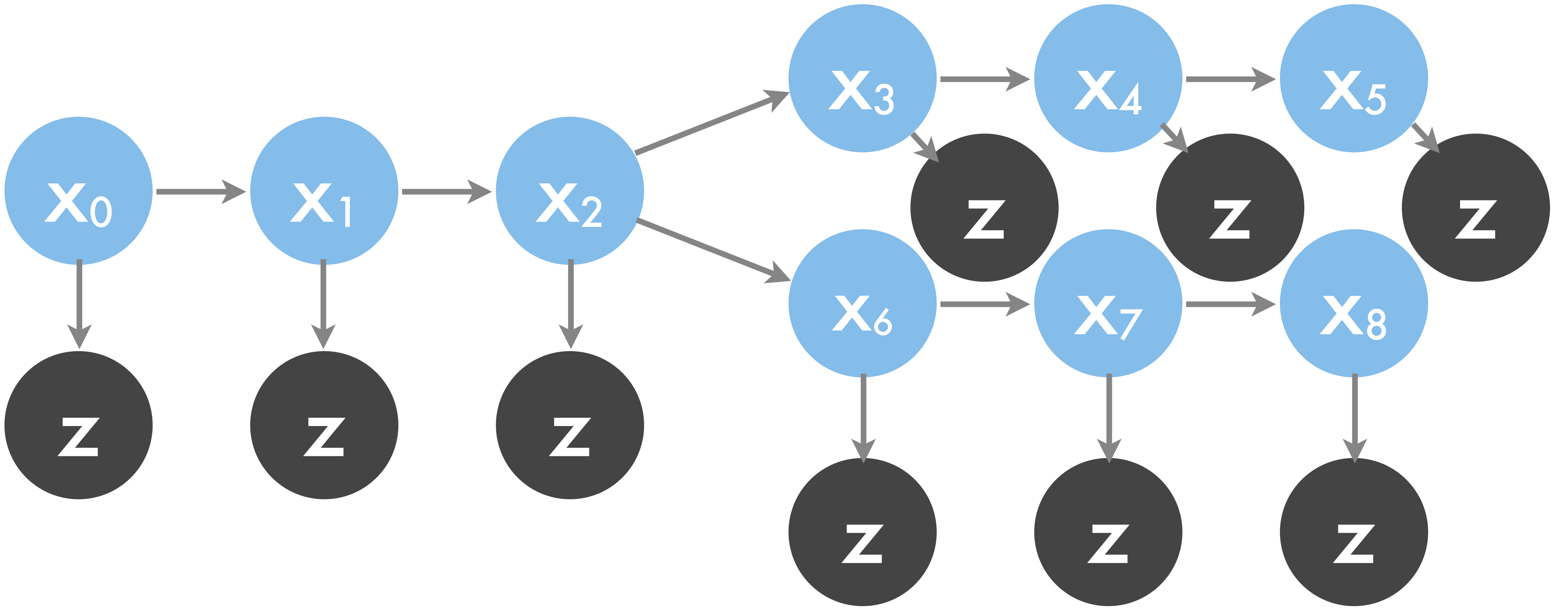
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use ...

$$m_{2\to 3}(x_3) = \sum_{x_2} m_{1\to 2}(x_2) m_{6\to 2}(x_2) f(x_2, x_3)$$

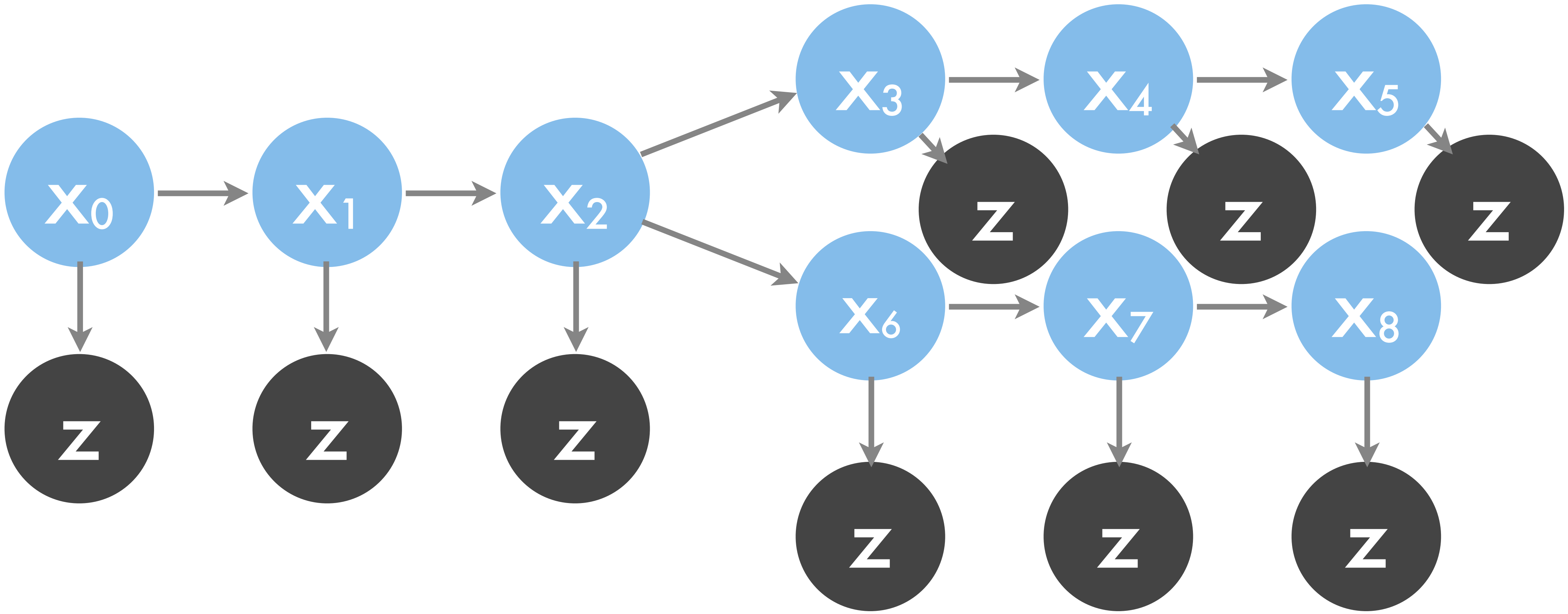
$$m_{2\to 6}(x_6) = \sum_{x_2} m_{1\to 2}(x_2) m_{3\to 2}(x_2) f(x_2, x_6)$$

$$m_{2\to 1}(x_1) = \sum_{x_2} m_{3\to 2}(x_2) m_{6\to 2}(x_2) f(x_1, x_2)$$

Trees



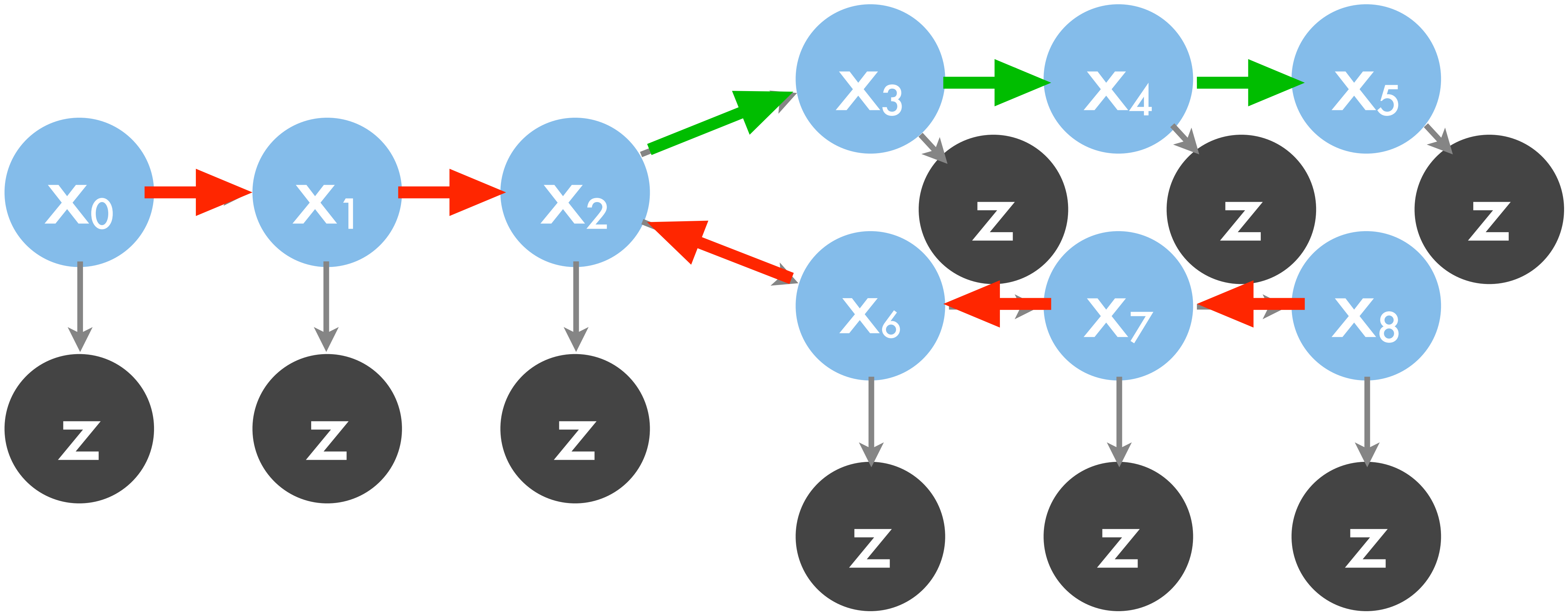
Trees



- Joint distribution over latent state and observations
- To compute conditional probability we need to normalize

$$p(x, z) = p(x) \prod_i p(z_i | x_i) = \prod_{i,j \in T} f(x_i, x_j) \prod_i g(x_i, z_i)$$

Trees

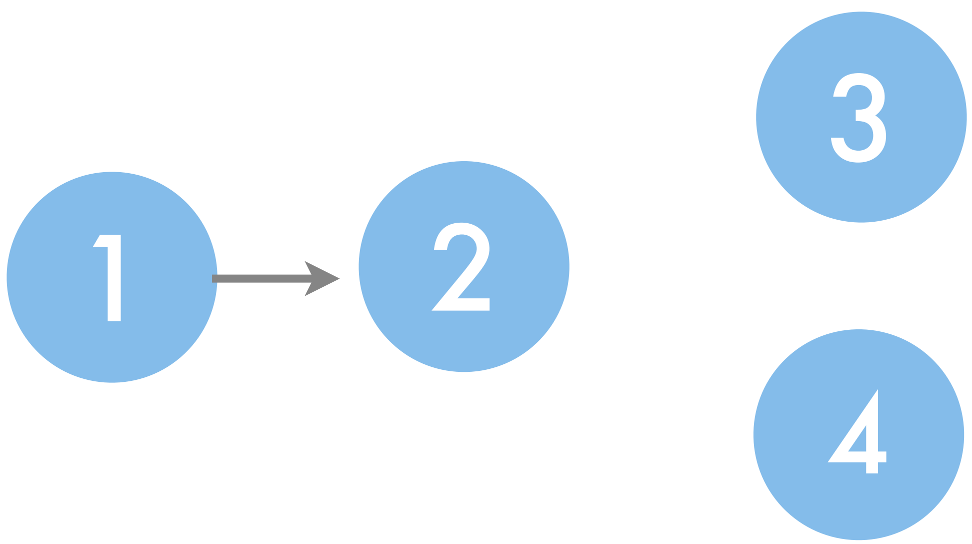


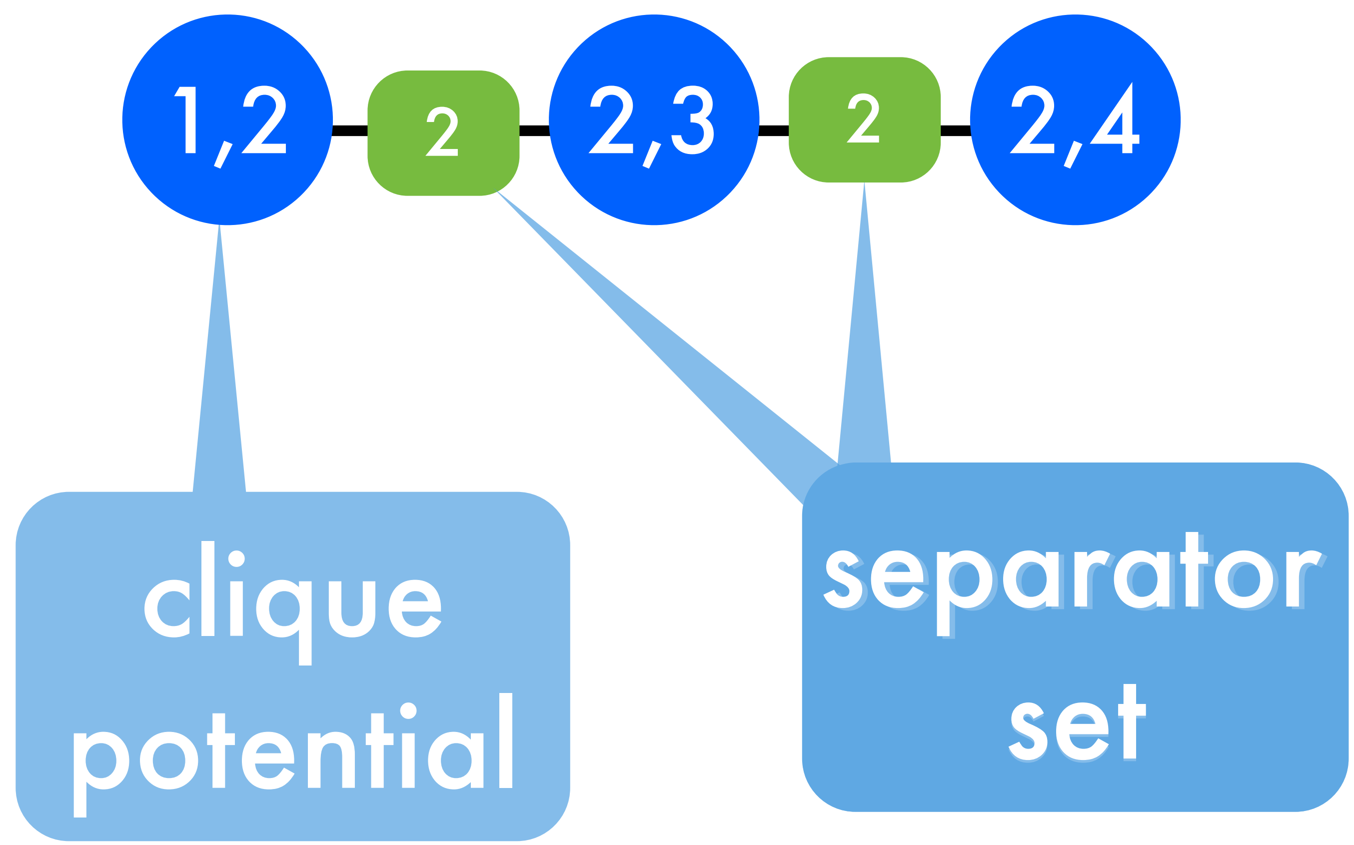
$$p(x_i|\text{rest}) \propto \sum_{x^{-i}} \left[\prod_{j,k \in T} f(x_j, x_k) \prod_j g(x_j) \right]$$
$$= g(x_i) \prod_{(j,i)} m_{j \to i}(x_i)$$

Junction Trees

Junction Trees

$$f(x_1, x_2) f(x_2, x_3) f(x_2, x_4)$$



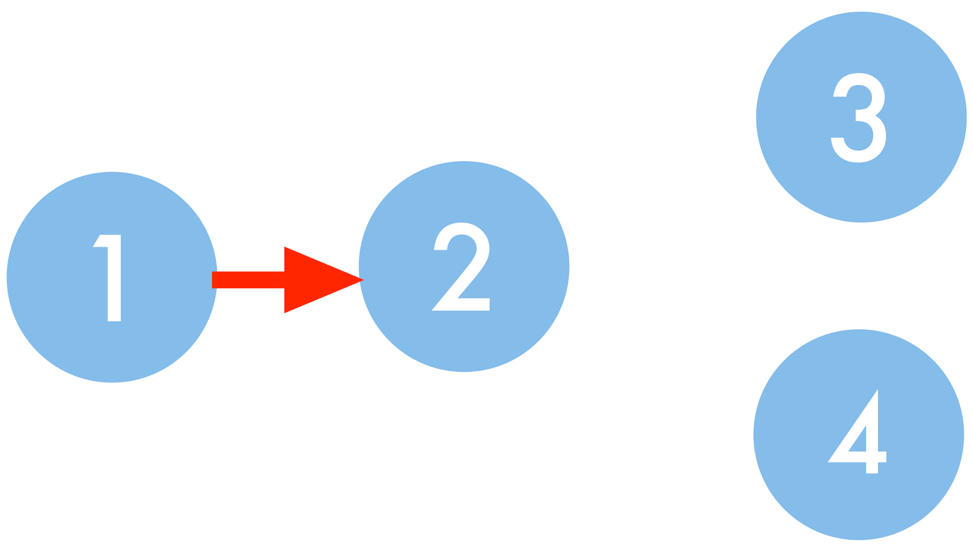


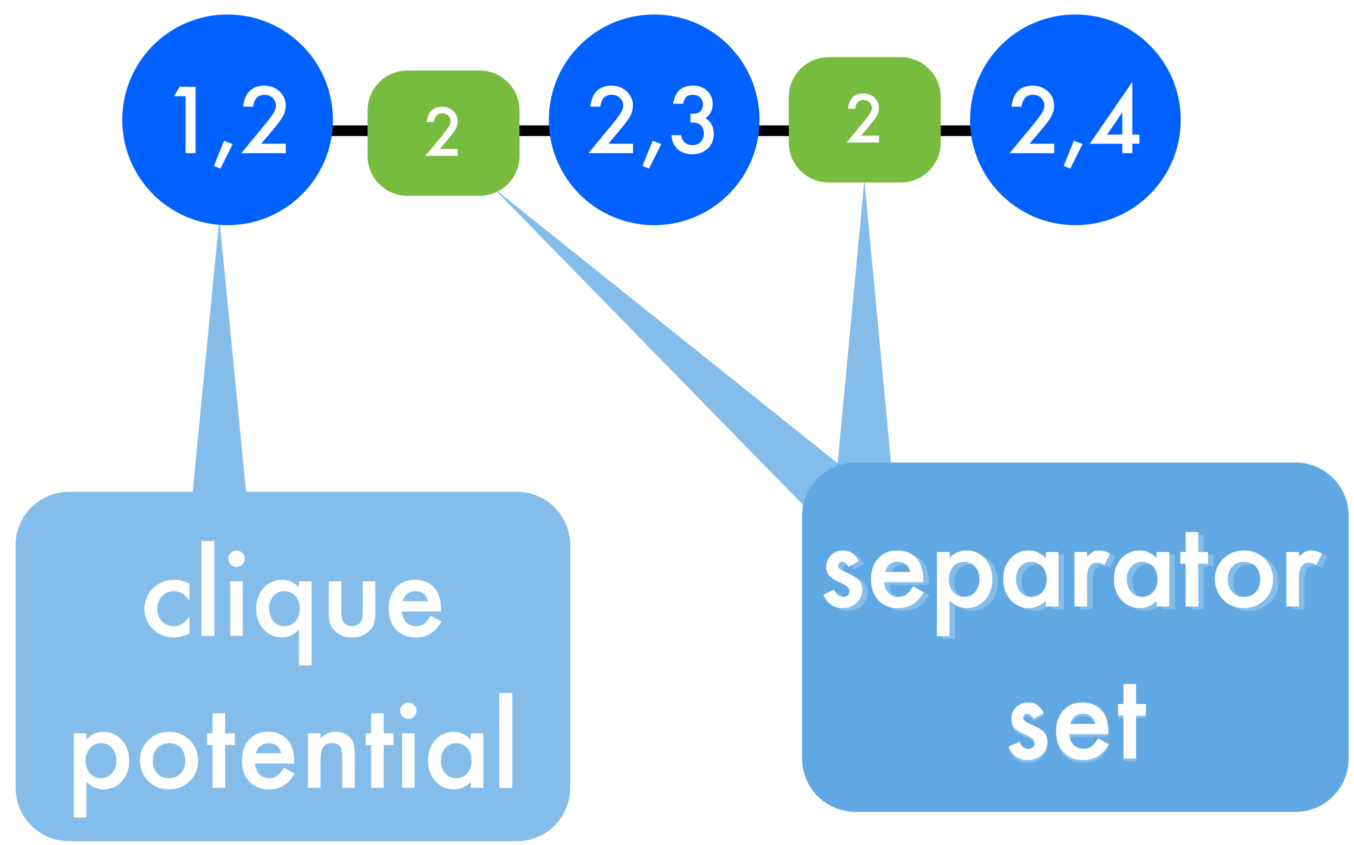
$$m_{i \to j}(x_j) = \sum_{x_i} f(x_i, x_j) \prod_{l \neq j} m_{l \to i}(x_j)$$

clique potential

Junction Trees

$$f(x_1, x_2) f(x_2, x_3) f(x_2, x_4)$$



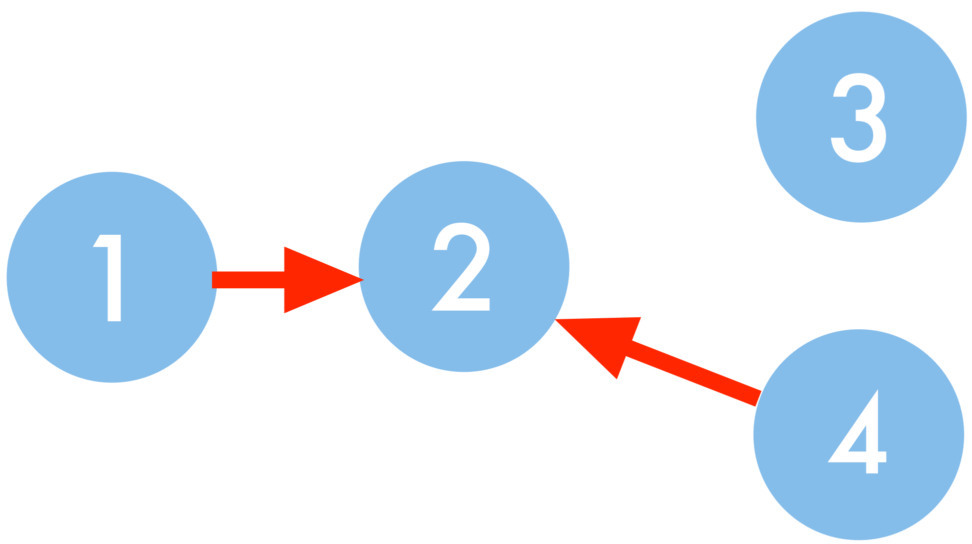


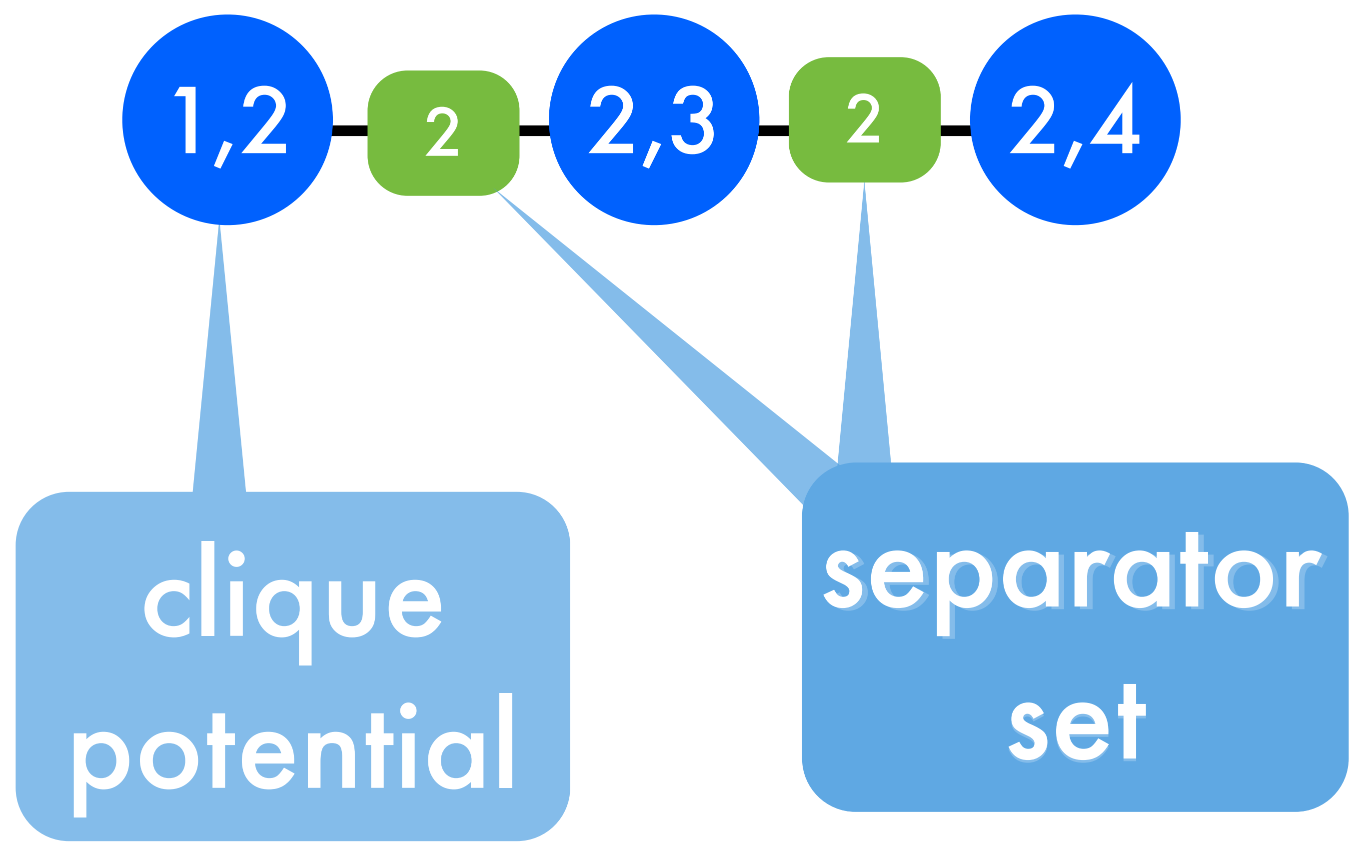
$$m_{i \to j}(x_j) = \sum_{x_i} f(x_i, x_j) \prod_{l \neq j} m_{l \to i}(x_j)$$

clique potential

Junction Trees

$$f(x_1, x_2) f(x_2, x_3) f(x_2, x_4)$$



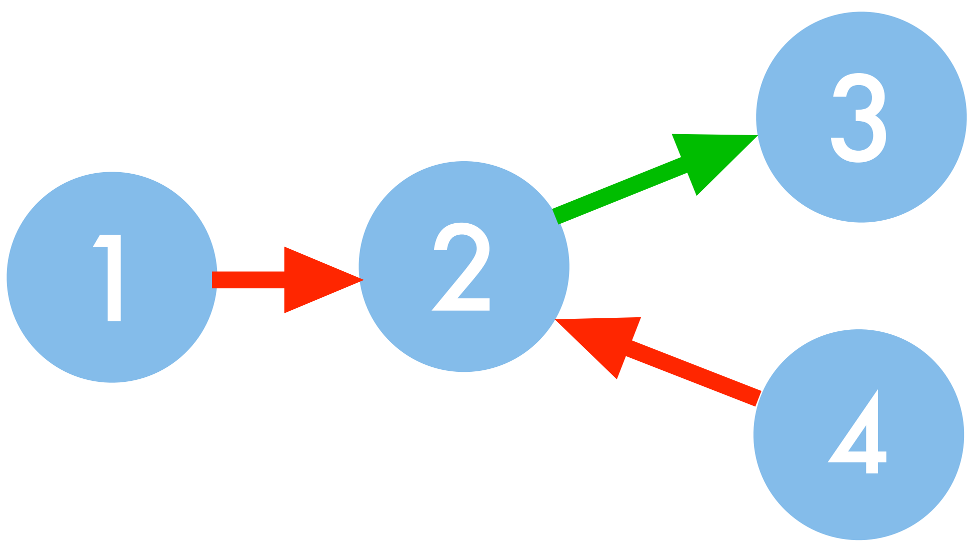


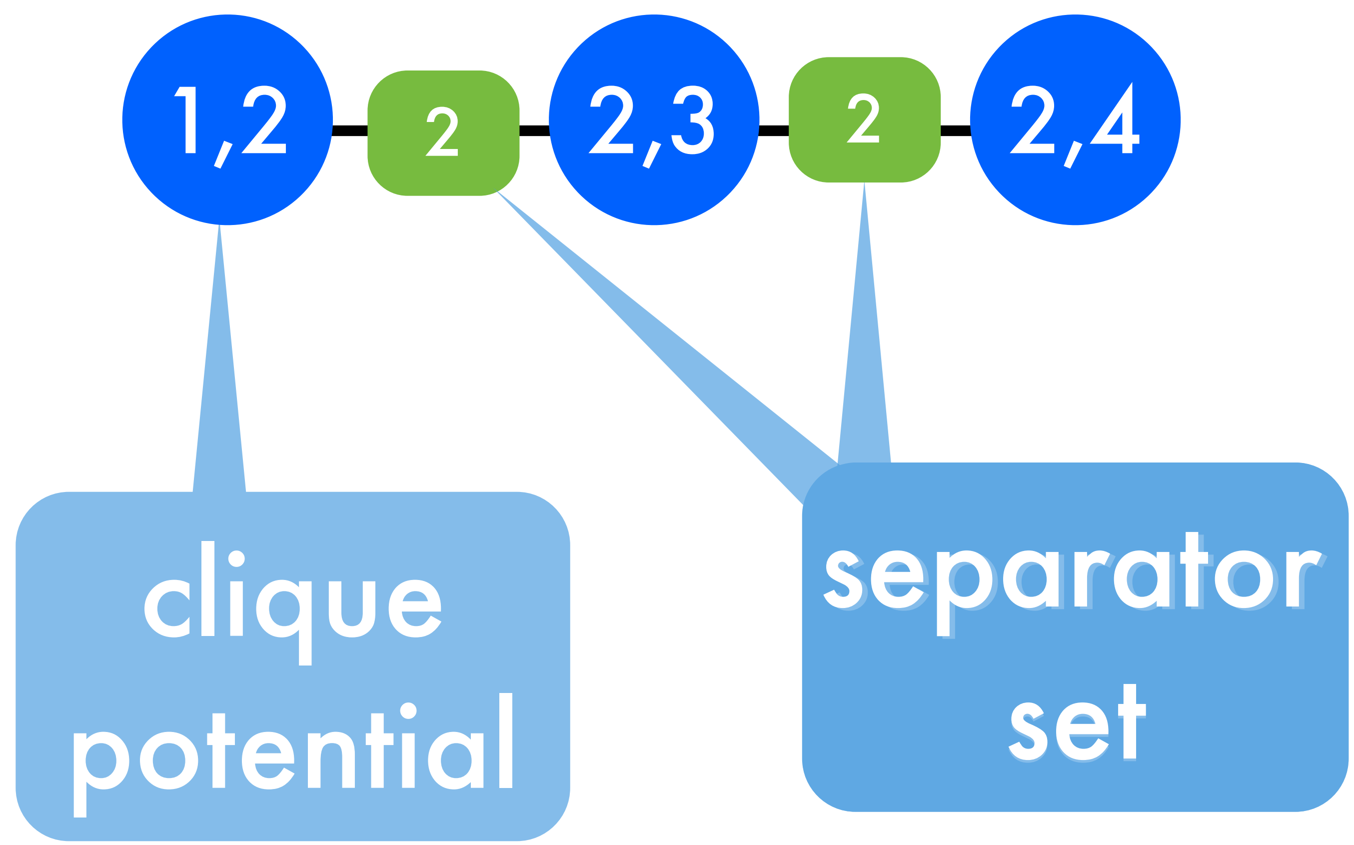
$$m_{i \to j}(x_j) = \sum_{x_i} f(x_i, x_j) \prod_{l \neq j} m_{l \to i}(x_j)$$

clique potential

Junction Trees

$$f(x_1, x_2) f(x_2, x_3) f(x_2, x_4)$$

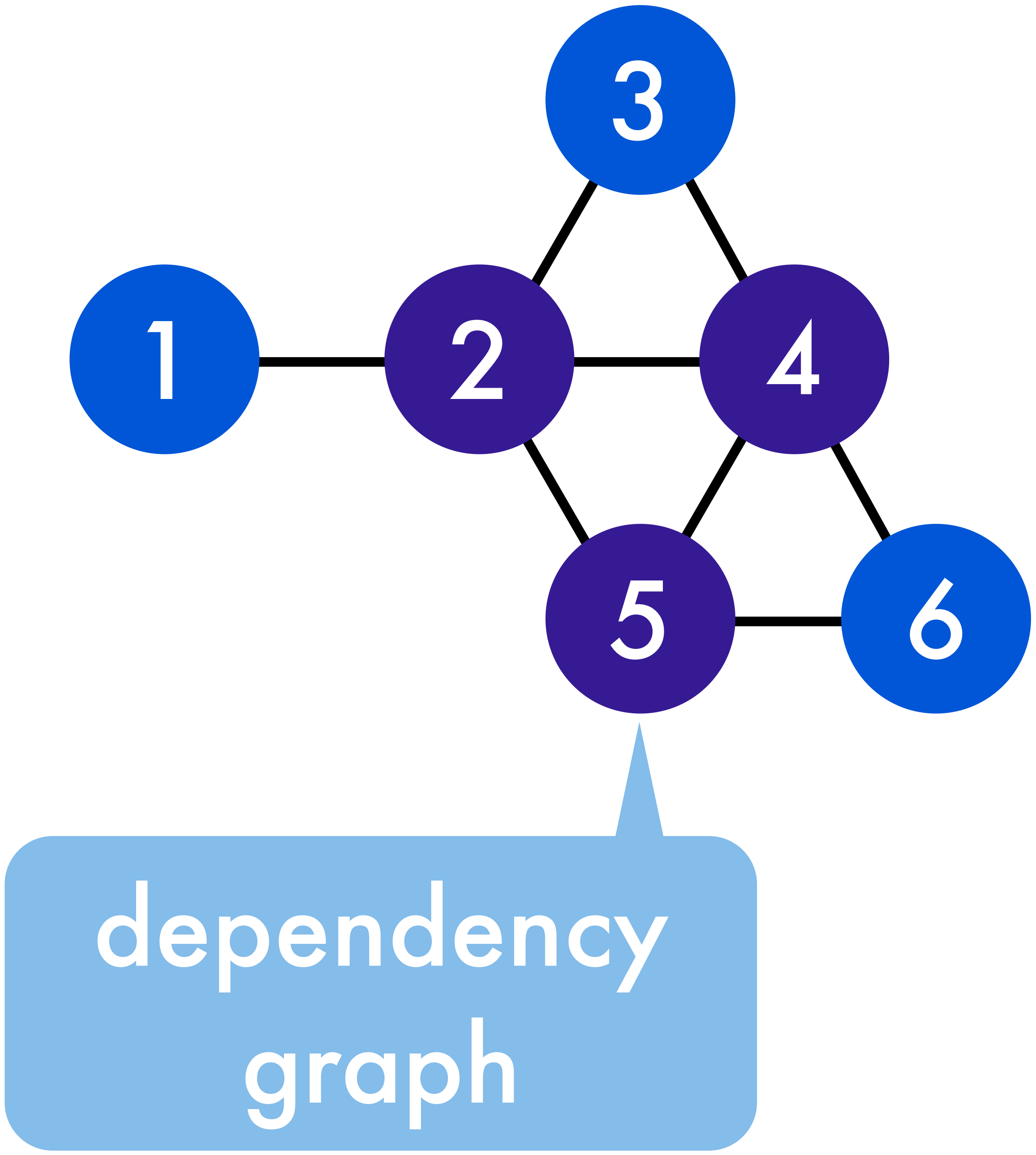




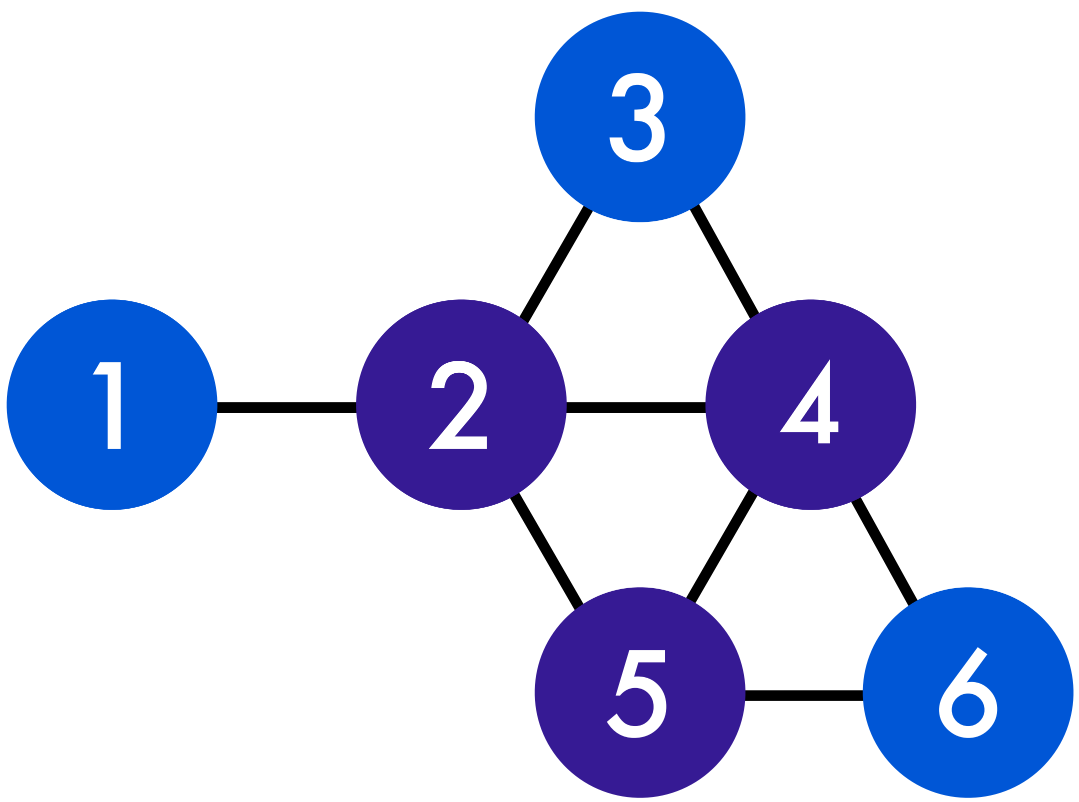
$$m_{i \to j}(x_j) = \sum_{x_i} f(x_i, x_j) \prod_{l \neq j} m_{l \to i}(x_j)$$

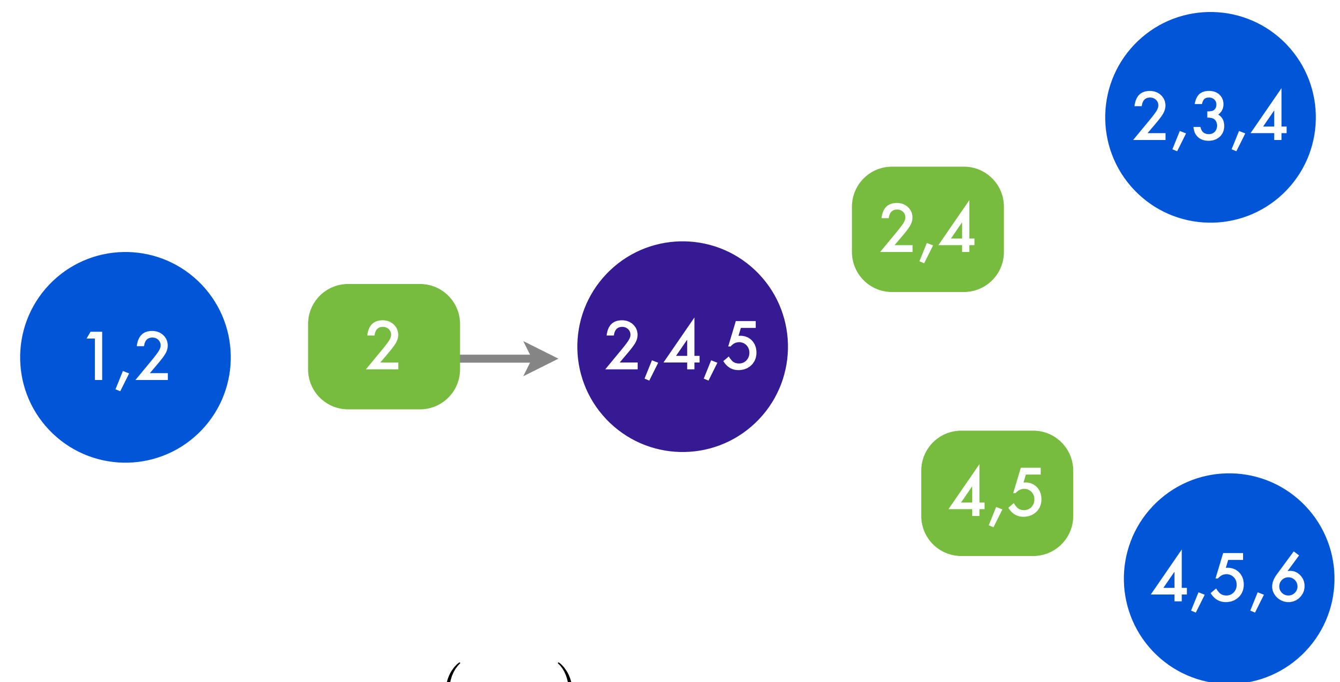
clique potential

Junction Trees



Junction Trees





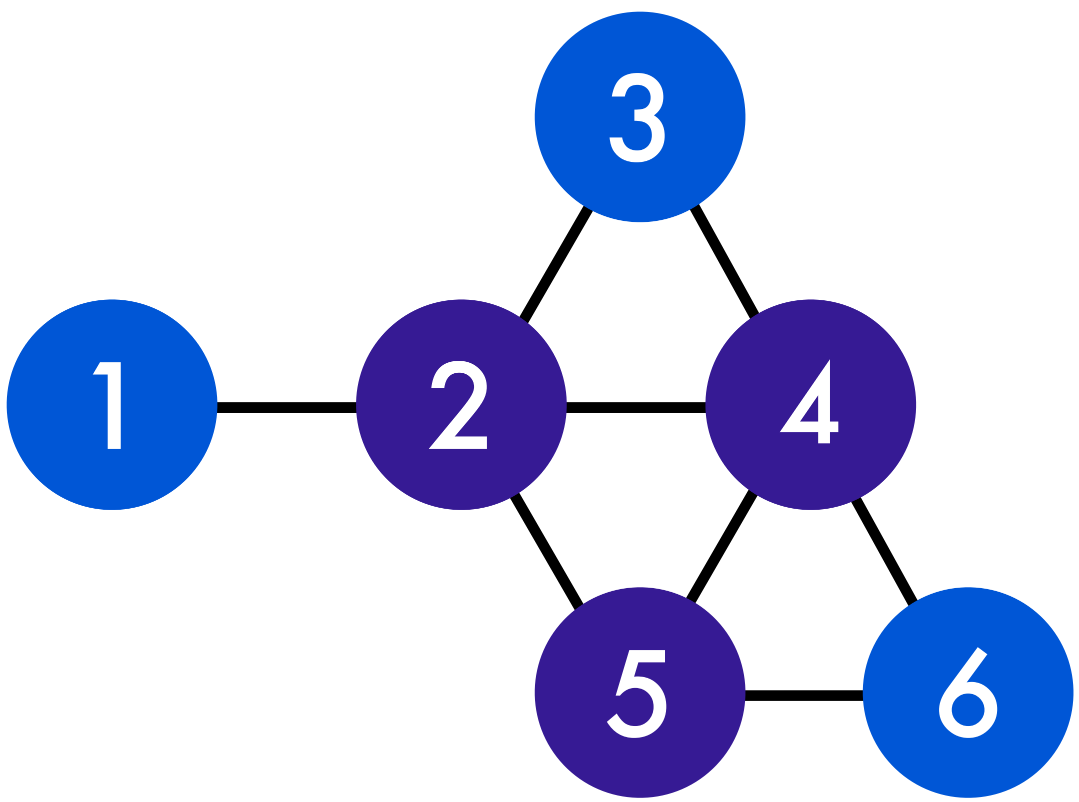
$$m_{245\to 234}(x_{24})$$
$$= \sum_{x_5} f(x_{245}) m_{12\to 245}(x_2) m_{457\to 245}(x_{45})$$

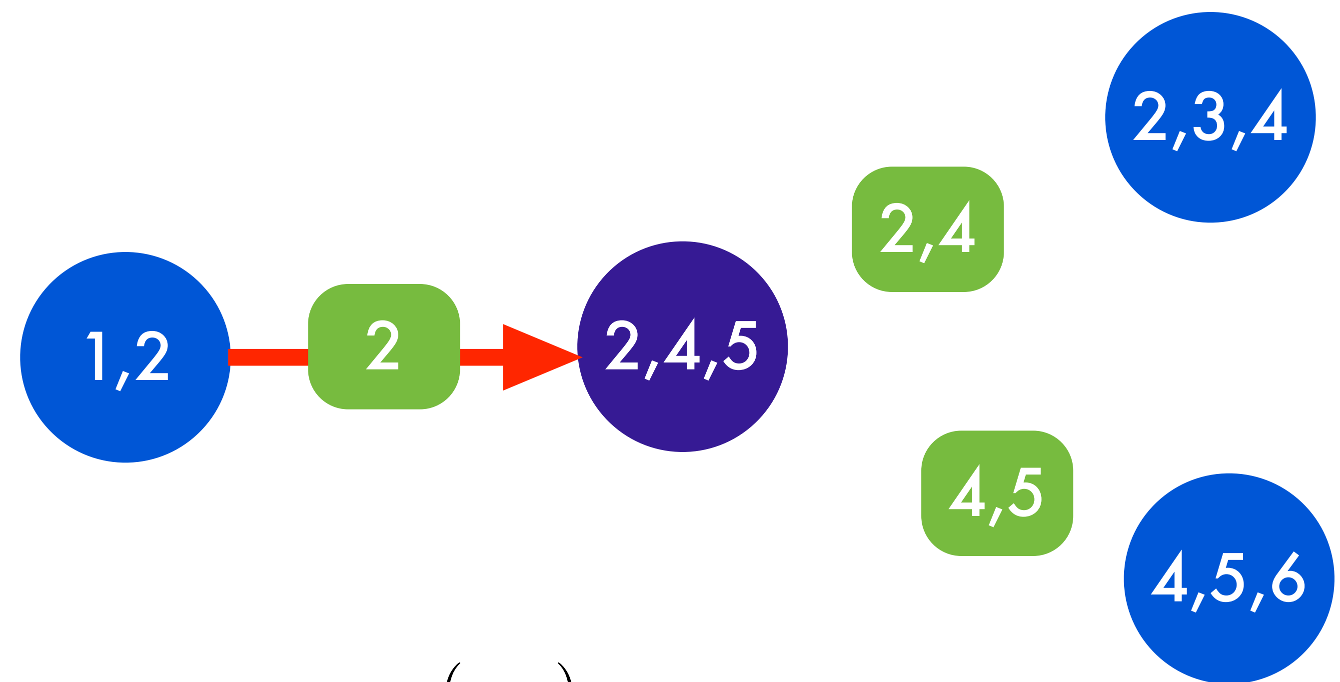
dependency graph

clique potential

separator set

Junction Trees





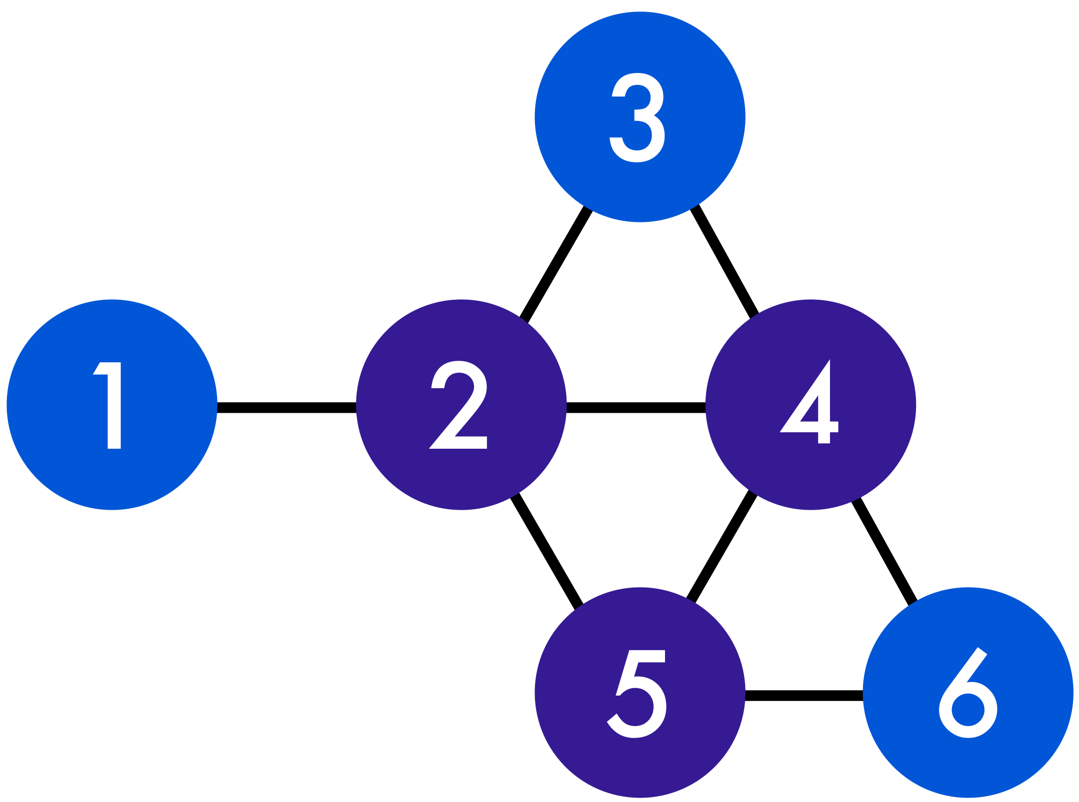
$$m_{245\to 234}(x_{24})$$
$$= \sum_{x_5} f(x_{245}) m_{12\to 245}(x_2) m_{457\to 245}(x_{45})$$

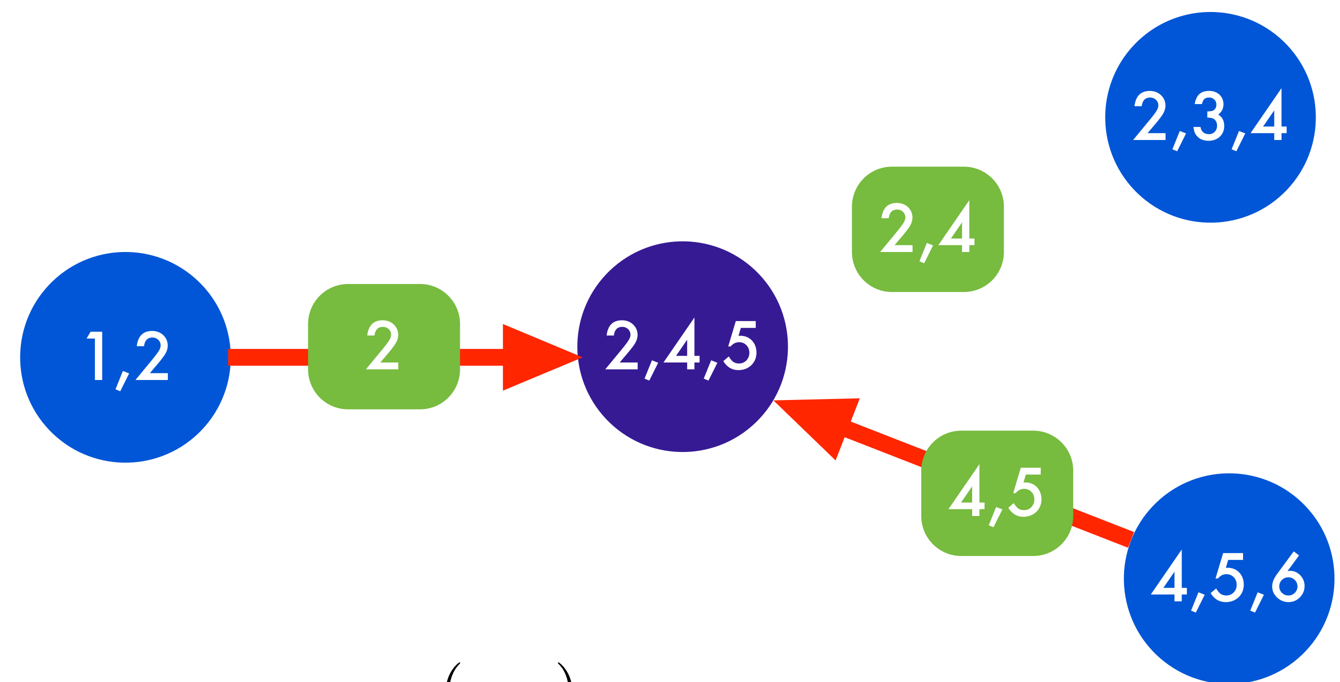
dependency graph

clique potential

separator set

Junction Trees





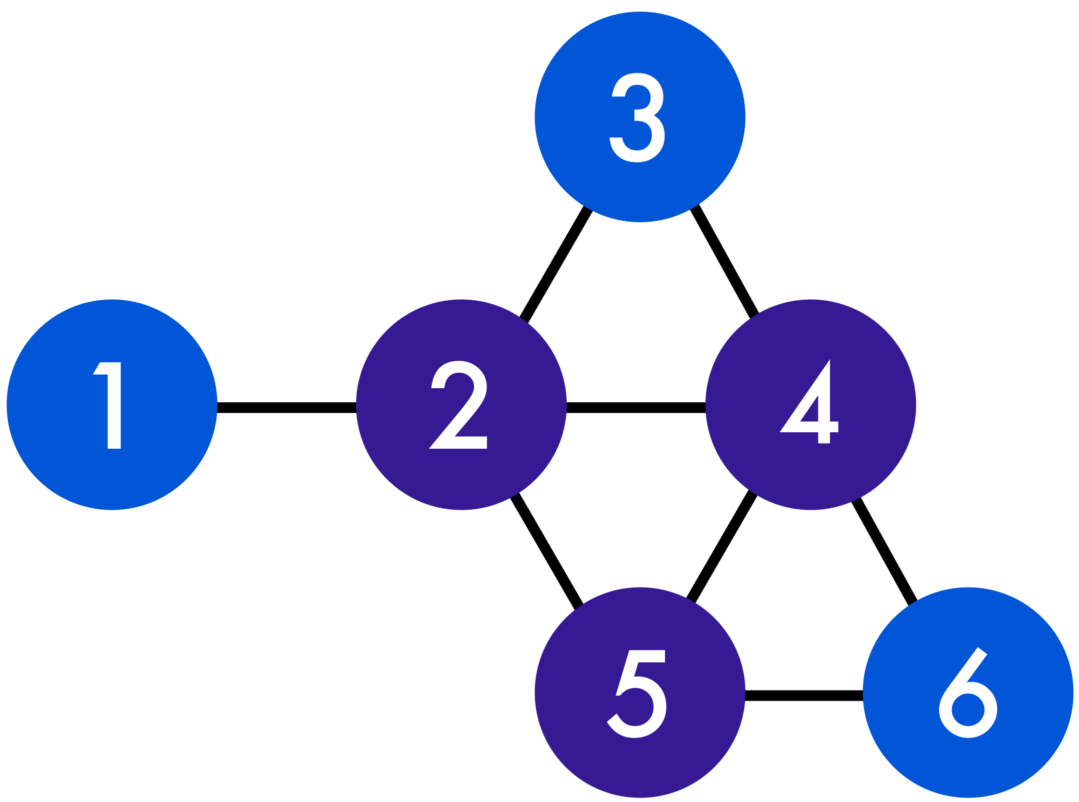
$$m_{245\to 234}(x_{24})$$
$$= \sum_{x_5} f(x_{245}) m_{12\to 245}(x_2) m_{457\to 245}(x_{45})$$

dependency graph

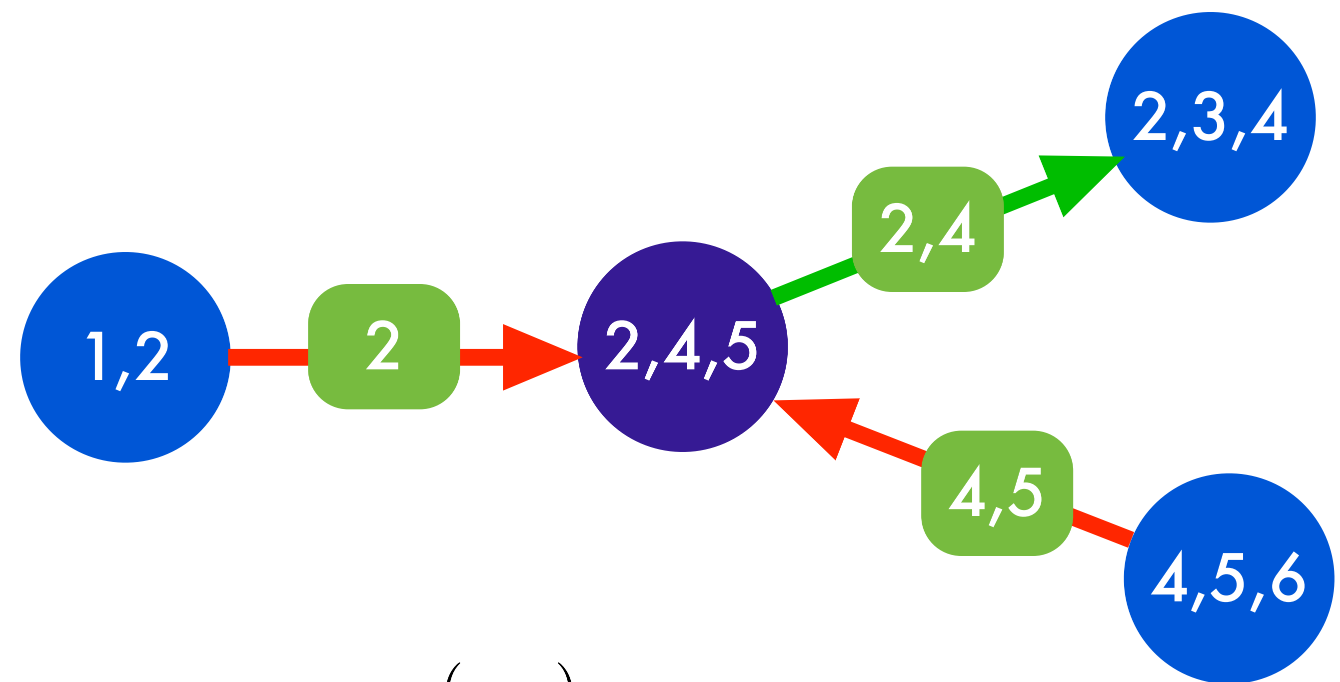
clique potential

separator set

Junction Trees



dependency graph

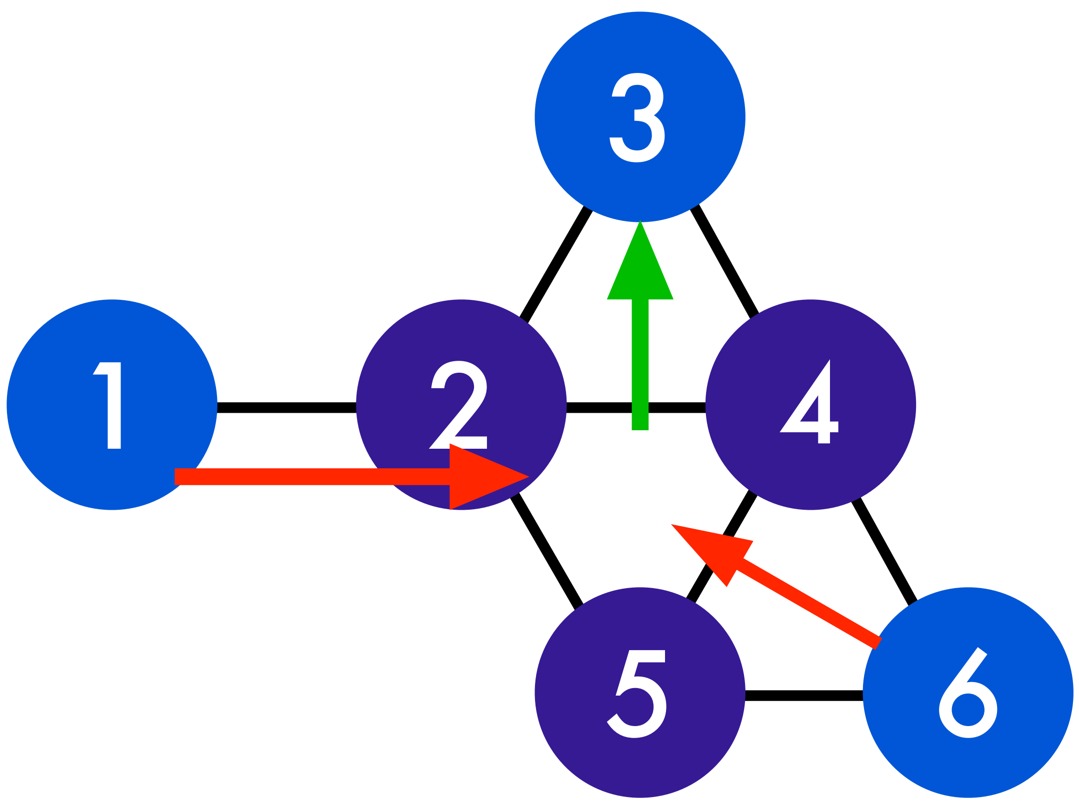


$$m_{245 \to 234}(x_{24}) = \sum_{x_5} f(x_{245}) m_{12 \to 245}(x_2) m_{457 \to 245}(x_{45})$$

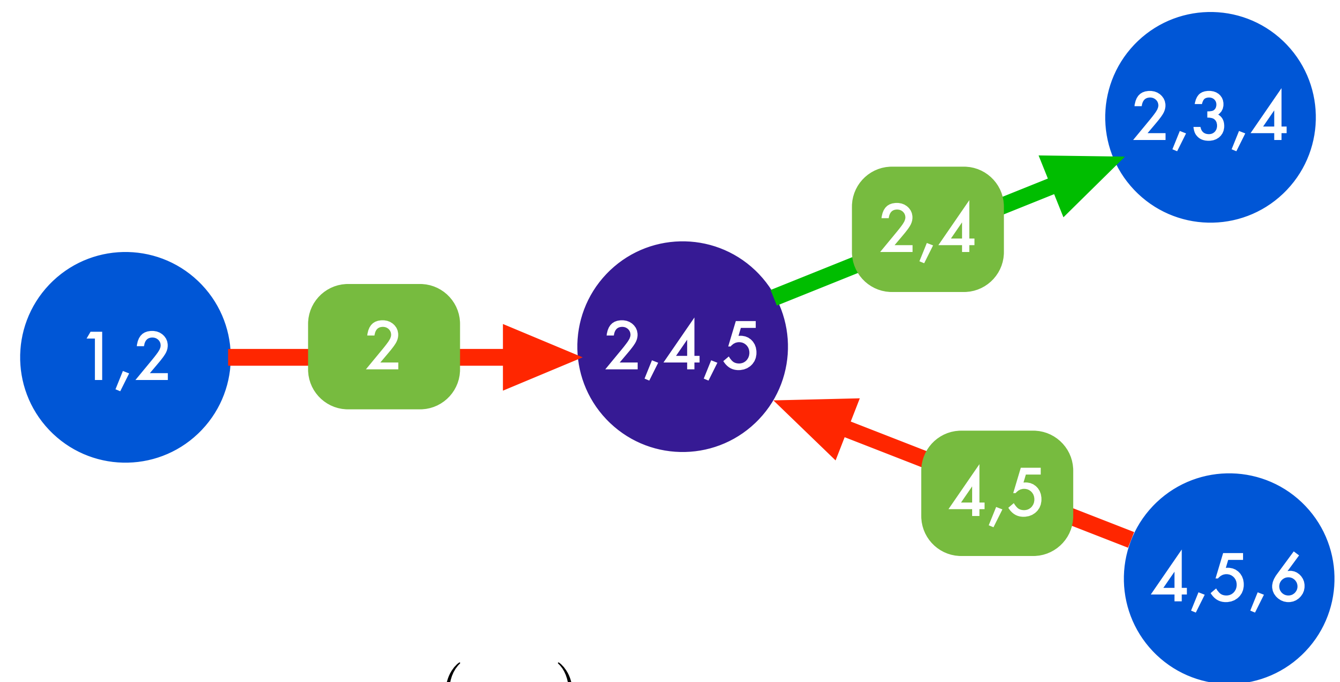
clique potential

separator set

Junction Trees



dependency graph

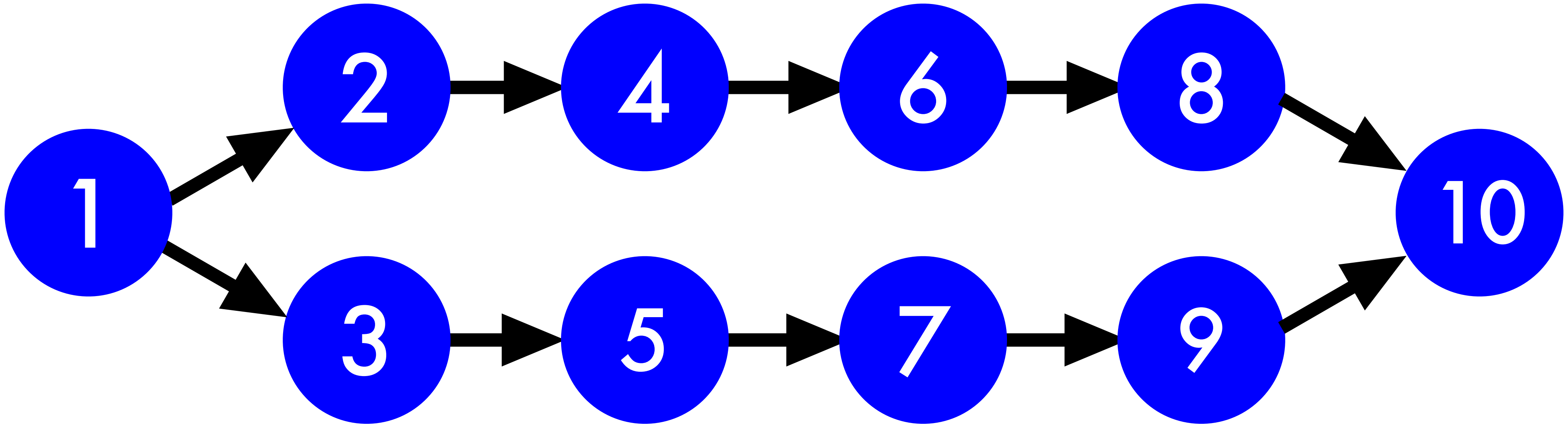


$$m_{245\to 234}(x_{24}) = \sum_{x_5} f(x_{245}) m_{12\to 245}(x_2) m_{457\to 245}(x_{45})$$

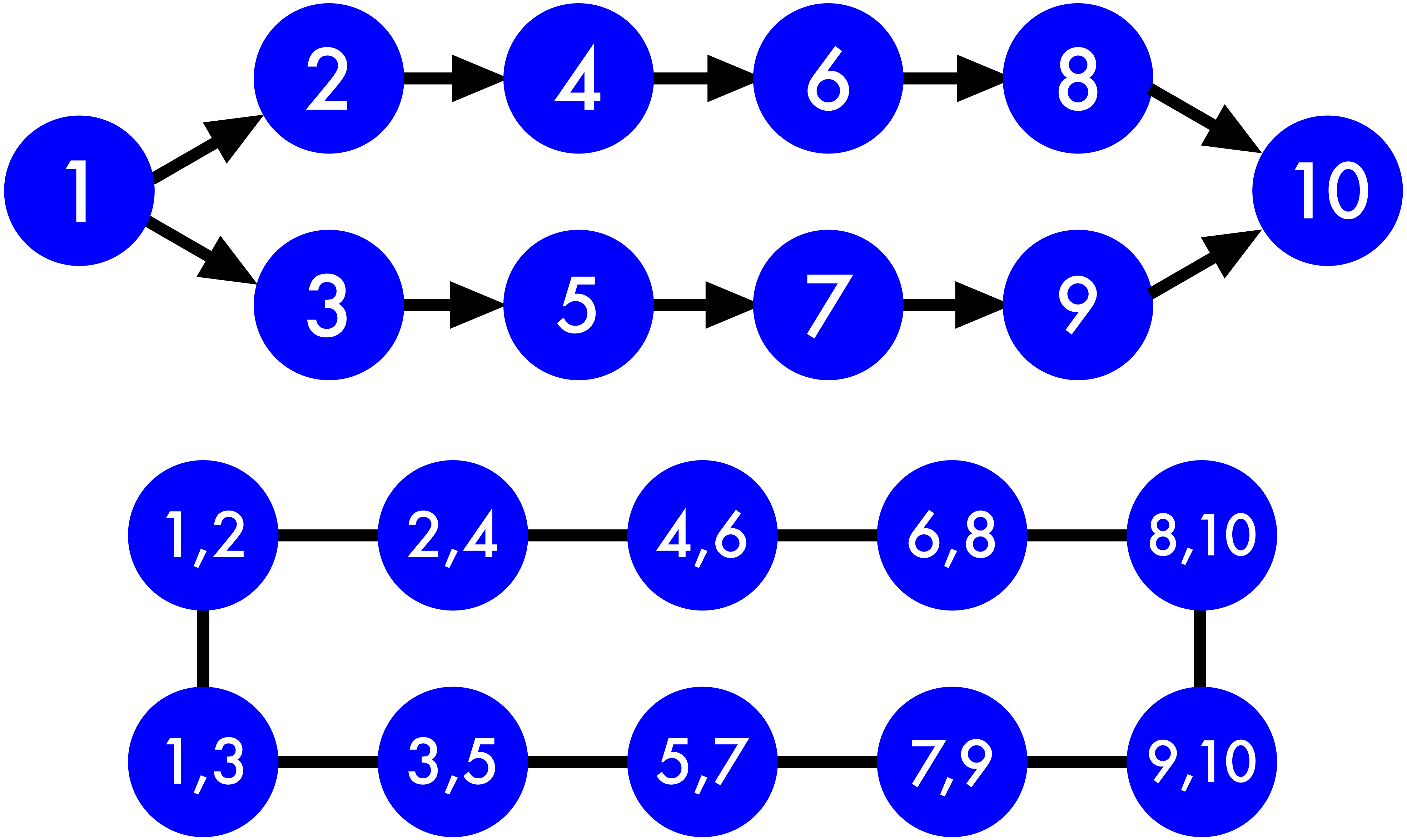
clique potential

separator set

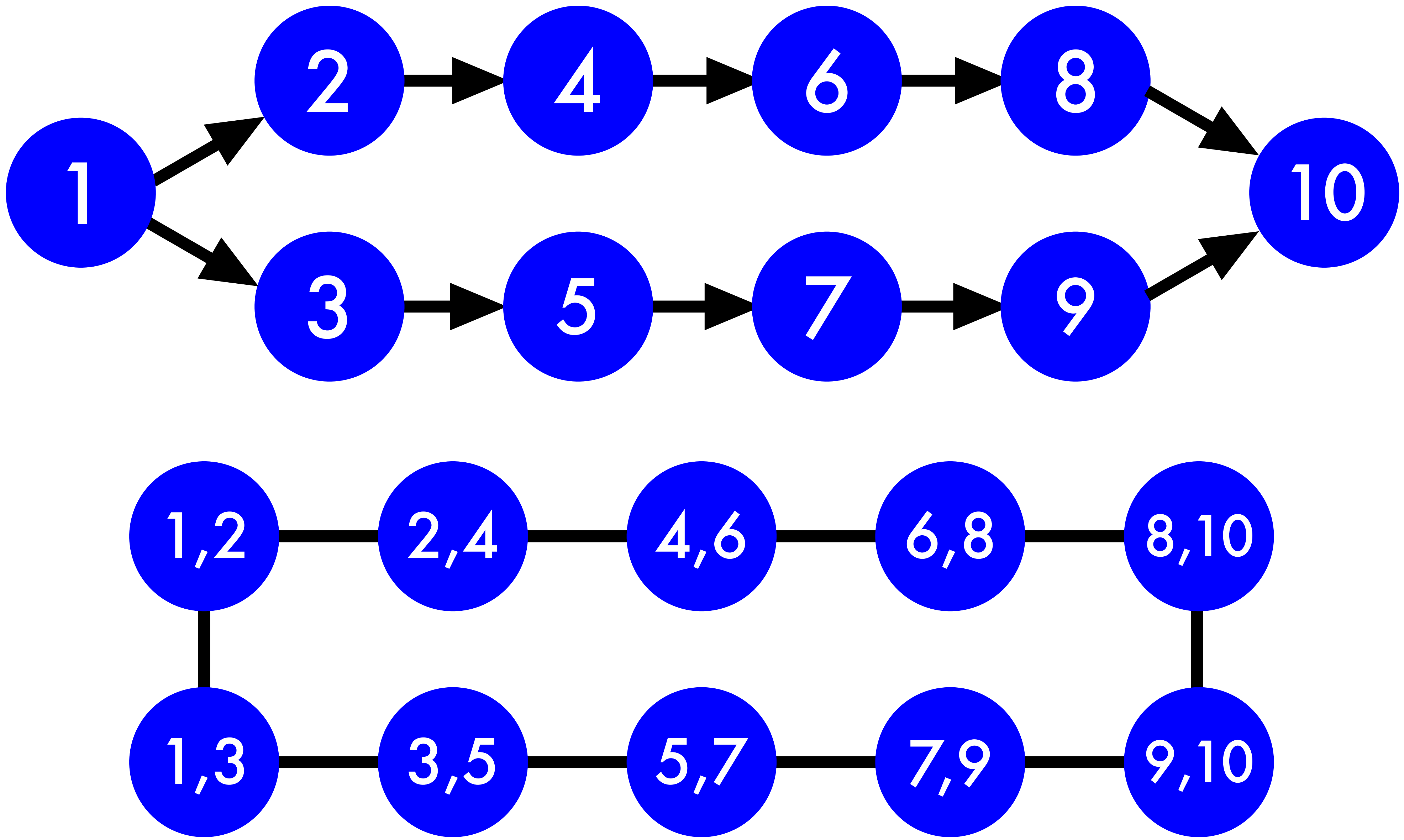
Caution



Caution

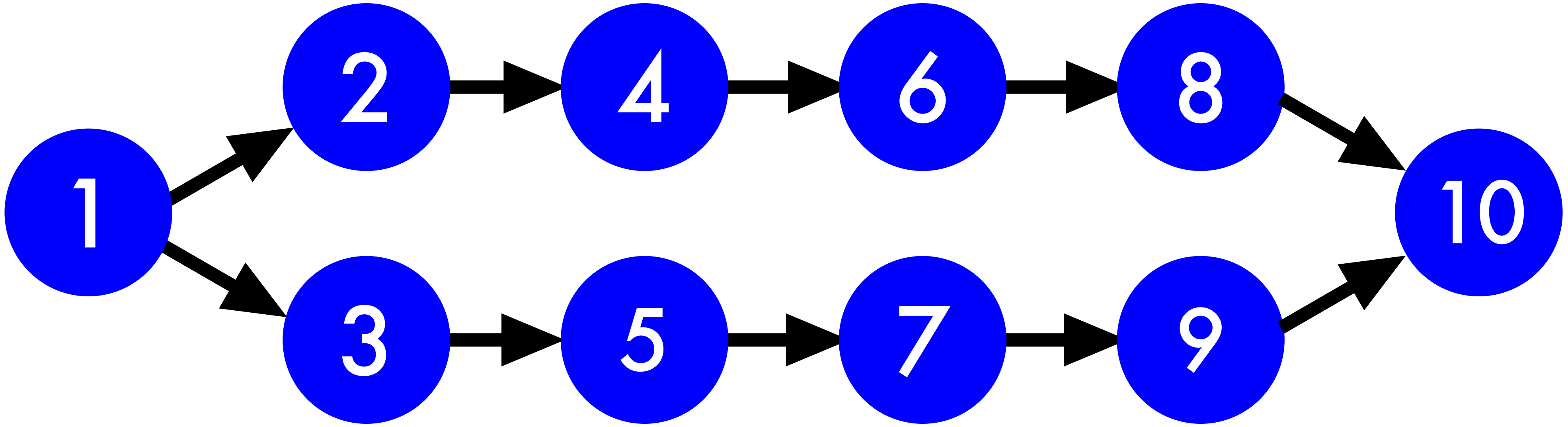


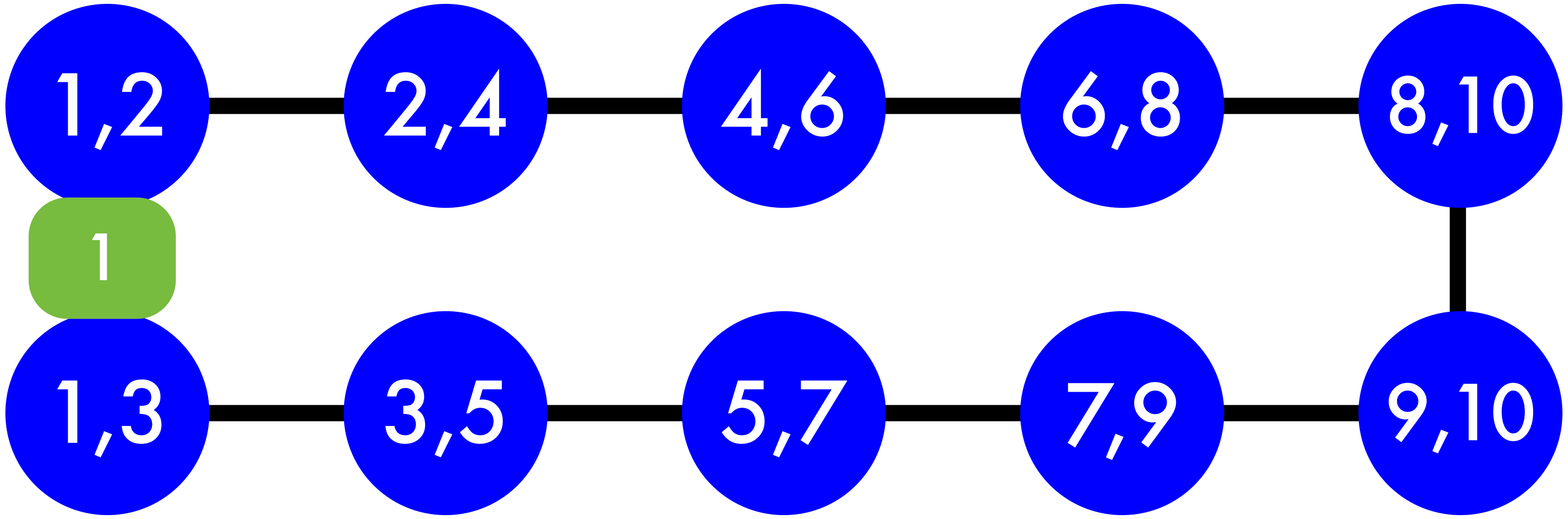
Caution



This is not a tree

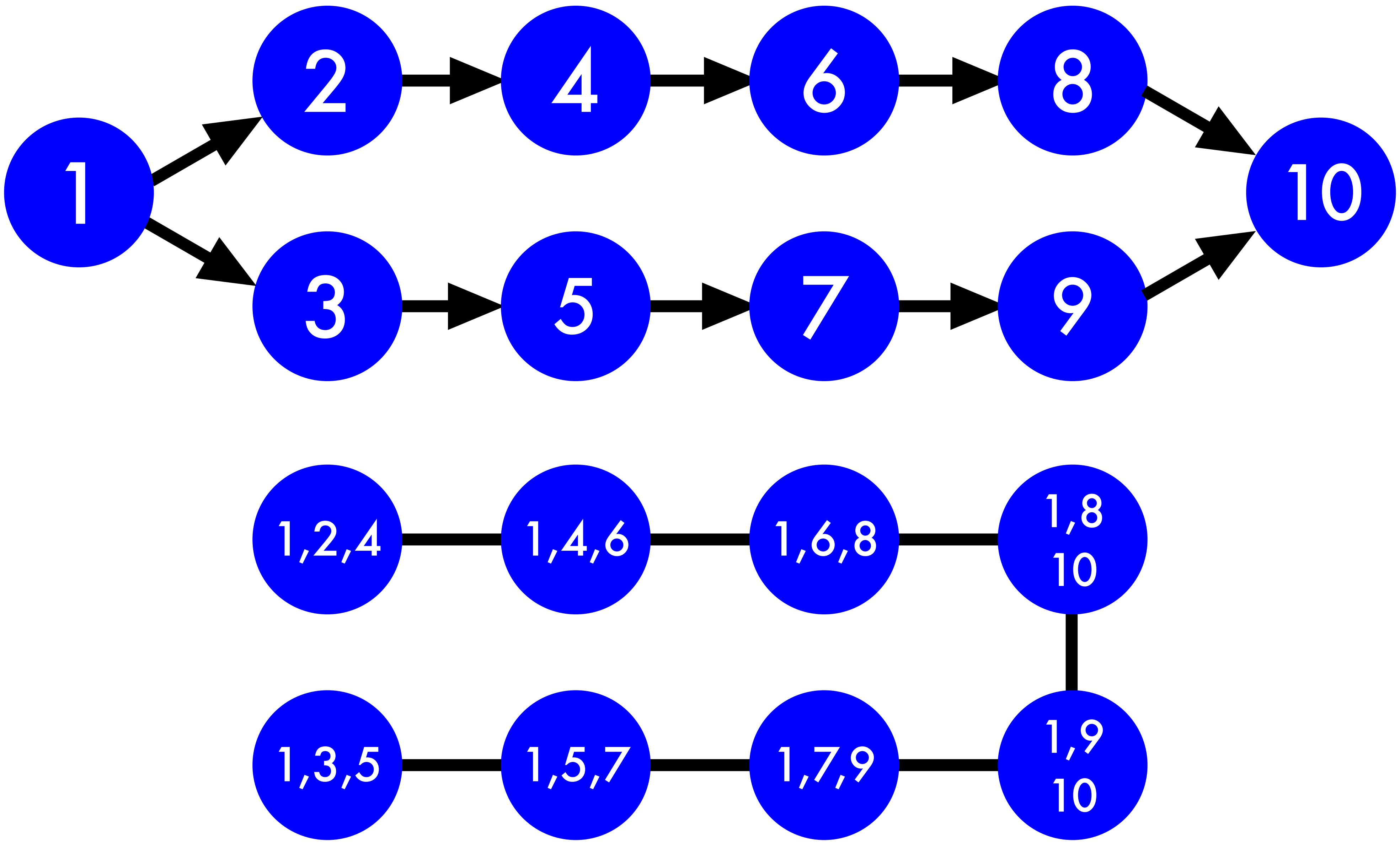
Caution



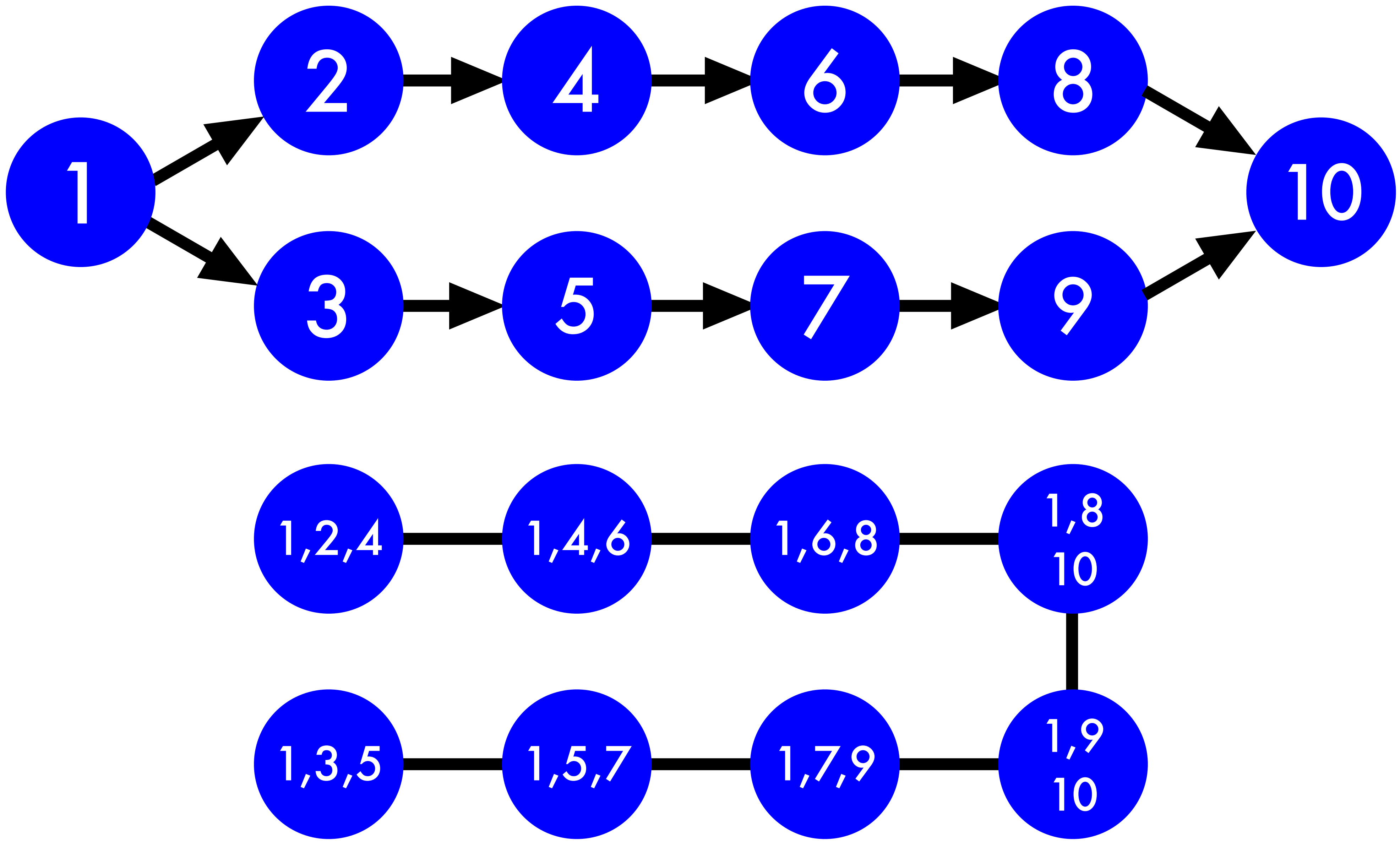


This is not a tree

Graph triangulation

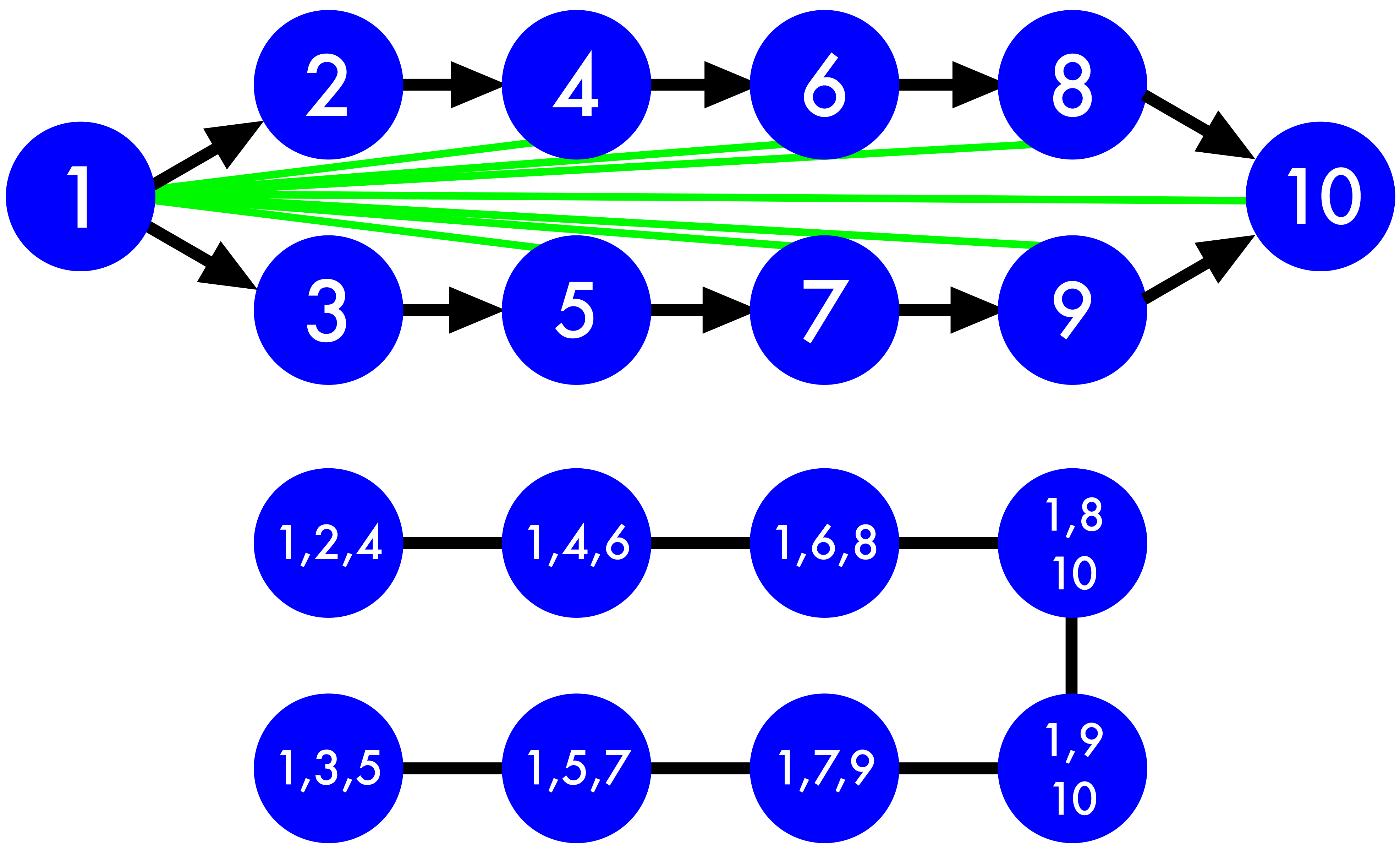


Graph triangulation



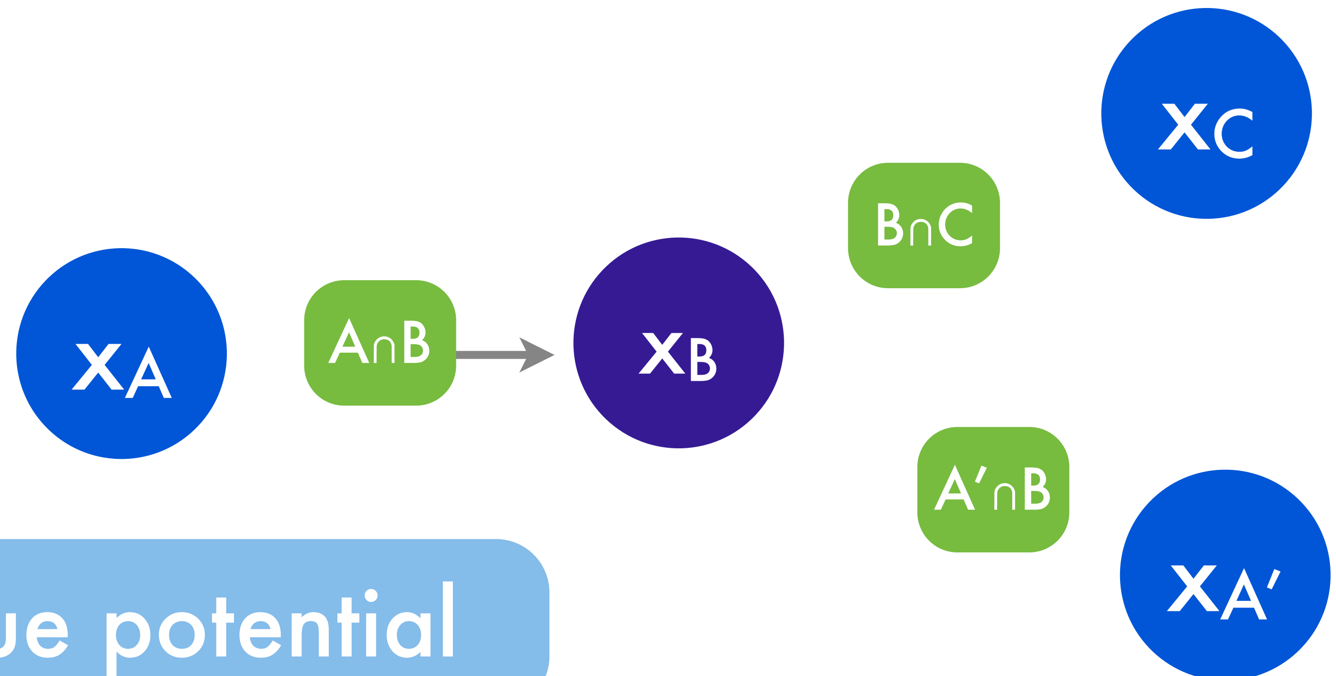
Separator set increases

Graph triangulation



Separator set increases

Update equations



incoming messages

clique potential

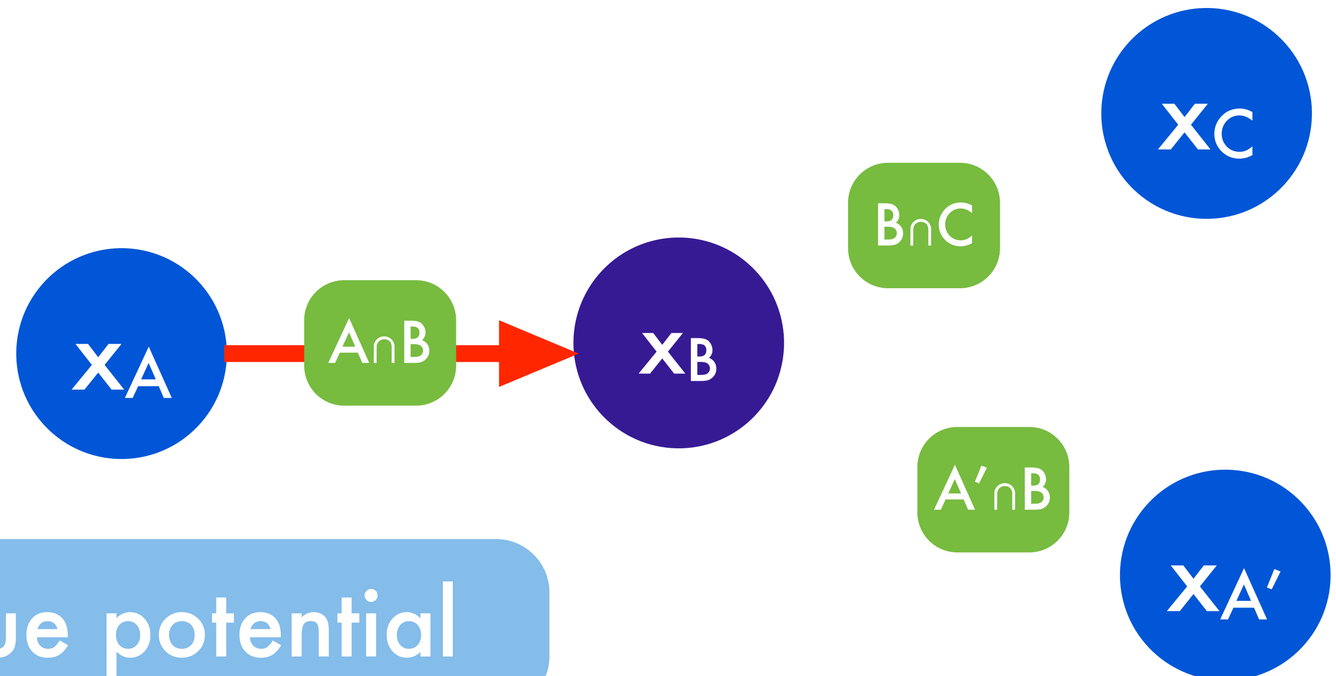
$$m_{B\to C}(x_{B\cap C}) = \sum_{x_{B\setminus C}} f(x_B) \prod_{A\neq C|A\sim B} m_{A\to B}(x_{A\cap B})$$

all but separator set

$$p(x_B) \propto f(x_B) \prod_{A\sim B} m_{A\to B}(x_{A\cap B})$$

unnormalized

Update equations



incoming messages

clique potential

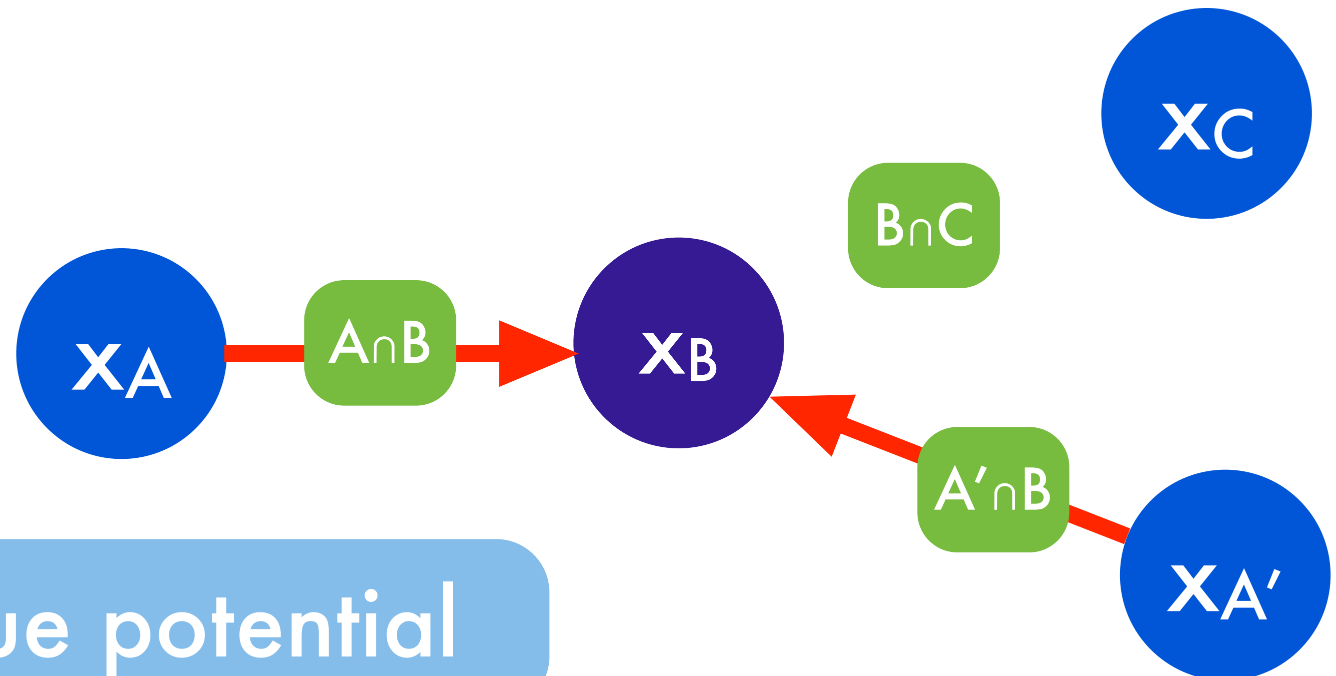
$$m_{B \to C}(x_{B \cap C}) = \sum_{x_{B \setminus C}} f(x_B) \prod_{A \neq C | A \sim B} m_{A \to B}(x_{A \cap B})$$

all but separator set

$$p(x_B) \propto f(x_B) \prod_{A \sim B} m_{A \to B}(x_{A \cap B})$$

unnormalized

Update equations



incoming messages

clique potential

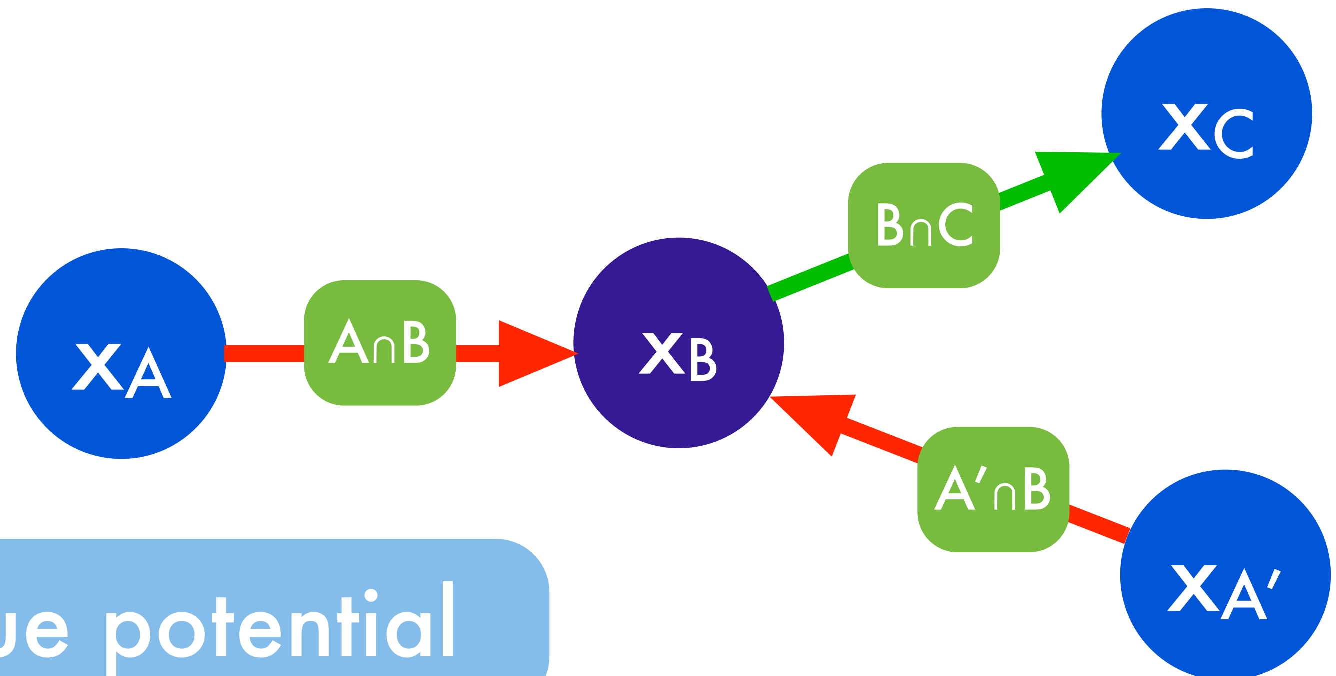
$$m_{B \to C}(x_{B \cap C}) = \sum_{x_{B \setminus C}} f(x_B) \prod_{A \neq C | A \sim B} m_{A \to B}(x_{A \cap B})$$

all but separator set

$$p(x_B) \propto f(x_B) \prod_{A \sim B} m_{A \to B}(x_{A \cap B})$$

unnormalized

Update equations



incoming messages

clique potential

$$m_{B\to C}\left(x_{B\cap C}\right) = \sum_{x_{B\setminus C}} f(x_B) \prod_{A\neq C | A\sim B} m_{A\to B}(x_{A\cap B})$$

all but separator set

$$p(x_B) \propto f(x_B) \prod_{A\sim B} m_{A\to B}(x_{A\cap B})$$

unnormalized

Generalize Distributive Law

Generalized Distributive Law

- Key Idea
  Dynamic programming uses only sums and multiplications, hence replace them with equivalent operations from other semirings
- Semiring
  - ‘addition’ and ‘summation’ equivalent
  - Associative law $(a+b)+c = a+(b+c)$
  - Distributive law $a(b+c) = ab+ac$

Generalized Distributive Law

- Integrating out probabilities (sum, product)

$$a \cdot (b + c) = a \cdot b + a \cdot c$$

- Finding the maximum (max, +)

$$a + \max(b, c) = \max(a + b, a + c)$$

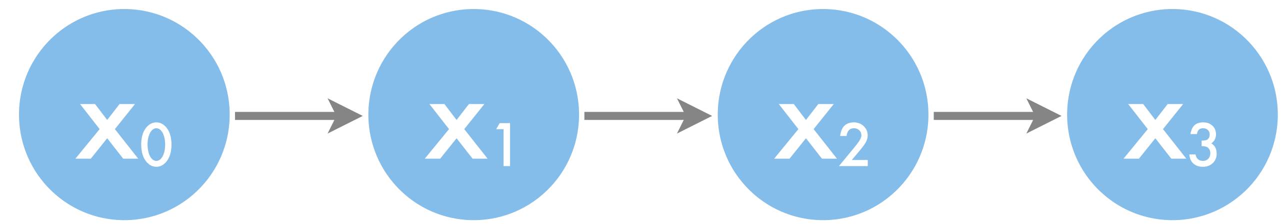
- Set algebra (union, intersection)

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$$

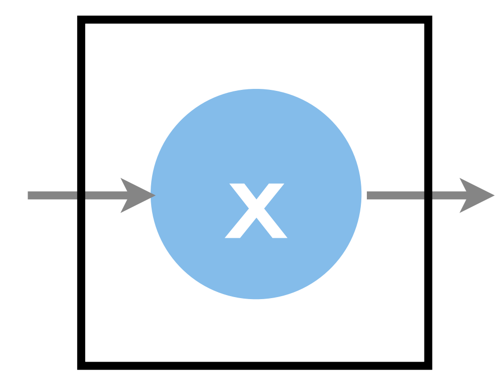
- Boolean semiring (AND, OR)
- Probability semiring (log +, +)
- Tropical semiring (min, +)

Chains ... again

$$\bar{s} = \max_{x} s(x_0) + \sum_{i=1}^{n-1} s(x_{i+1}|x_i)$$



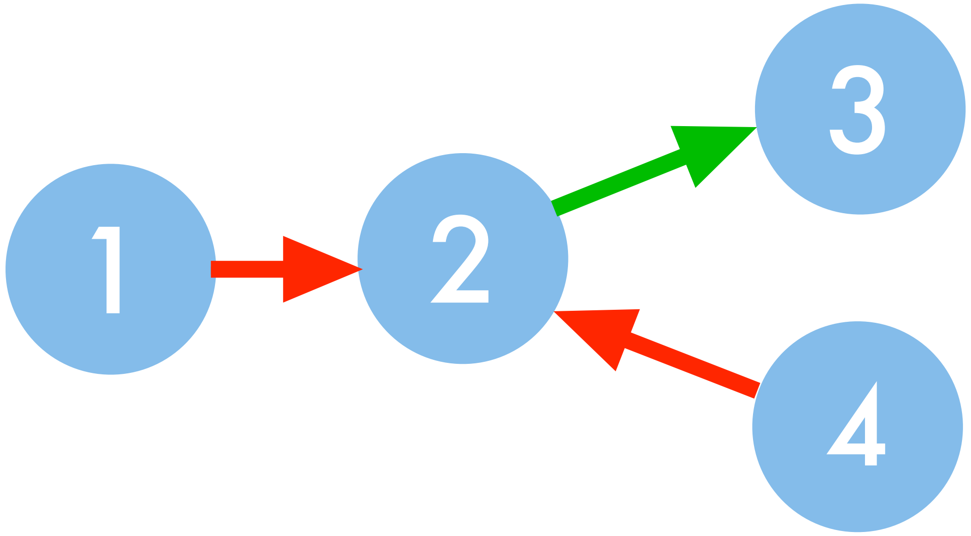
$$\bar{s} = \max_{x_{0\ldots n}} \underbrace{s(x_0)}_{:=l_0(x_0)} + \sum_{j=1}^{n} s(x_j|x_{j-1})$$



$$= \max_{x_{1\ldots n}} \underbrace{\max_{x_0} [l_0(x_0)s(x_1|x_0)]}_{:=l_1(x_1)} + \sum_{j=2}^{n} s(x_j|x_{j-1})$$

$$= \max_{x_{2\ldots n}} \underbrace{\max_{x_1} [l_1(x_1)s(x_2|x_1)]}_{:=l_2(x_2)} + \sum_{j=3}^{n} s(x_j|x_{j-1})$$

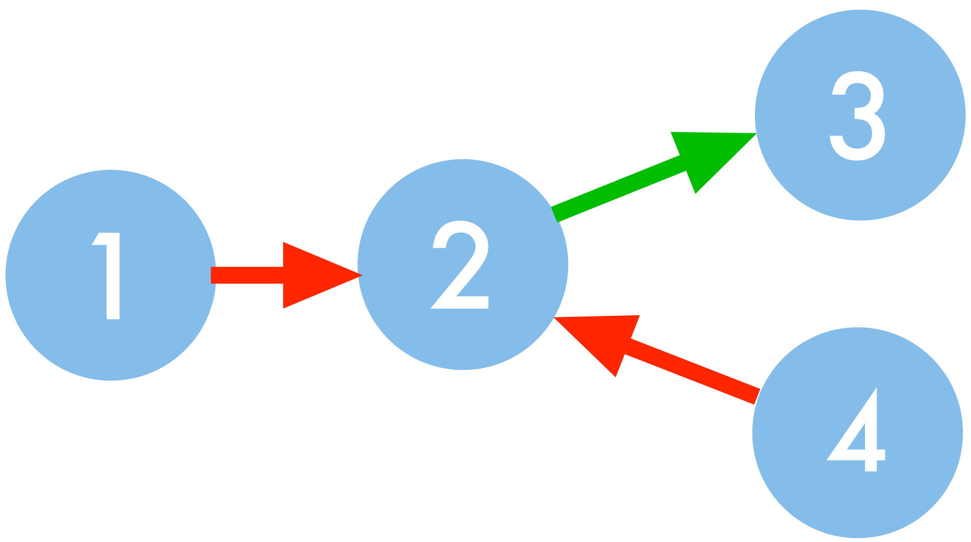
Junction Trees

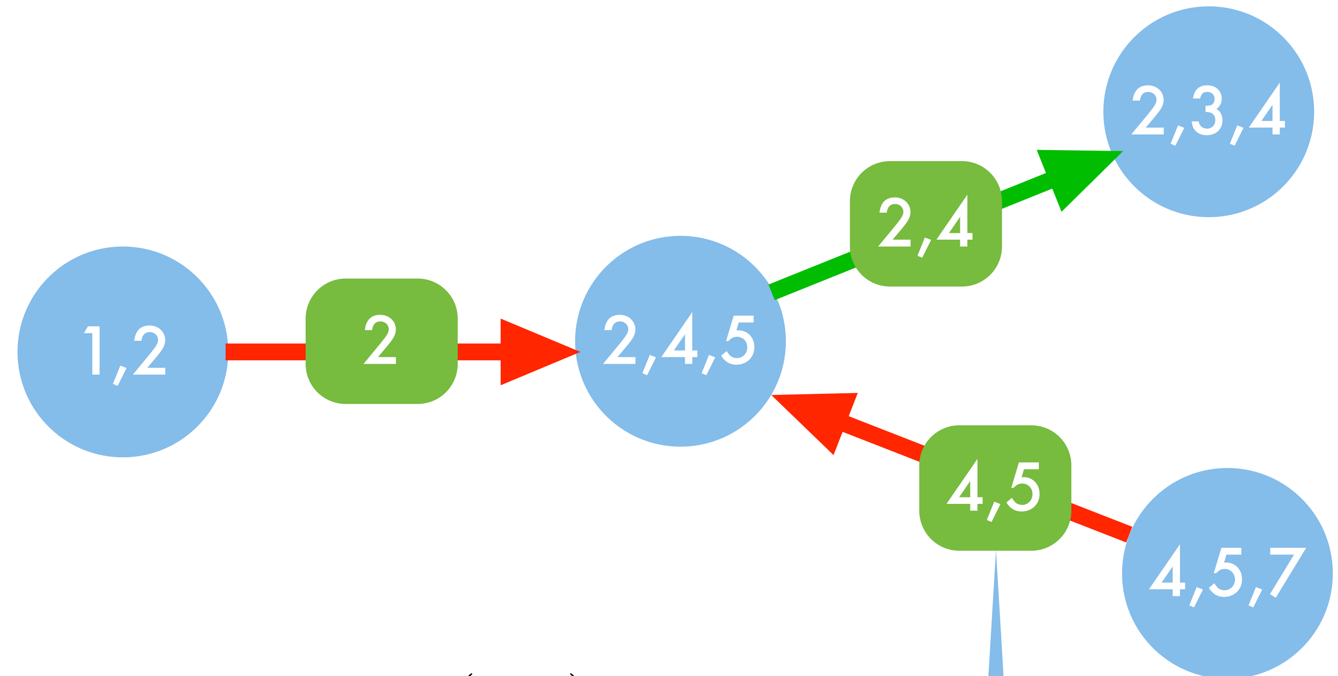


$$m_{i \to j}(x_j) = \max_{x_i} f(x_i, x_j) + \sum_{l \neq j} m_{l \to i}(x_j)$$

clique potential

Junction Trees





$$m_{i \to j}(x_j) = \max_{x_i} f(x_i, x_j) + \sum_{l \neq j} m_{l \to i}(x_j)$$

clique potential

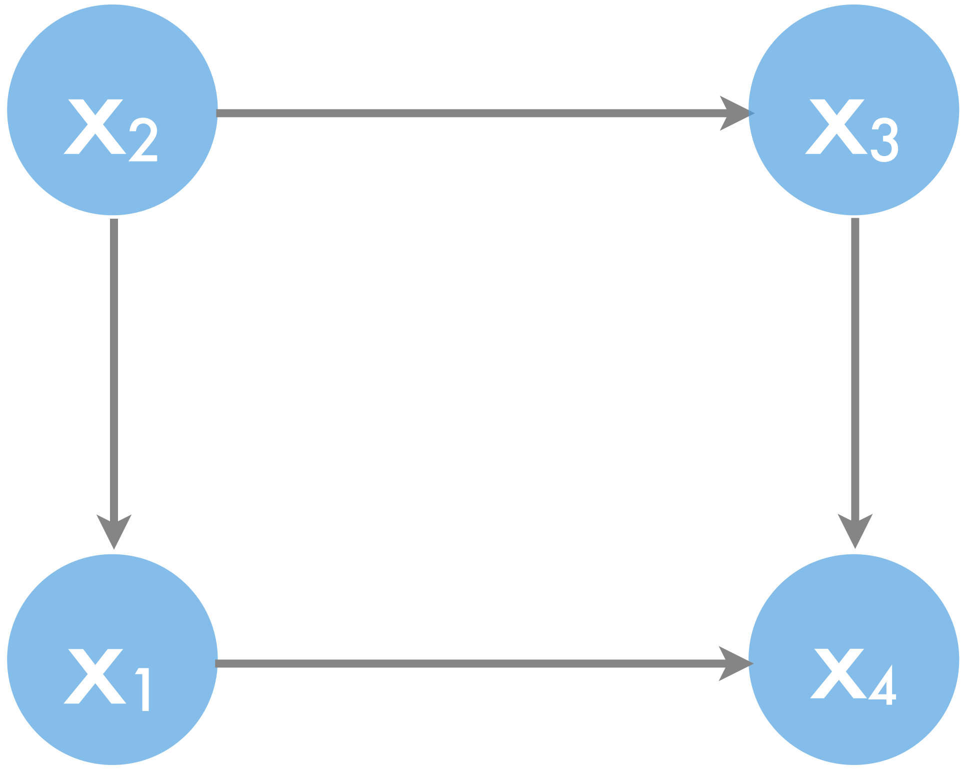
$$m_{245 \to 234}(x_{24})$$
$$= \max_{x_5} f(x_{245}) + m_{12 \to 245}(x_2) + m_{457 \to 245}(x_{45})$$

clique potential

separator set

No loops allowed

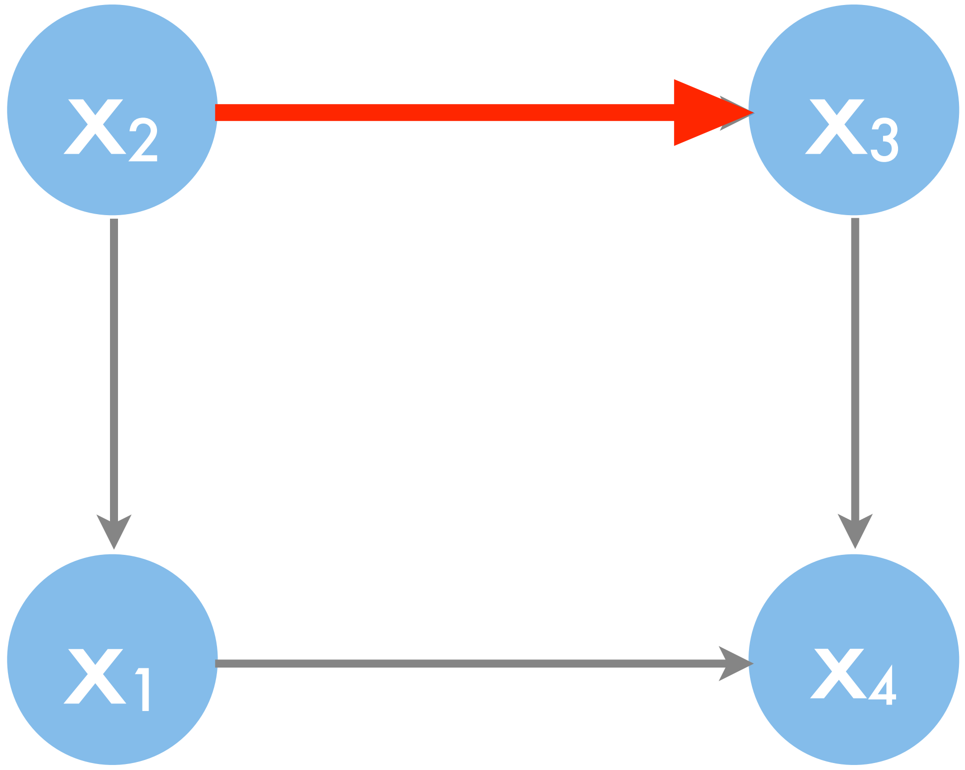
$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



Often use it anyway – Loopy Belief Propagation (Turbo Codes, Markov Random Fields, etc.)

No loops allowed

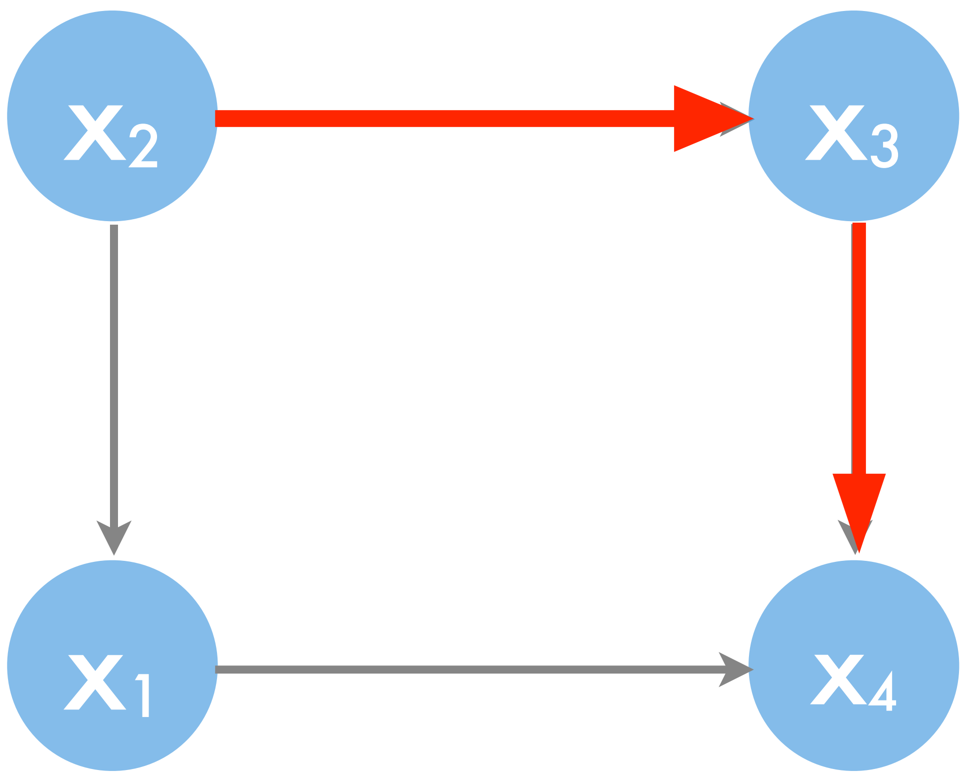
$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



Often use it anyway – Loopy Belief Propagation (Turbo Codes, Markov Random Fields, etc.)

No loops allowed

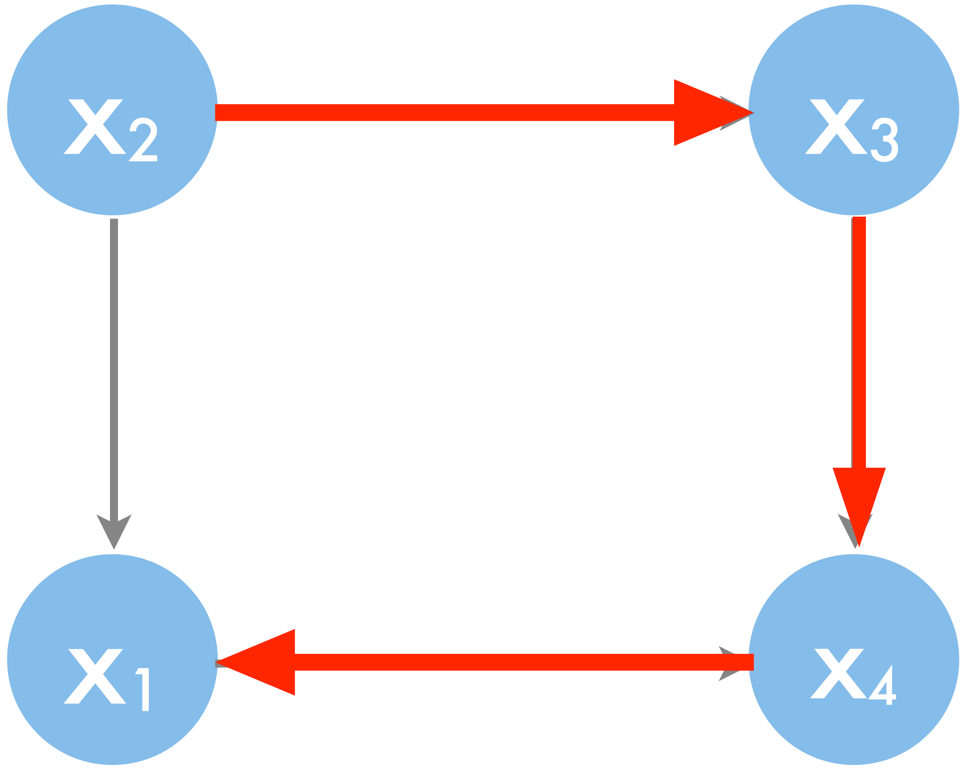
$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



Often use it anyway – Loopy Belief Propagation (Turbo Codes, Markov Random Fields, etc.)

No loops allowed

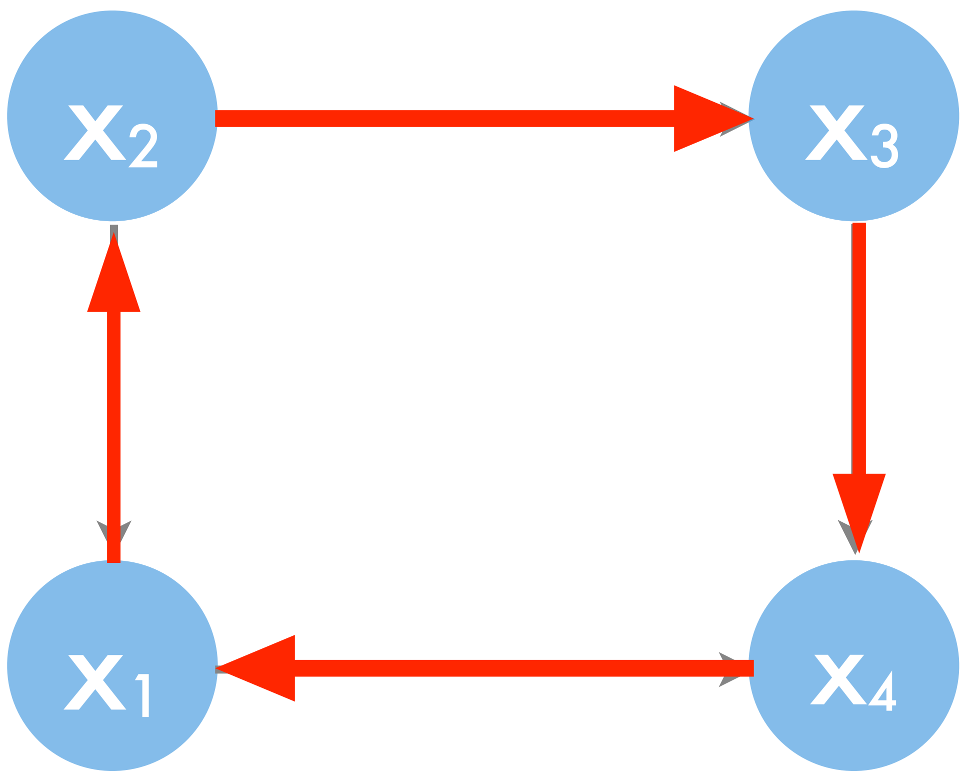
$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



Often use it anyway – Loopy Belief Propagation (Turbo Codes, Markov Random Fields, etc.)

No loops allowed

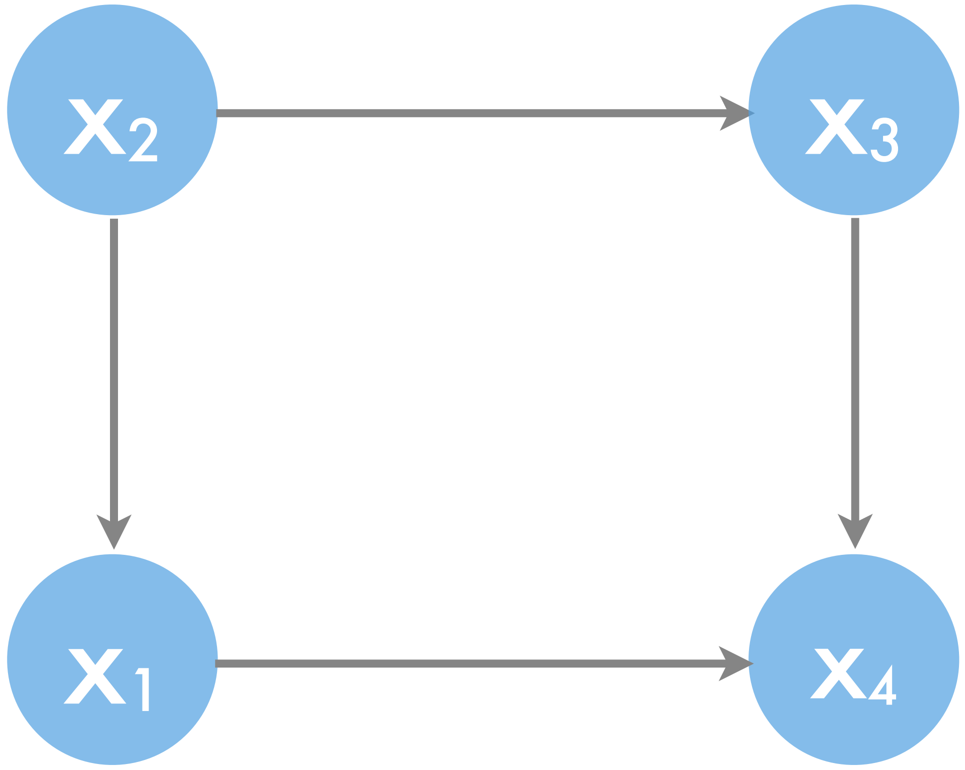
$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



Often use it anyway – Loopy Belief Propagation
(Turbo Codes, Markov Random Fields, etc.)

No loops allowed

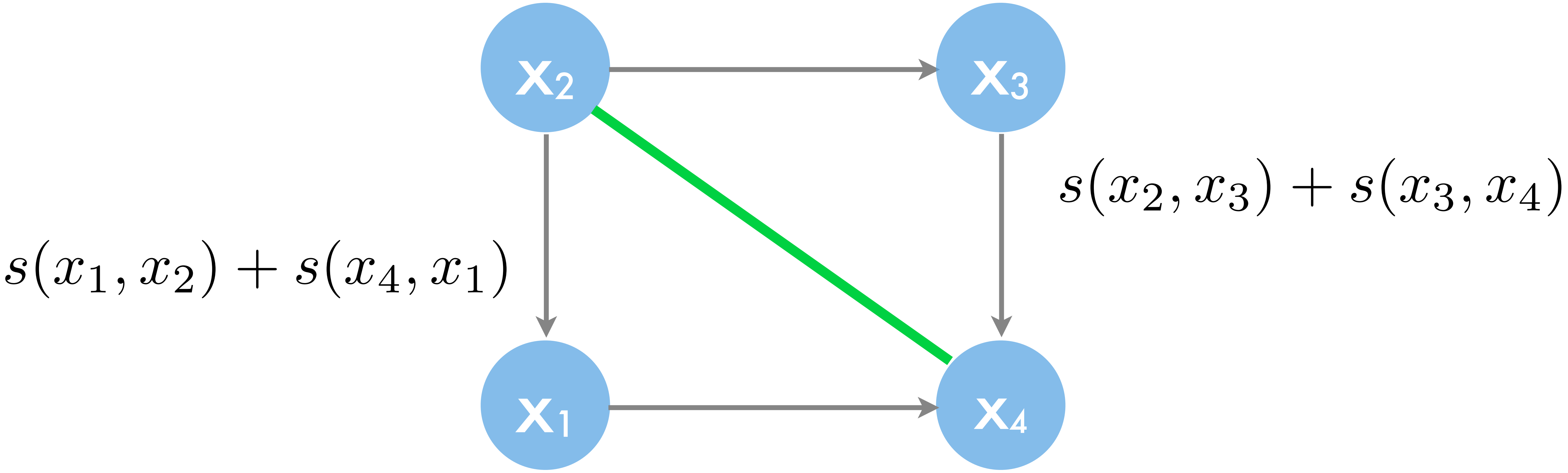
$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



Often use it anyway – Loopy Belief Propagation
(Turbo Codes, Markov Random Fields, etc.)

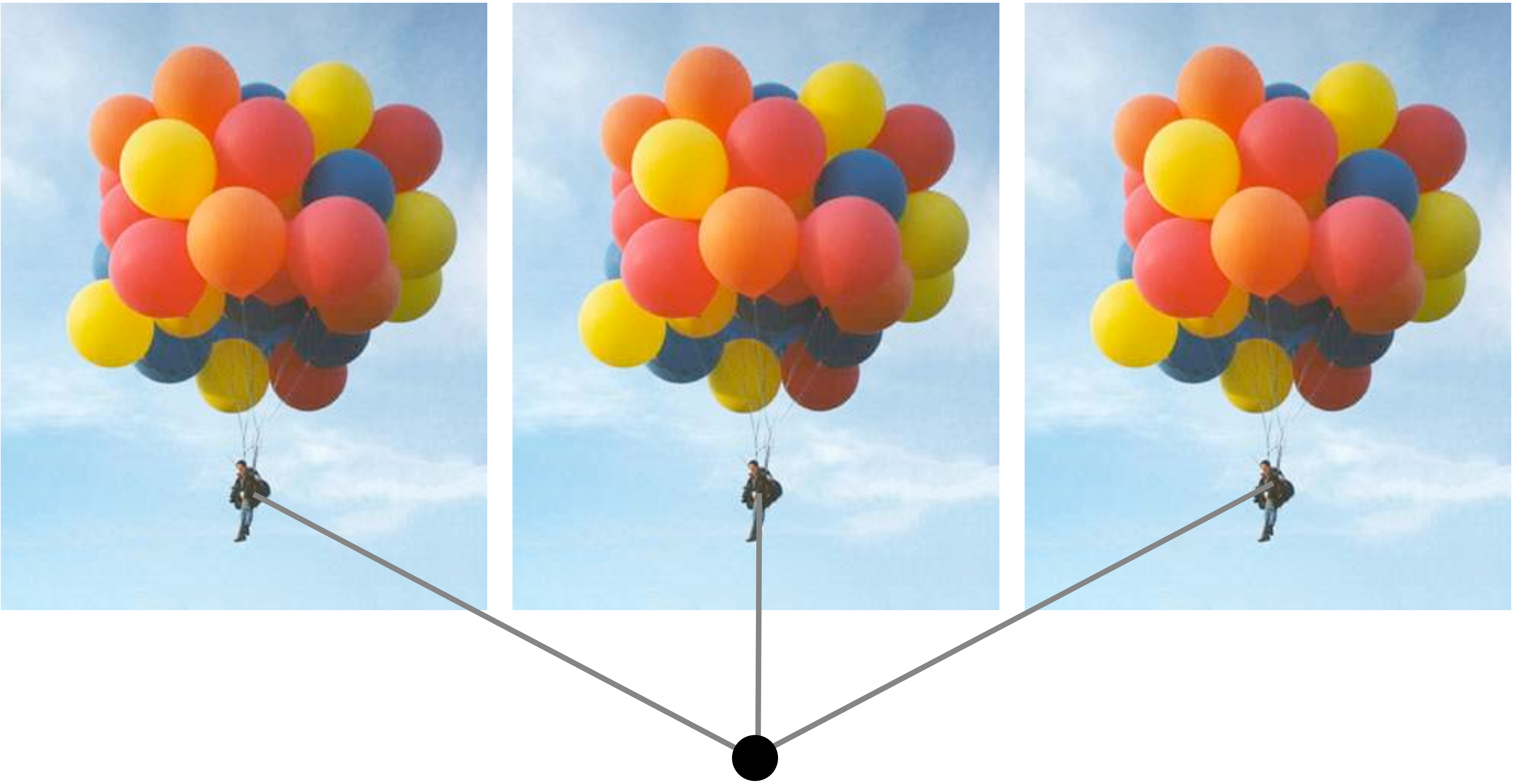
No loops allowed

$$s(x_1, x_2) + s(x_2, x_3) + s(x_3, x_4) + s(x_4, x_1)$$



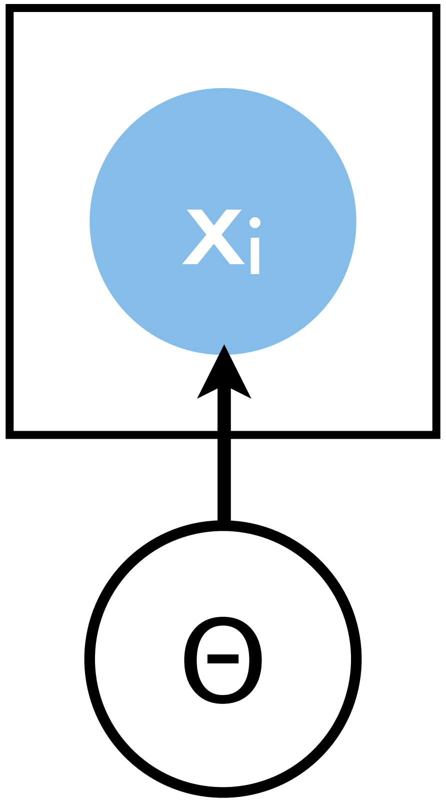
Often use it anyway – Loopy Belief Propagation (Turbo Codes, Markov Random Fields, etc.)

Clustering



Basic Idea

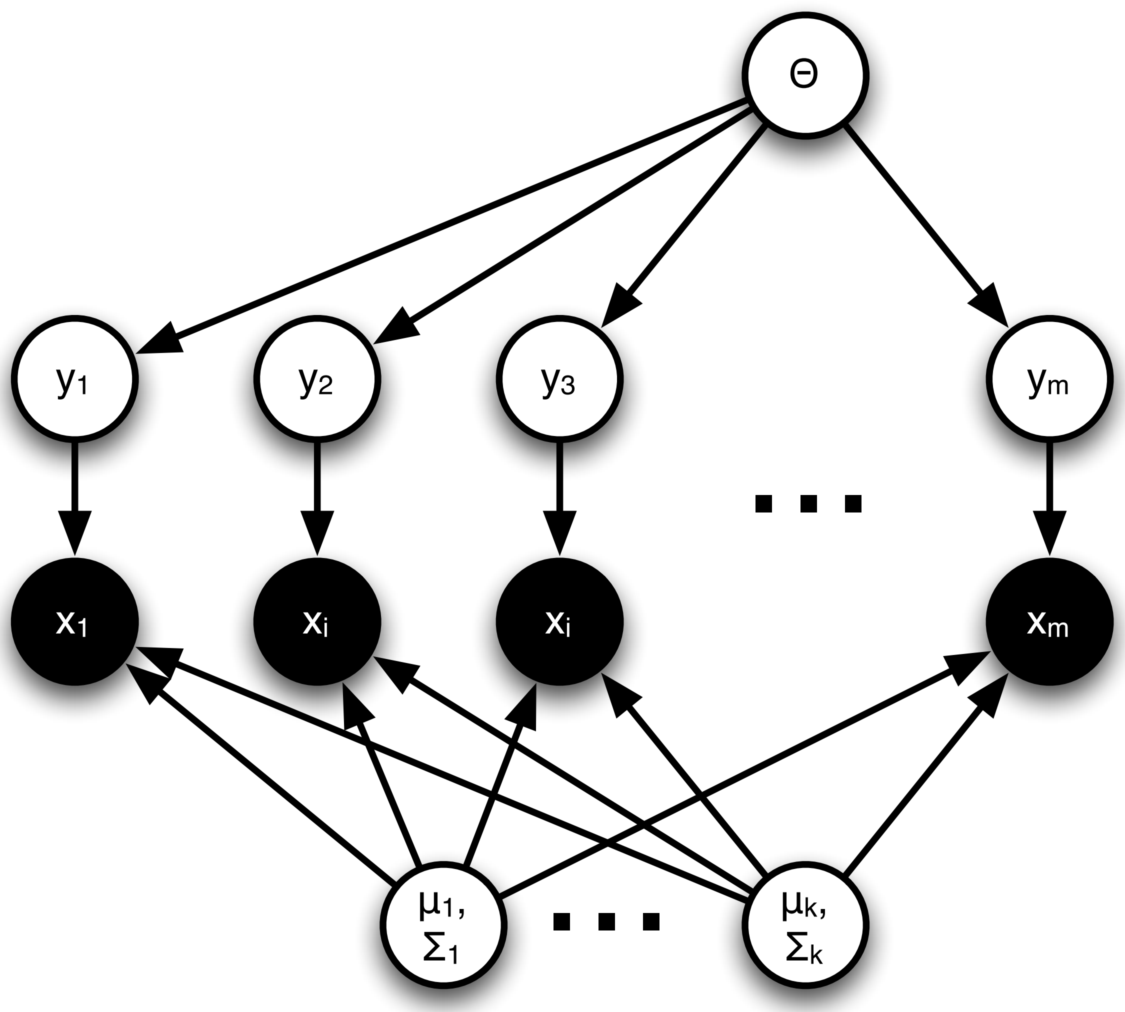
Density Estimation



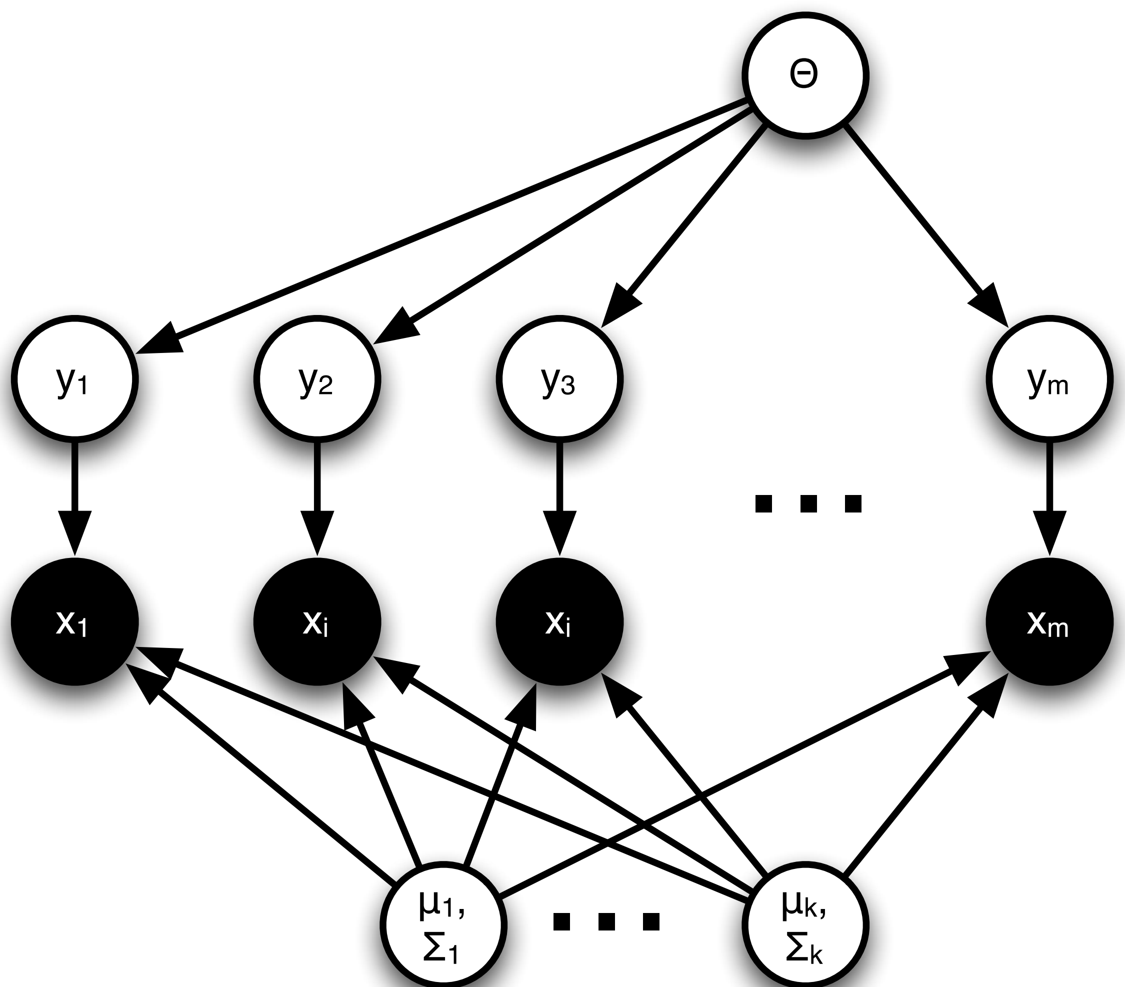
$$p(X|\theta) = \prod_{i=1}^{m} p(x_i|\theta)$$

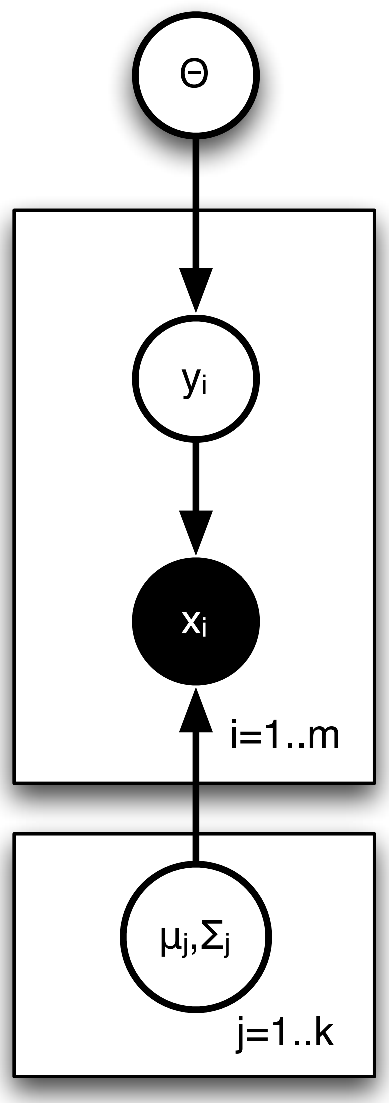
- Draw latent parameter Θ
- For all i draw observed x_i given Θ
- What if the basic model doesn't fit all data?

One size doesn't fit all

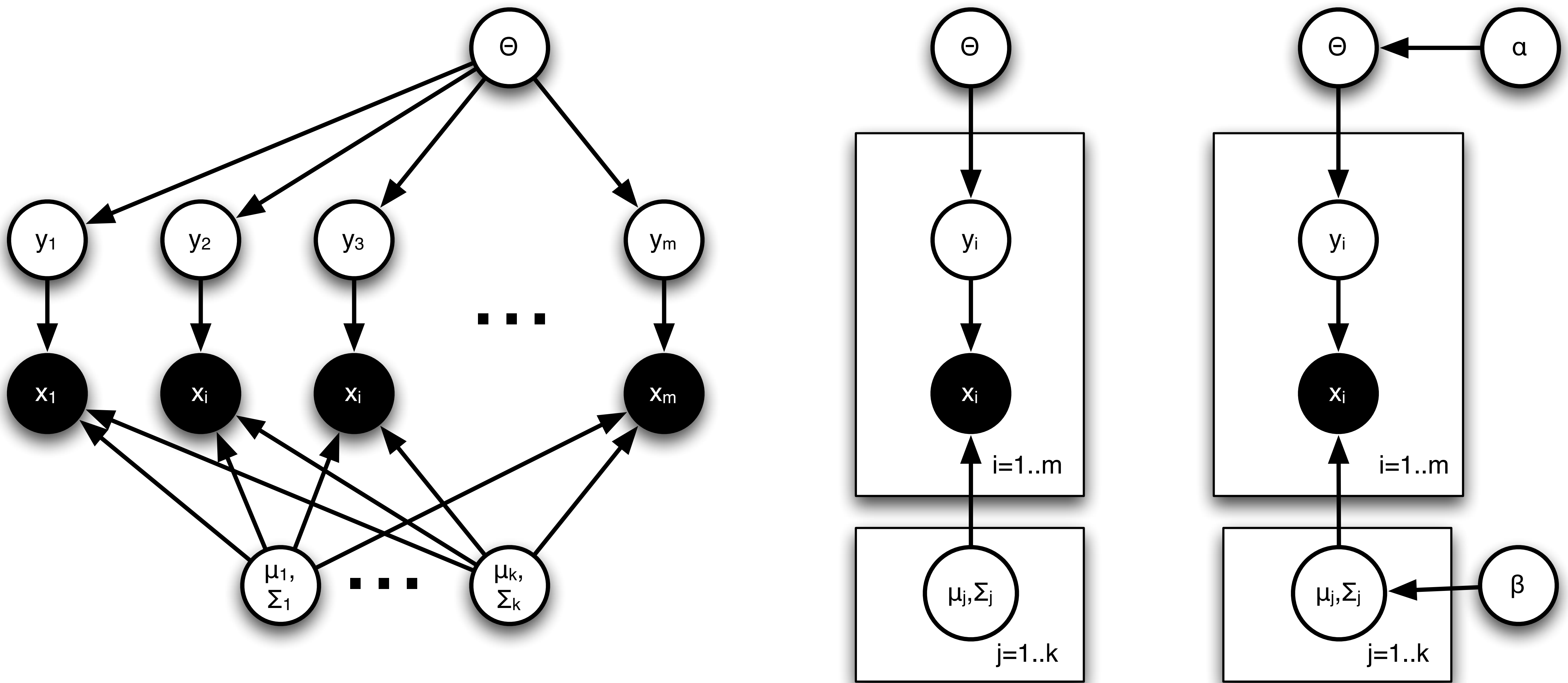


One size doesn’t fit all

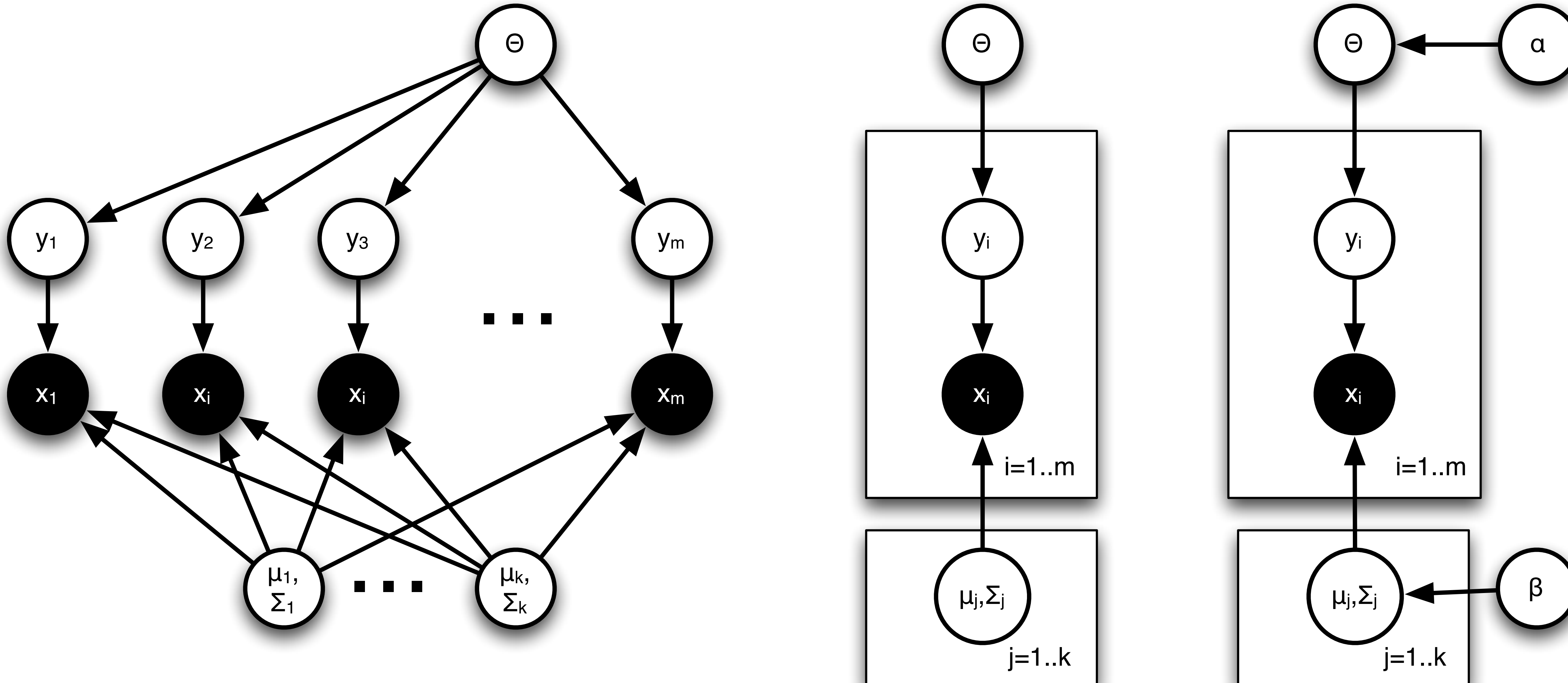




One size doesn't fit all



One size doesn't fit all



$$p(X, Y|\theta, \sigma, \mu) = \prod_{i=1}^{n} p(x_i|y_i, \sigma, \mu)p(y_i|\theta)$$

What can we cluster?

What can we cluster?

products

urls

mails

text

news

users

queries

locations

spammers

ads

events

abuse

Mixture of Gaussians

- Draw cluster label y from discrete distribution
- Draw data x from Gaussian for given cluster y

- Prior for discrete distribution - Dirichlet
- Prior for Gaussians - Gauss-Wishart distribution

- Problem: we don’t know y
  - If we knew the parameters we could get y
  - If we knew y we could get the parameters

k-means

- Fixed uniform variance for all Gaussians
- Fixed uniform distribution over clusters
- Initialize centers with random subset of points
- Find most likely cluster y for x

$$y_i = \operatorname*{argmax}_y p(x_i|y, \sigma, \mu)$$

- Find most likely center for given cluster

$$\mu_y = \frac{1}{n_y} \sum_i \{y_i = y\} x_i$$

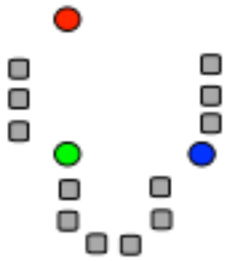
- Repeat until converged

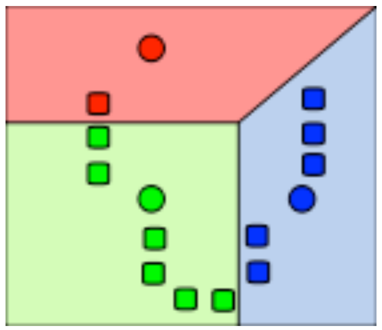
k-means

- Pro
  - simple algorithm
  - can be implemented by MapReduce passes
- Con
  - no proper probabilistic representation
  - can get stuck easily in local minima

k-means

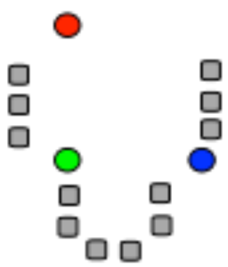
partitioning





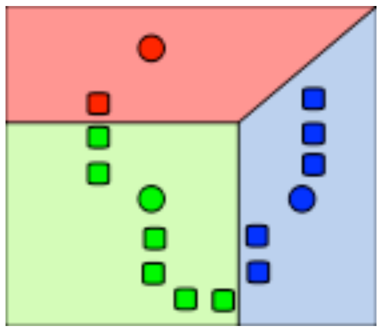
initialization

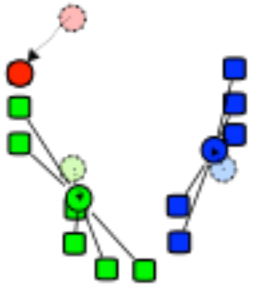
k-means



initialization

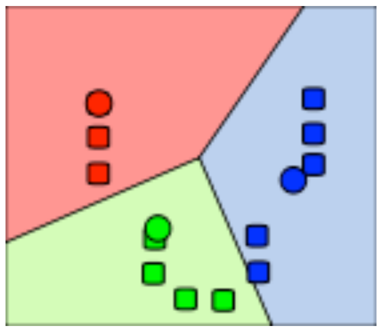
partitioning





update

partitioning



Variational Inference

Expectation Maximization

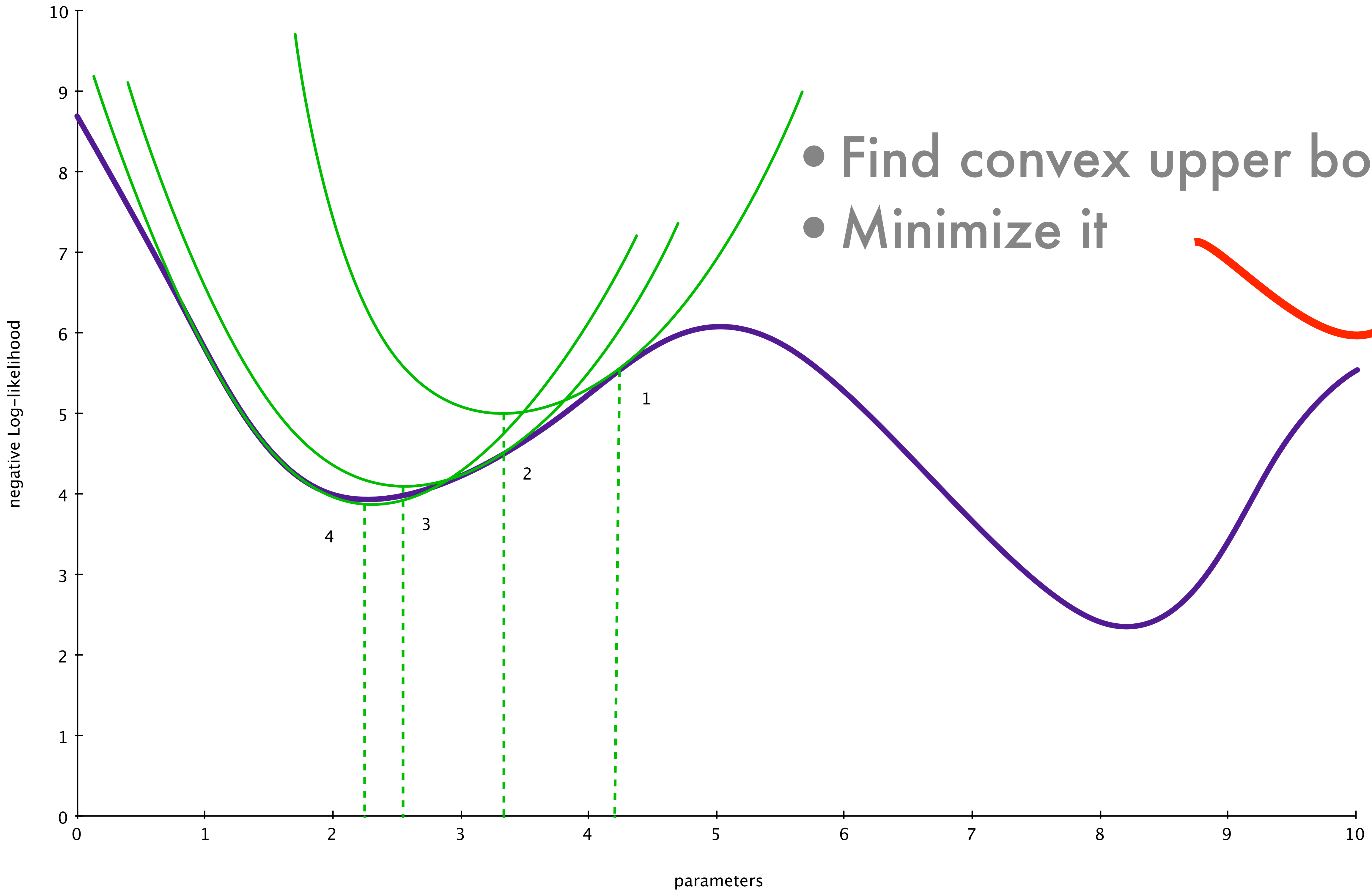
- Optimization Problem

$$\underset{\theta,\mu,\sigma}{\text{maximize}}\, p(X|\theta,\sigma,\mu) = \underset{\theta,\mu,\sigma}{\text{maximize}} \sum_{Y} \prod_{i=1}^{n} p(x_i|y_i,\sigma,\mu)p(y_i|\theta)$$

  This problem is nonconvex and difficult to solve

- Key idea
  If we knew p(y|x) we could estimate the remaining parameters easily and vice versa

Nonconvex Minimization



- Find convex upper bound
- Minimize it

Expectation Maximization

Expectation Maximization

- Variational Bound

$$
\begin{aligned}
\log p(x;\theta) &\geq \log p(x;\theta) - D(q(y)\|p(y|x;\theta)) \\
&= \int dq(y)\left[\log p(x;\theta) + \log p(y|x;\theta) - \log q(y)\right] \\
&= \int dq(y) \log p(x,y;\theta) - \int dq(y) \log q(y)
\end{aligned}
$$

This inequality is tight for p(y|x) = q(y)

Expectation Maximization

- Variational Bound

$$\begin{aligned}\log p(x;\theta) &\geq \log p(x;\theta) - D(q(y)\|p(y|x;\theta)) \\ &= \int dq(y)\left[\log p(x;\theta) + \log p(y|x;\theta) - \log q(y)\right] \\ &= \int dq(y) \log p(x,y;\theta) - \int dq(y) \log q(y)\end{aligned}$$

  This inequality is tight for p(y|x) = q(y)

- Expectation step

$$q(y) = p(y|x;\theta)$$

Expectation Maximization

- Variational Bound

$$\begin{aligned}\log p(x;\theta) &\geq \log p(x;\theta) - D(q(y)\|p(y|x;\theta)) \\ &= \int dq(y)\left[\log p(x;\theta) + \log p(y|x;\theta) - \log q(y)\right] \\ &= \int dq(y)\log p(x,y;\theta) - \int dq(y)\log q(y)\end{aligned}$$

  This inequality is tight for p(y|x) = q(y)

- Expectation step

$$q(y) = p(y|x;\theta)$$

find bound

Expectation Maximization

- Variational Bound

$$\begin{aligned}\log p(x;\theta) &\geq \log p(x;\theta) - D(q(y)\|p(y|x;\theta)) \\ &= \int dq(y)\left[\log p(x;\theta) + \log p(y|x;\theta) - \log q(y)\right] \\ &= \int dq(y) \log p(x,y;\theta) - \int dq(y) \log q(y)\end{aligned}$$

  This inequality is tight for p(y|x) = q(y)

- Expectation step

$$q(y) = p(y|x;\theta)$$

find bound

- Maximization step

$$\theta^* = \operatorname*{argmax}_\theta \int dq(y) \log p(x,y;\theta)$$

Expectation Maximization

- Variational Bound

$$\begin{aligned}\log p(x;\theta) &\geq \log p(x;\theta) - D(q(y)\|p(y|x;\theta)) \\ &= \int dq(y)\left[\log p(x;\theta) + \log p(y|x;\theta) - \log q(y)\right] \\ &= \int dq(y) \log p(x,y;\theta) - \int dq(y) \log q(y)\end{aligned}$$

This inequality is tight for p(y|x) = q(y)

- Expectation step

$$q(y) = p(y|x;\theta)$$

find bound

- Maximization step

$$\theta^* = \operatorname*{argmax}_\theta \int dq(y) \log p(x,y;\theta)$$

maximize it

Expectation Step

- Factorizing distribution

$$q(Y) = \prod_i q_i(y)$$

- E-Step

$$q_i(y) \propto p(x_i|y_i, \mu, \sigma)p(y_i|\theta) \text{ hence}$$

$$m_{iy} := \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma_y|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(x_i - \mu_y)\Sigma_y^{-1}(x_i - \mu_y)\right] p(y)$$

$$\boxed{q_i(y) = \frac{m_{iy}}{\sum_{y'} m_{iy'}}}$$

Maximization Step

- Log-likelihood

$$\log p(X, Y|\theta, \mu, \sigma) = \sum_{i=1}^{n} \log p(x_i|y_i, \mu, \sigma) + \log p(y_i|\theta)$$

- Cluster distribution (weighted Gaussian MLE)

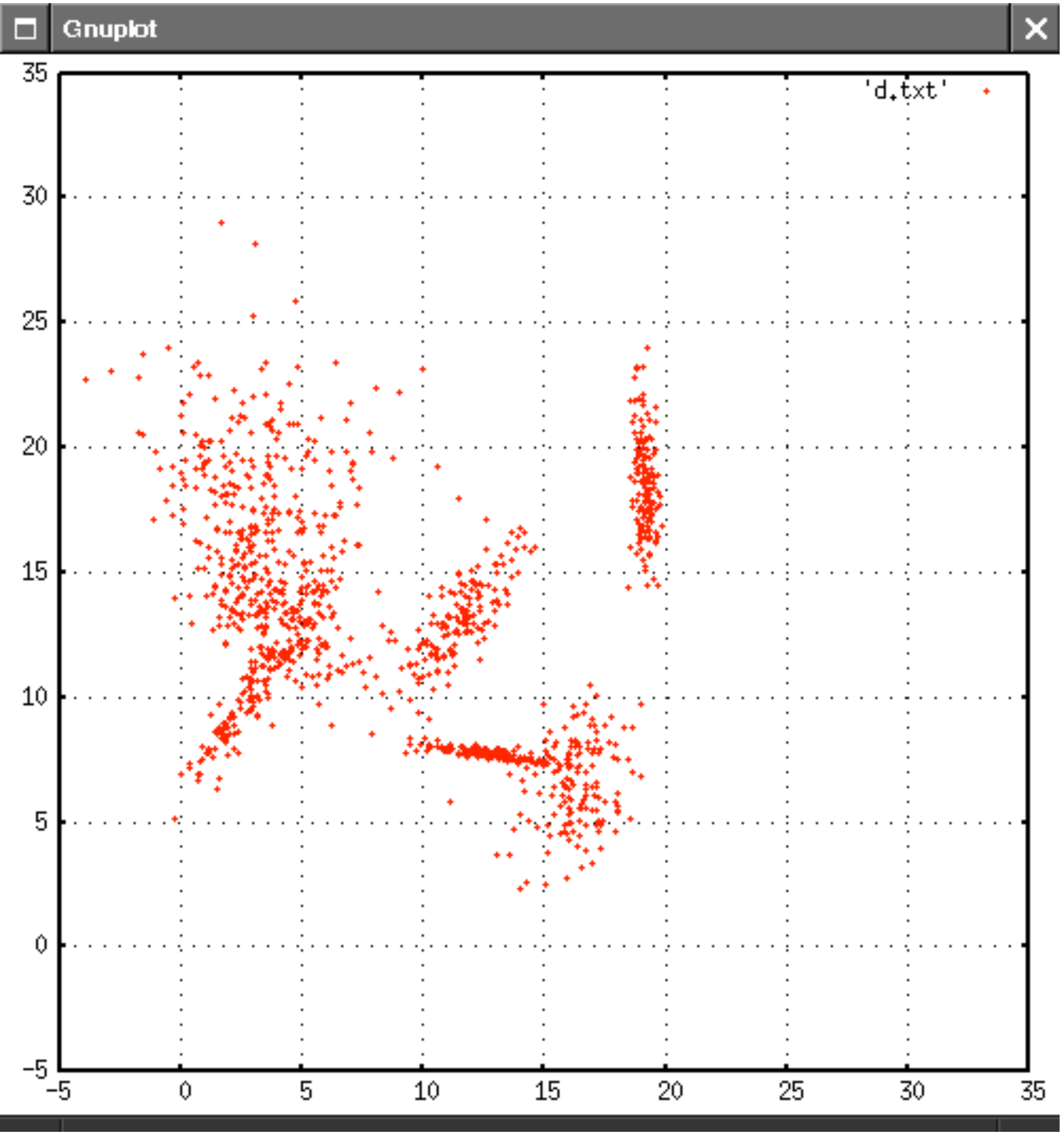
$$n_y = \sum_i q_i(y)$$

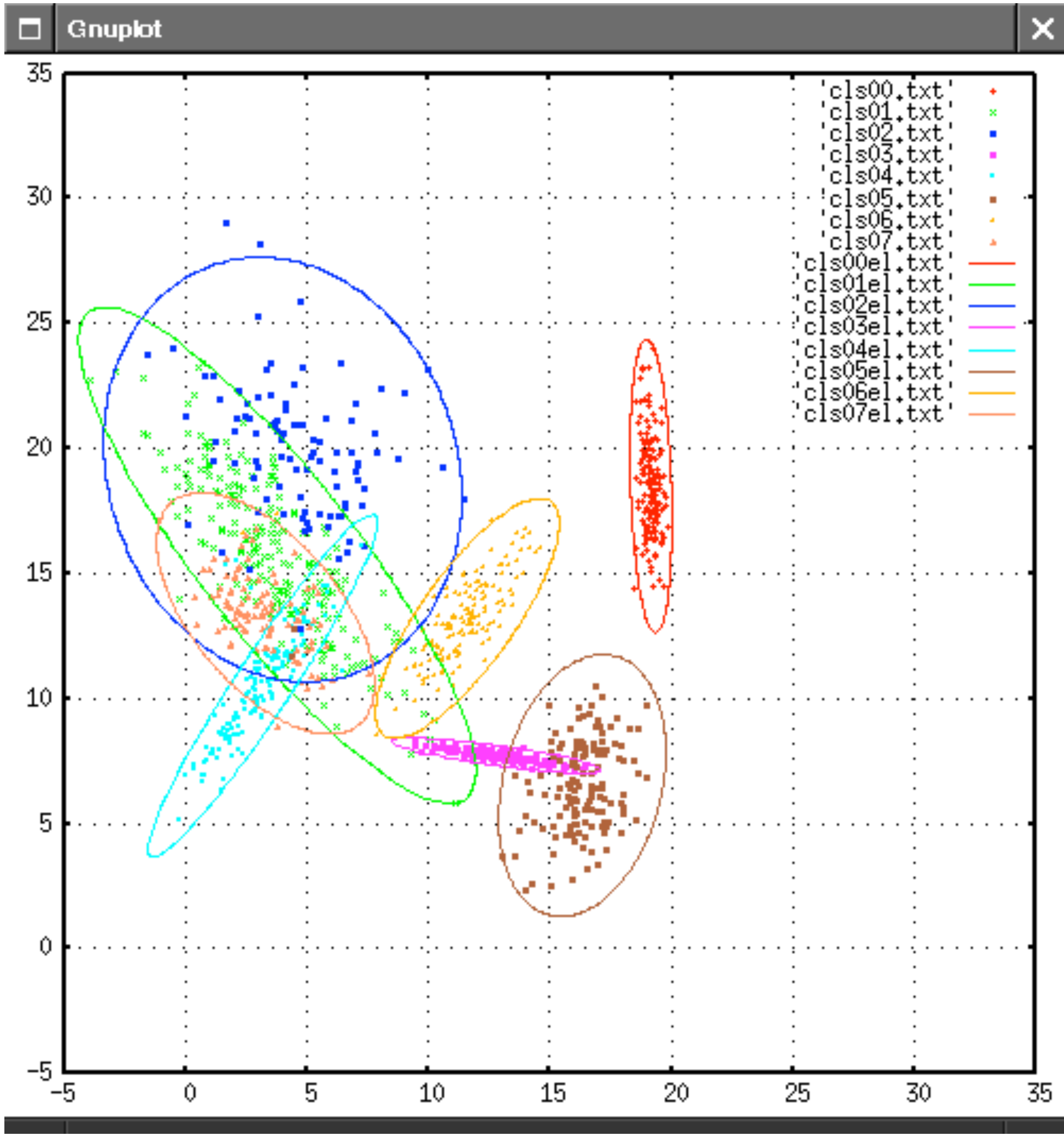
$$\boxed{\begin{aligned} \mu_y &= \frac{1}{n_y} \sum_{i=1}^{n} q_i(y) x_i \\ \Sigma_y &= \frac{1}{n_y} \sum_{i=1}^{n} q_i(y) x_i x_i^\top - \mu_y \mu_y^\top \end{aligned}}$$

- Cluster probabilities

$$\theta^* = \operatorname*{argmax}_\theta \sum_{i=1}^{n} \sum_y q_i(y) \log p(y_i|\theta) \text{ hence } \boxed{p(y|\theta) = \frac{n_y}{n}}$$

EM Clustering in action





Problem

Estimates will diverge
(infinite variance, zero probability, tiny clusters)

Solution

- Use priors for μ, σ, θ
  - Dirichlet distribution for cluster probabilities
  - Gauss-Wishart for Gaussian
- Cluster distribution

$$n_y = n_0 + \sum_i q_i(y)$$

$$\mu_y = \frac{1}{n_y} \sum_{i=1}^{n} q_i(y) x_i$$

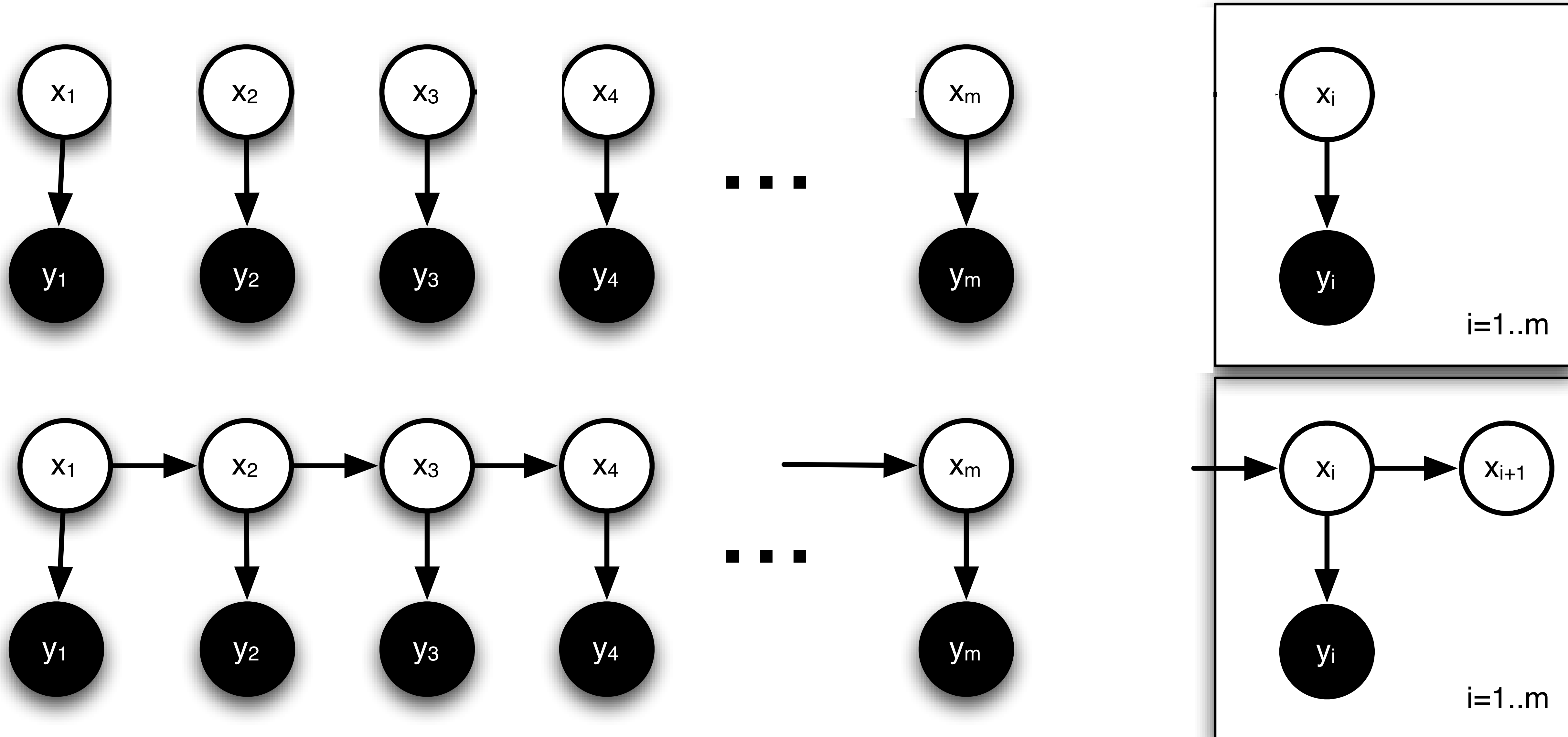
$$\Sigma_y = \frac{1}{n_y} \sum_{i=1}^{n} q_i(y) x_i x_i^\top + \frac{n_0}{n_y} \mathbf{1} - \mu_y \mu_y^\top$$

- Cluster probabilities

$$p(y|\theta) = \frac{n_y}{n + k \cdot n_0}$$

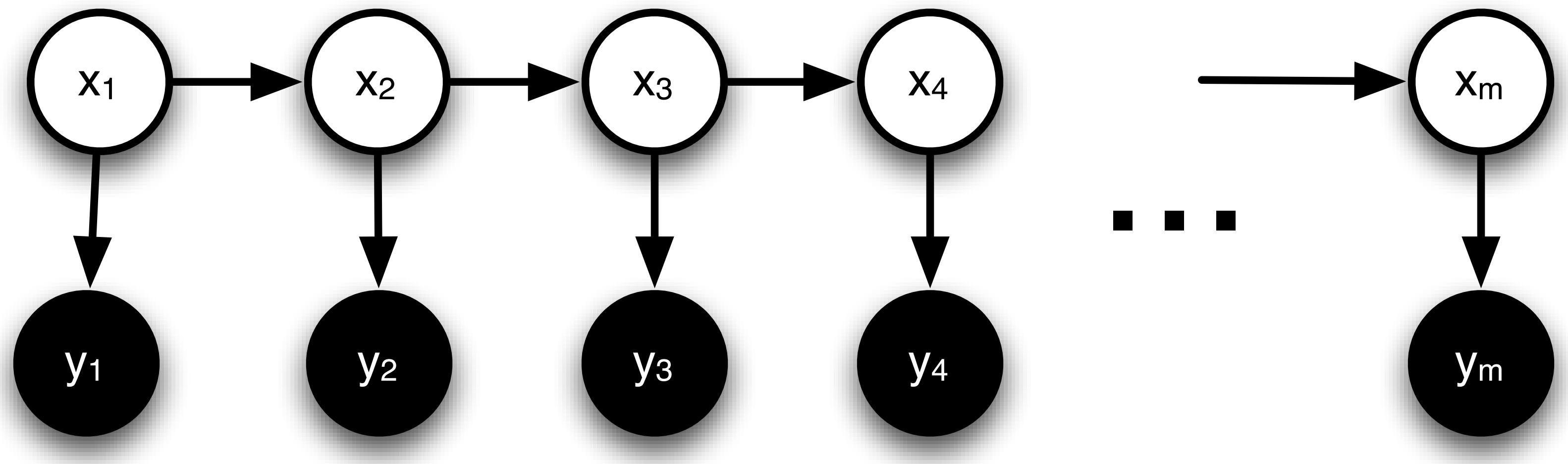
Hidden Markov Models

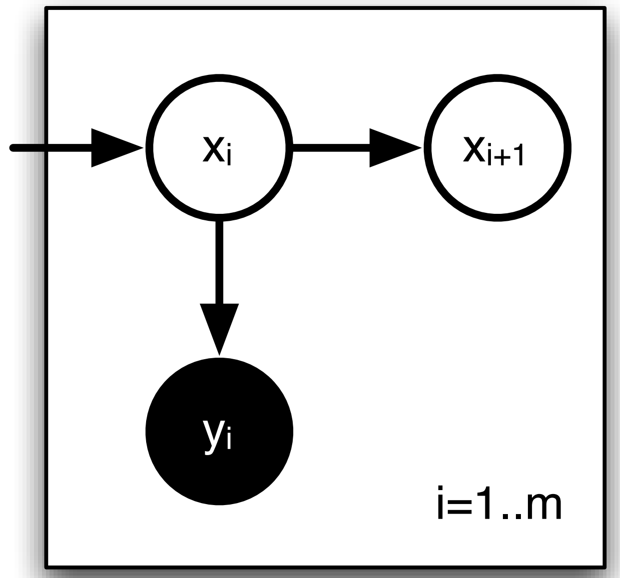
Clustering and Hidden Markov Models



- Clustering - no dependence between observations
- Hidden Markov Model - dependence between states

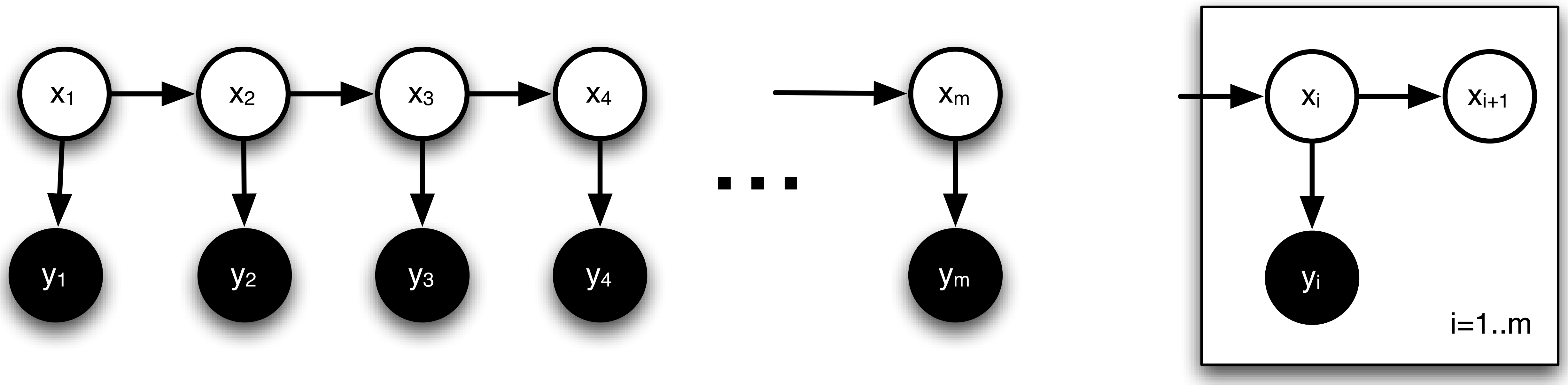
Applications





- Speech recognition (sound|text)
- Optical character recognition (writing|text)
- Gene finding (DNA sequence|genes)
- Activity recognition (accelerometer|activity)

Inference



$$p(x,y) = p(y_1)\left[\prod_{i=1}^{m-1} p(y_{i+1}|y_i)p(x_i|y_i)\right] p(x_m|y_m)$$

- Summing over y possible via dynamic programming
- Log-likelihood is nonconvex

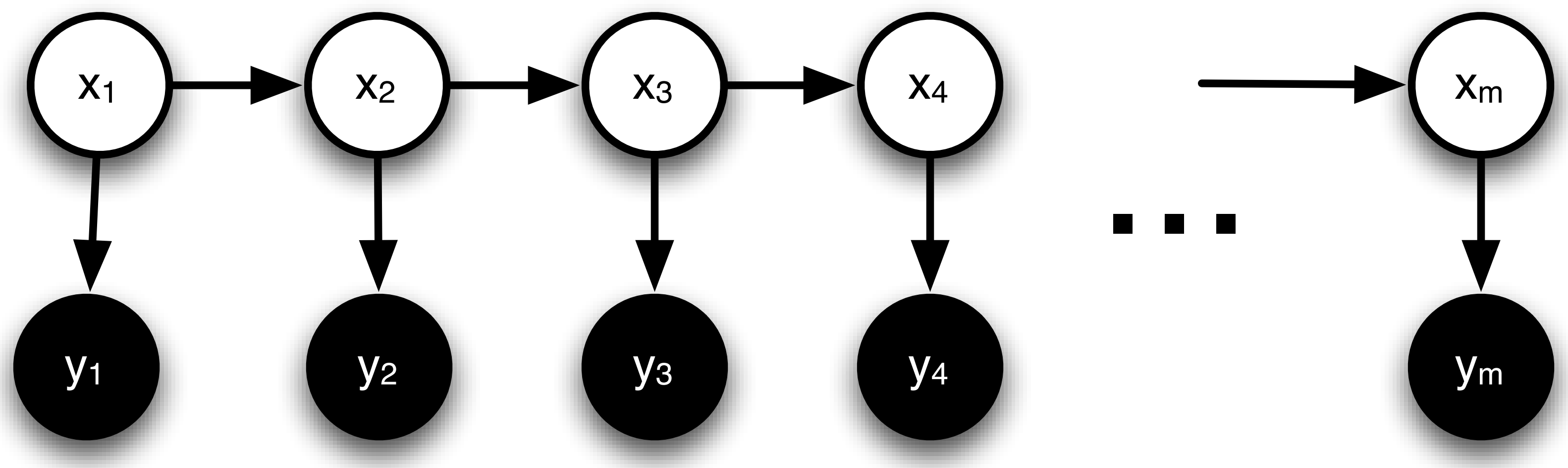
Variational Approximation

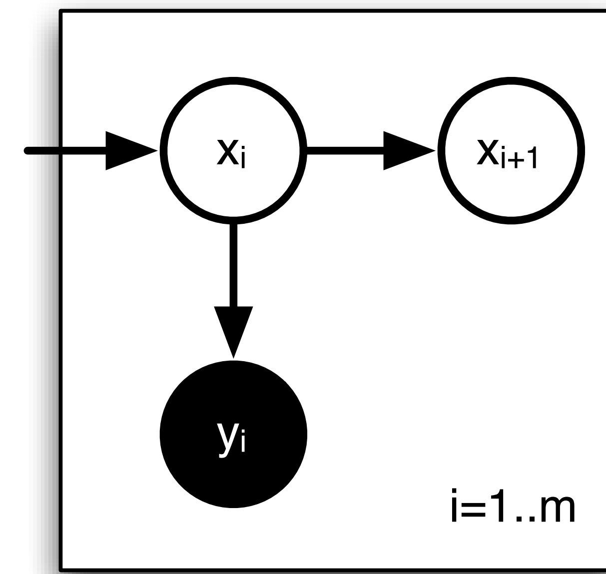
- Lower bound on log-likelihood

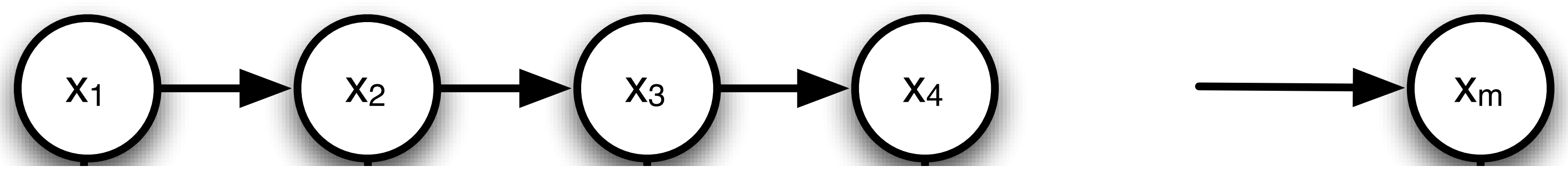
$$\log p(x;\theta) \geq \int dq(y) \log p(x,y;\theta) - \int dq(y) \log q(y)$$

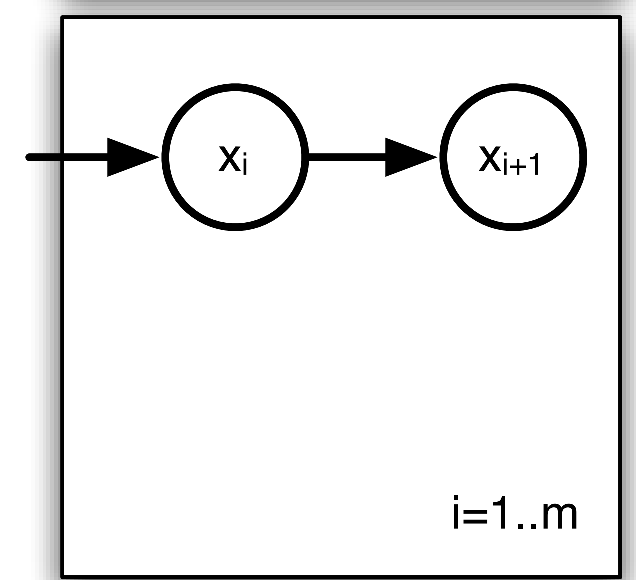
- Key insight - inequality holds for any q
  - Find q within subset Q to tighten inequality
  - Find parameters to maximize for fixed q
- Inference for graphical models where joint probability computation is infeasible

Variational Approximation









- Variational approximation via

$$q(y) = q(y_1) \prod_{x=2}^{m} q(y_i | y_{i-1})$$

- Compute p(x|y) via dynamic programming

Variational Method

- Initialize parameters somehow
- Set $q(x) = p(x|y)$
  Dynamic programming yields chain
- Maximizing the log-likelihod w.r.t. q

$$\log p(x;\theta) \geq \int dq(y) \log p(x,y;\theta) - \int dq(y) \log q(y)$$

$$p(x,y) = p(y_1) \left[\prod_{i=1}^{m-1} p(y_{i+1}|y_i) p(x_i|y_i) \right] p(x_m|y_m)$$

q(y_1)

q(y_{i+1} | y_i)

q(y_i)

Parameter Estimation

$$\mathbf{E}_{y\sim q}\left[\log p(x,y;\theta)\right] = \mathbf{E}_{y_1\sim q}\log p(y_1;\theta) + \sum_{i=1}^{\dots}\mathbf{E}_{y_i\sim q}\log p(x_i|y_i;\theta)$$

$$+\sum_{i=1}^{m-1}\mathbf{E}_{y_{i+1},y_i\sim q}\log p(y_{i+1}|y_i;\theta)$$

- $p(y_1)$
  Since we have $\mathbf{E}_{q(y_1)}[\log p(y_1)]$ **set $p(y_1) = q(y_1)$**
- $p(x_i|y_i)$
  Same as clustering
  e.g. for Gaussians

$$\mu_y = \frac{1}{n_y}\sum_{i=1}^{n} q_i(y)x_i$$

$$\Sigma_y = \frac{1}{n_y}\sum_{i=1}^{n} q_i(y)x_i x_i^\top - \mu_y\mu_y^\top$$

Parameter Estimation

$$\mathbf{E}_{y \sim q}\left[\log p(x, y; \theta)\right] = \mathbf{E}_{y_1 \sim q} \log p(y_1; \theta) + \sum_{i=1}^{m} \mathbf{E}_{y_i \sim q} \log p(x_i | y_i; \theta)$$

$$+ \sum_{i=1}^{m-1} \mathbf{E}_{y_{i+1}, y_i \sim q} \log p(y_{i+1} | y_i; \theta)$$

- **Maximum likelihood estimate for p(y'|y)**

$$\sum_{i=1}^{m-1} q(y_{i+1} = a, y_i = b) \log p(a|b)$$

$$\text{hence } p(a|b) = \frac{\sum_{i=1}^{m-1} q(y_{i+1} = a, y_i = b)}{\sum_{i=1}^{m-1} q(y_i = b)}$$

effective sample

Smoothed Estimates

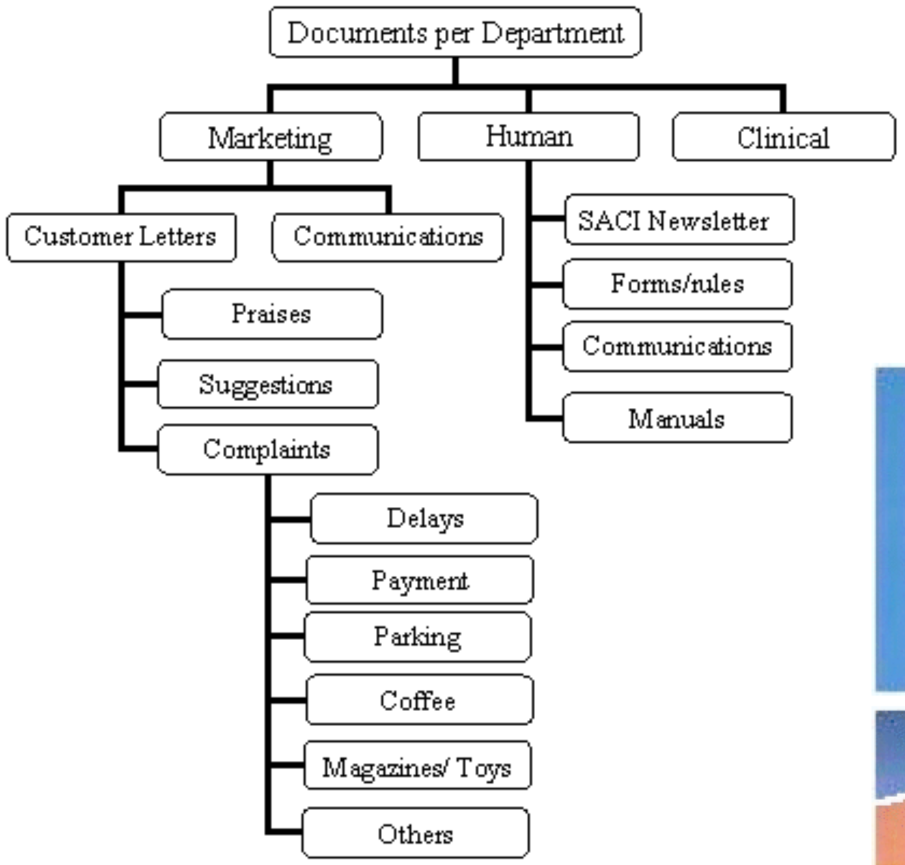
- Laplace prior on latent state distribution
  - Uniform distribution over states
  - Alternatively assume that state remains

$$p(a|b) = \frac{n_{a|b} + \sum_{i=1}^{m-1} q(y_{i+1} = a, y_i = b)}{n_b + \sum_{i=1}^{m-1} q(y_i = b)}$$

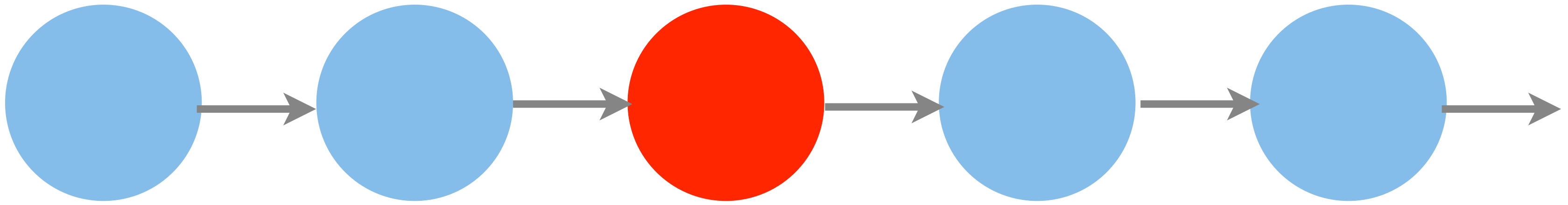
transition smoother

aggregate mass

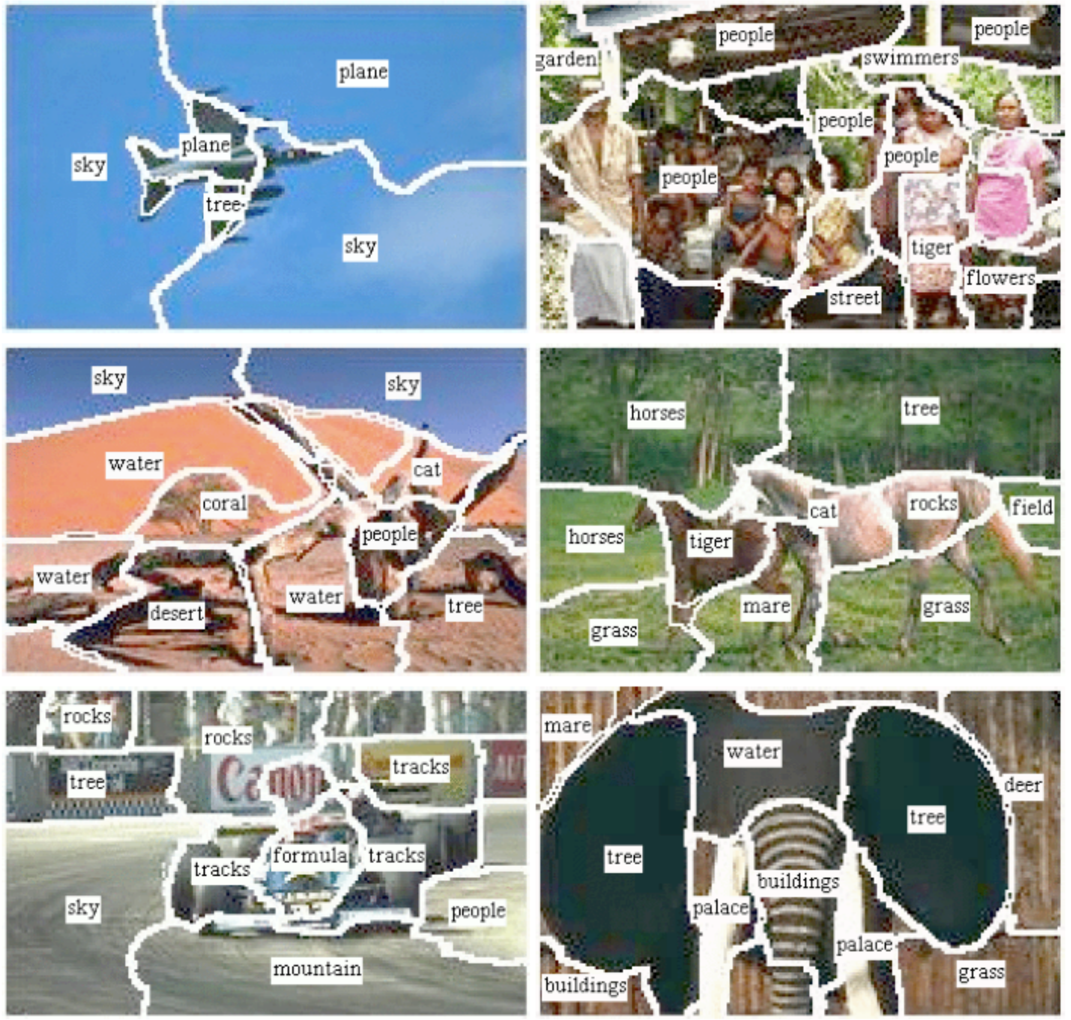
Beyond mixtures



taxonomies



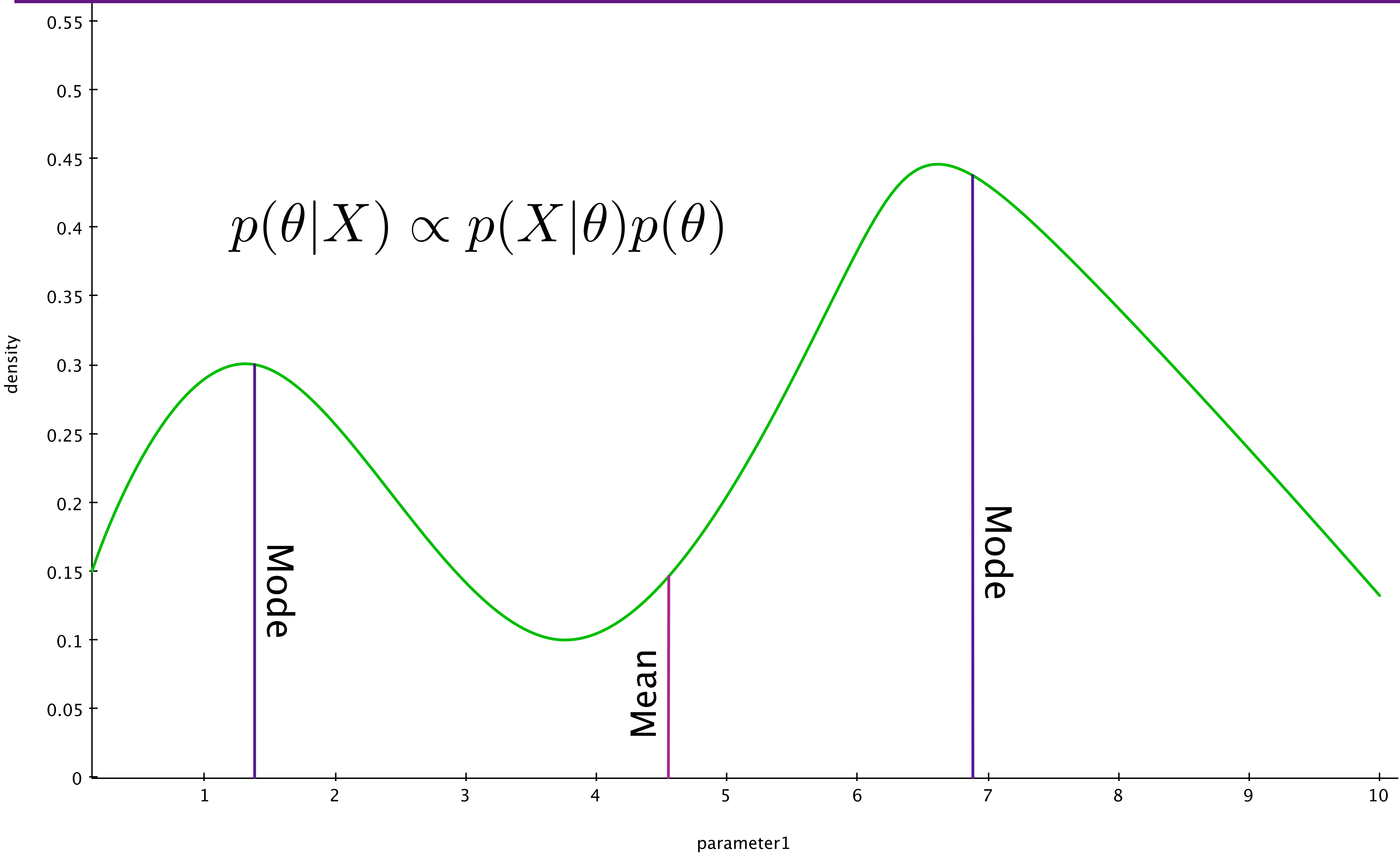
chains



topics

Sampling

Is maximization (always) good?



Sampling

- Key idea
  - Want accurate distribution of the posterior
  - Sample from posterior distribution rather than maximizing it
- Problem - direct sampling is usually intractable
- Solutions
  - Markov Chain Monte Carlo (complicated)
  - Gibbs Sampling (somewhat simpler)

$$x \sim p(x|x') \text{ and then } x' \sim p(x'|x)$$

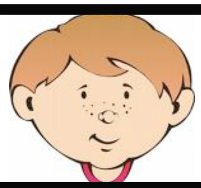
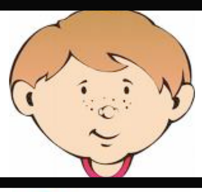
Gibbs sampling

- Gibbs sampling:
  - In most cases direct sampling not possible
  - Draw one set of variables at a time

		
	0.45	0.05
	0.05	0.45

Gibbs sampling

- Gibbs sampling:
  - In most cases direct sampling not possible
  - Draw one set of variables at a time

		
	0.45	0.05
	0.05	0.45

(b,g) - draw p(.,g)

Gibbs sampling

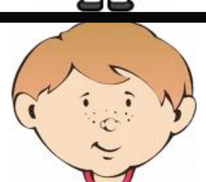
- Gibbs sampling:
  - In most cases direct sampling not possible
  - Draw one set of variables at a time

		
	0.45	0.05
	0.05	0.45

(b,g) - draw p(.,g)
(g,g) - draw p(g,.)

Gibbs sampling

- Gibbs sampling:
  - In most cases direct sampling not possible
  - Draw one set of variables at a time

		
	0.45	0.05
	0.05	0.45

(b,g) - draw p(.,g)
(g,g) - draw p(g,.)
(g,g) - draw p(.,g)

Gibbs sampling

- Gibbs sampling:
  - In most cases direct sampling not possible
  - Draw one set of variables at a time

		
	0.45	0.05
	0.05	0.45

(b,g) - draw p(.,g)
(g,g) - draw p(g,.)
(g,g) - draw p(.,g)
(b,g) - draw p(b,.)

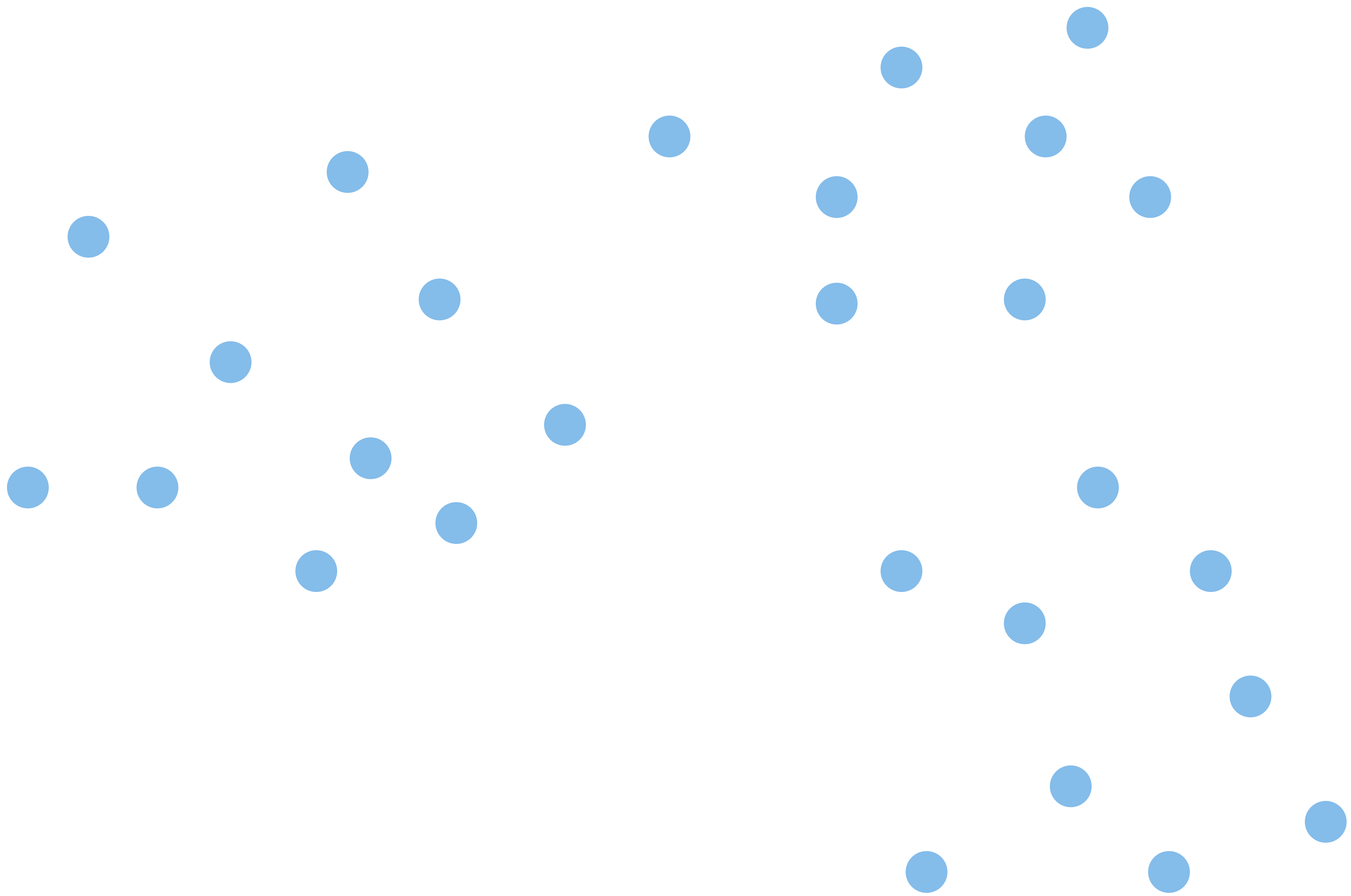
Gibbs sampling

- Gibbs sampling:
  - In most cases direct sampling not possible
  - Draw one set of variables at a time

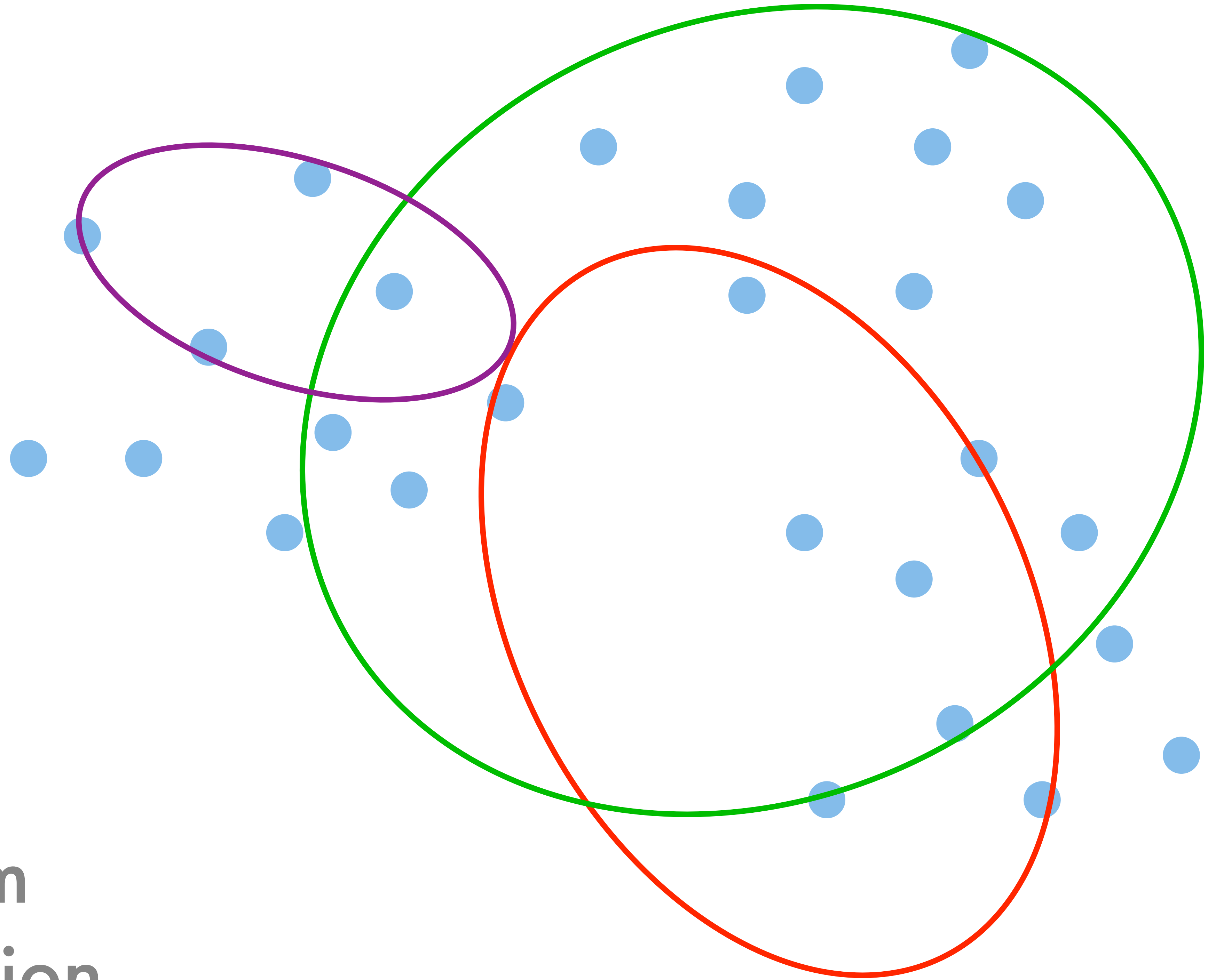
		
	0.45	0.05
	0.05	0.45

(b,g) - draw p(.,g)
(g,g) - draw p(g,.)
(g,g) - draw p(.,g)
(b,g) - draw p(b,.)
(b,b) ...

Gibbs sampling for clustering

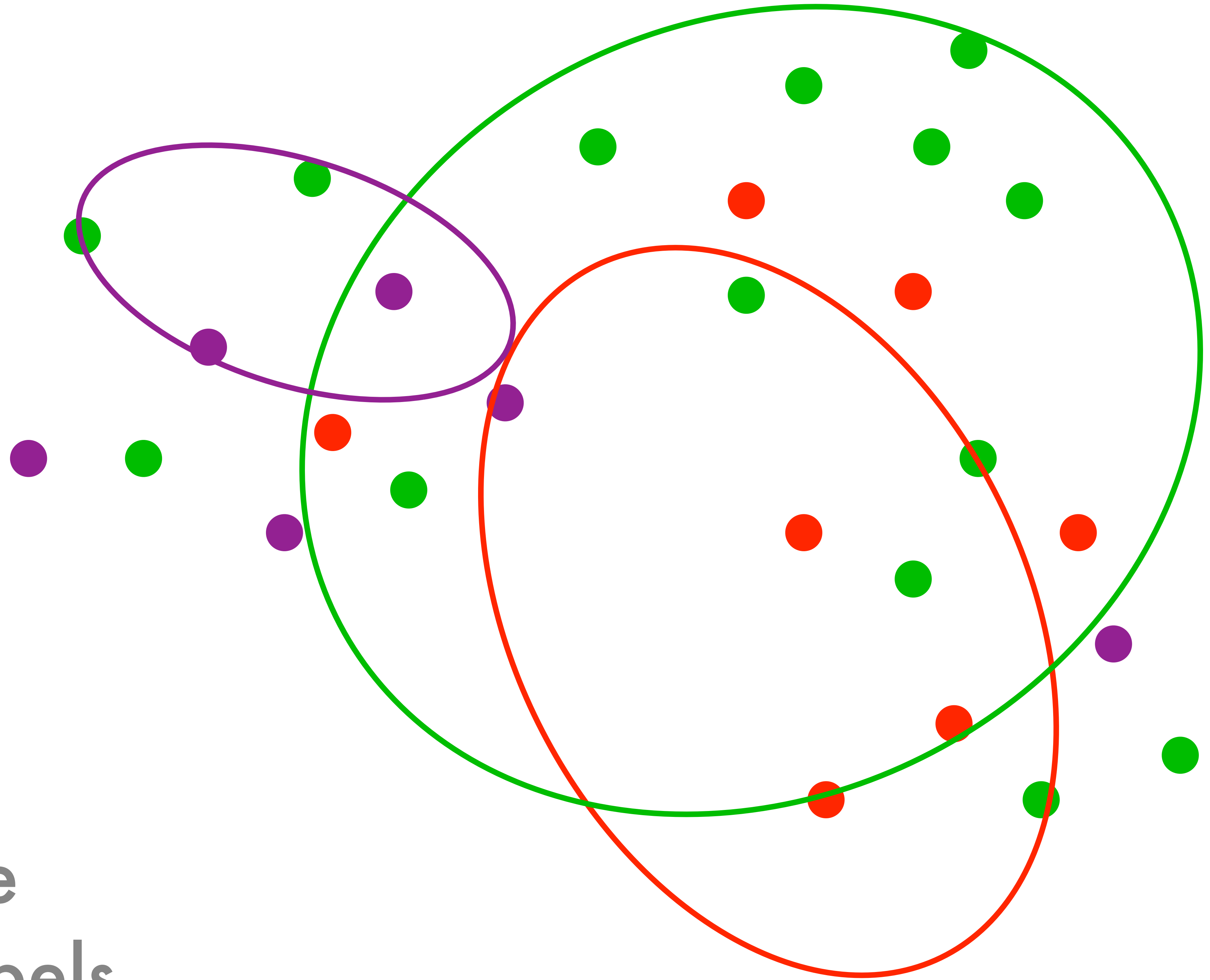


Gibbs sampling for clustering



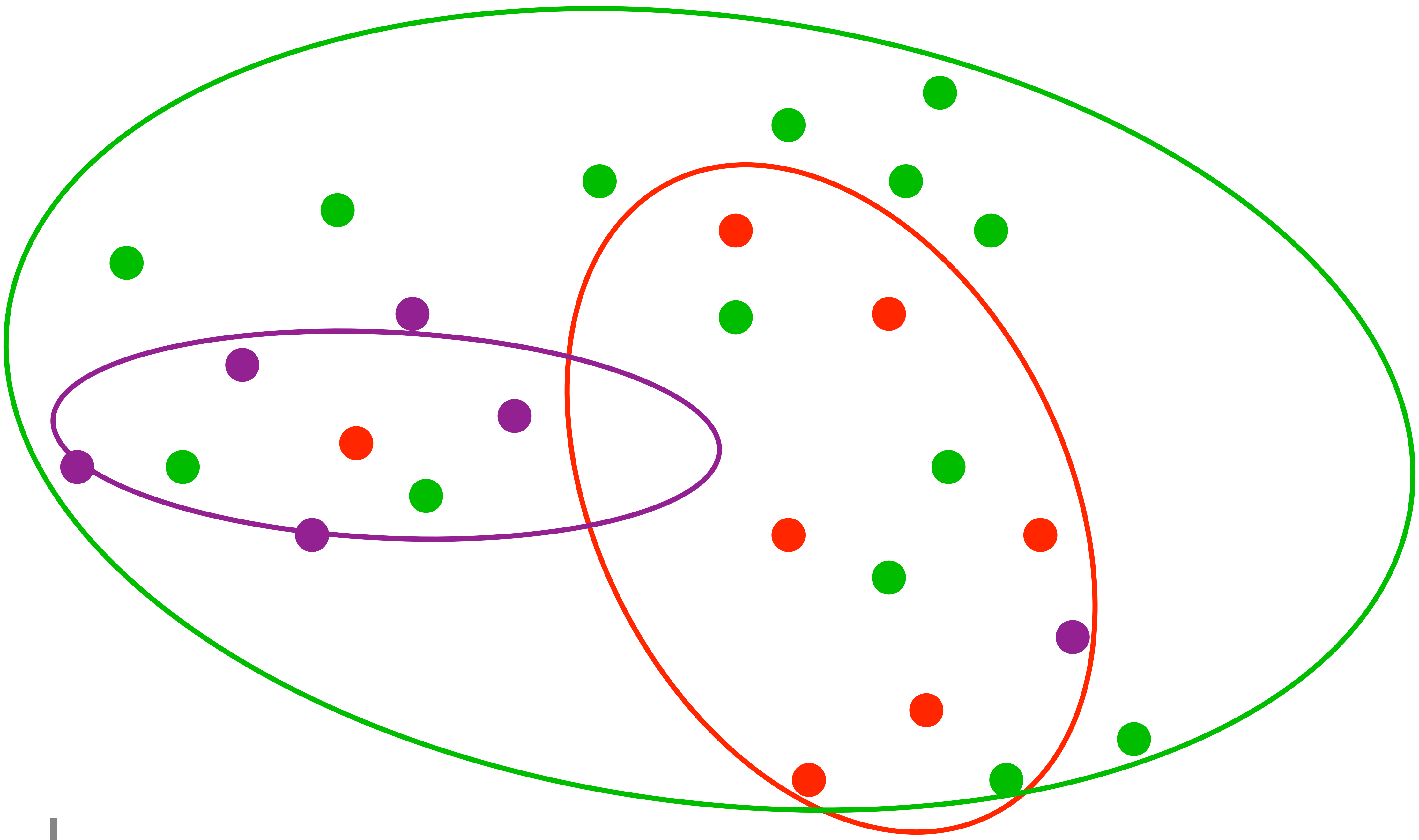
random
initialization

Gibbs sampling for clustering



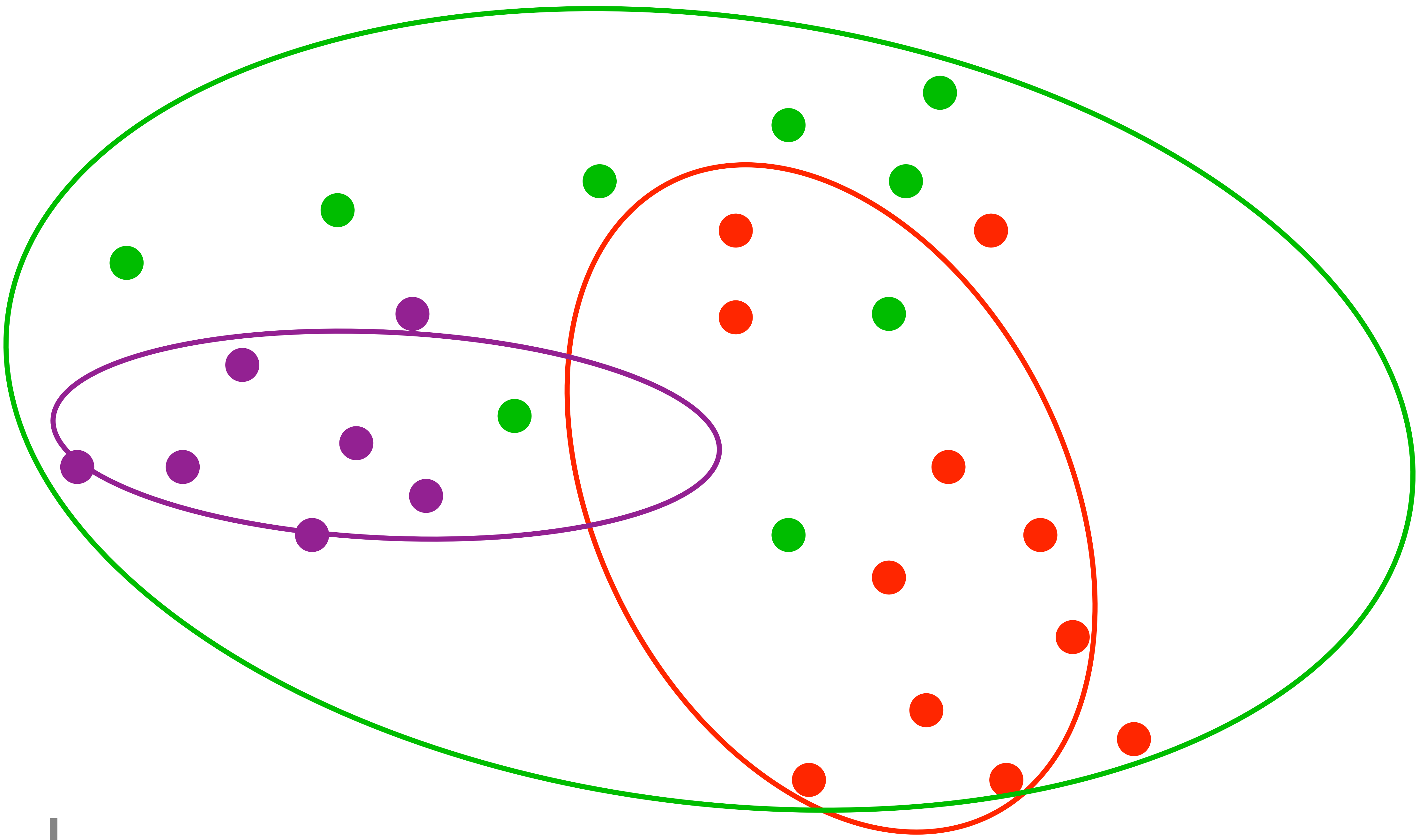
sample
cluster labels

Gibbs sampling for clustering



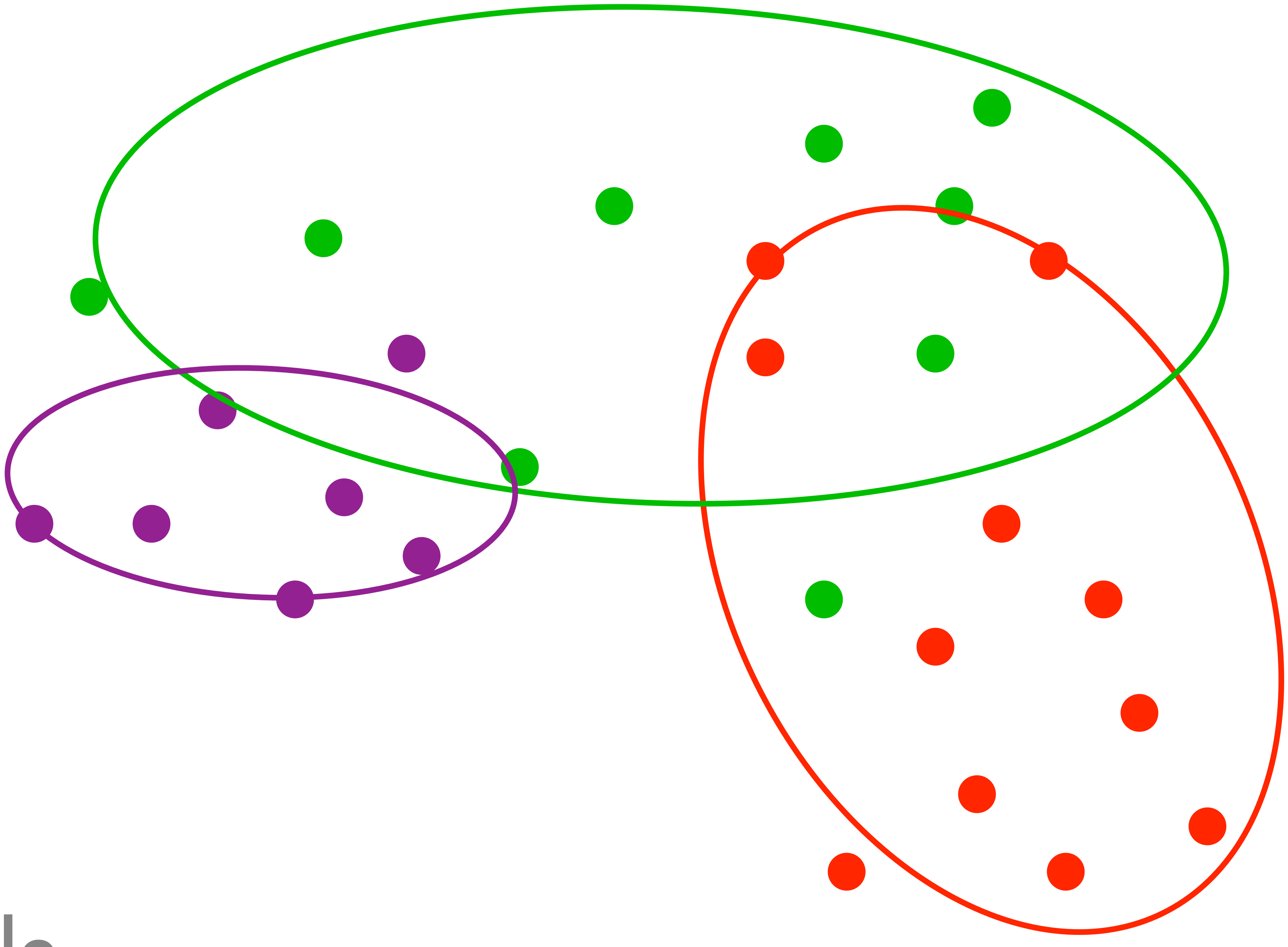
resample
cluster model

Gibbs sampling for clustering



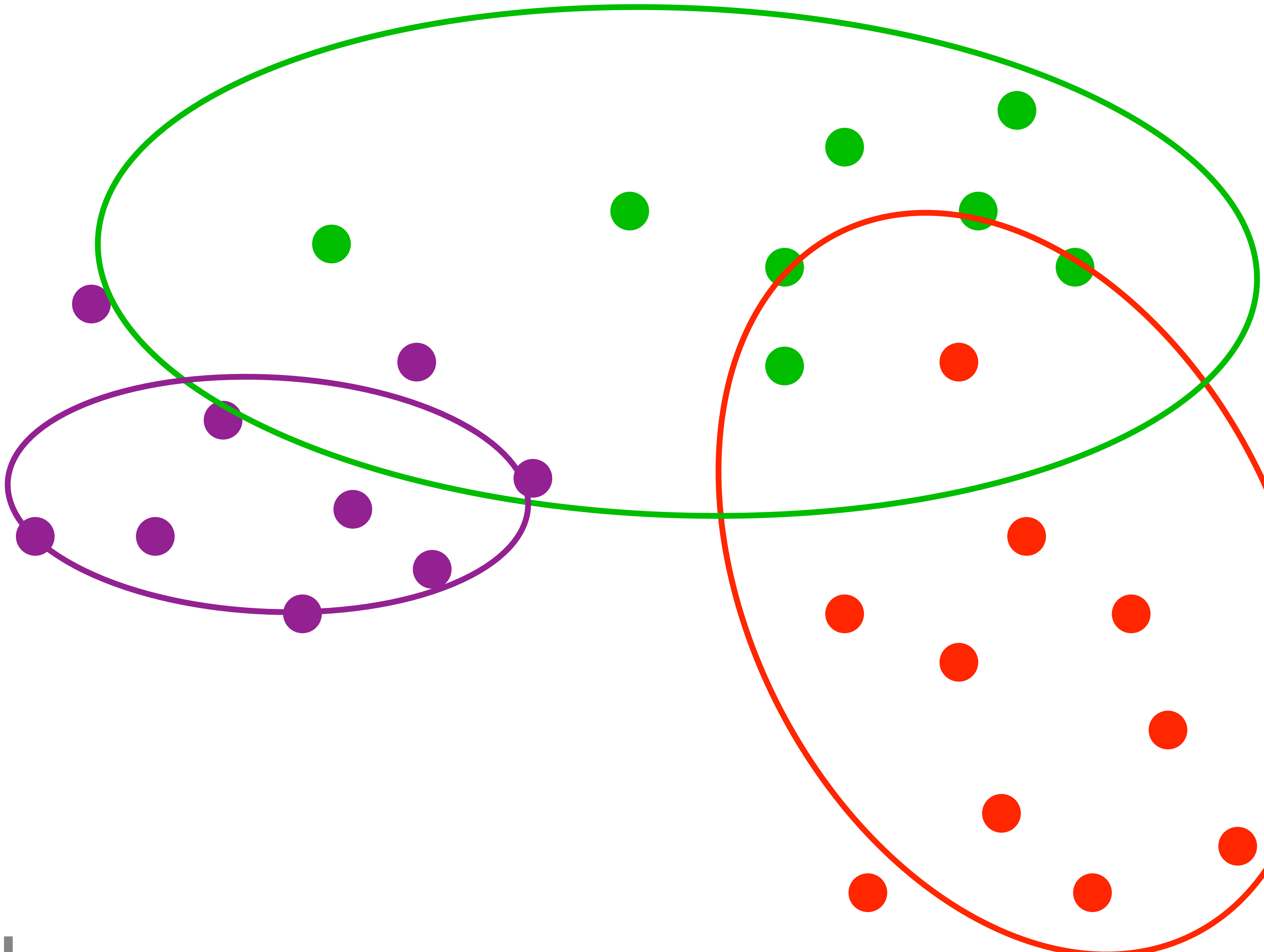
resample
cluster labels

Gibbs sampling for clustering



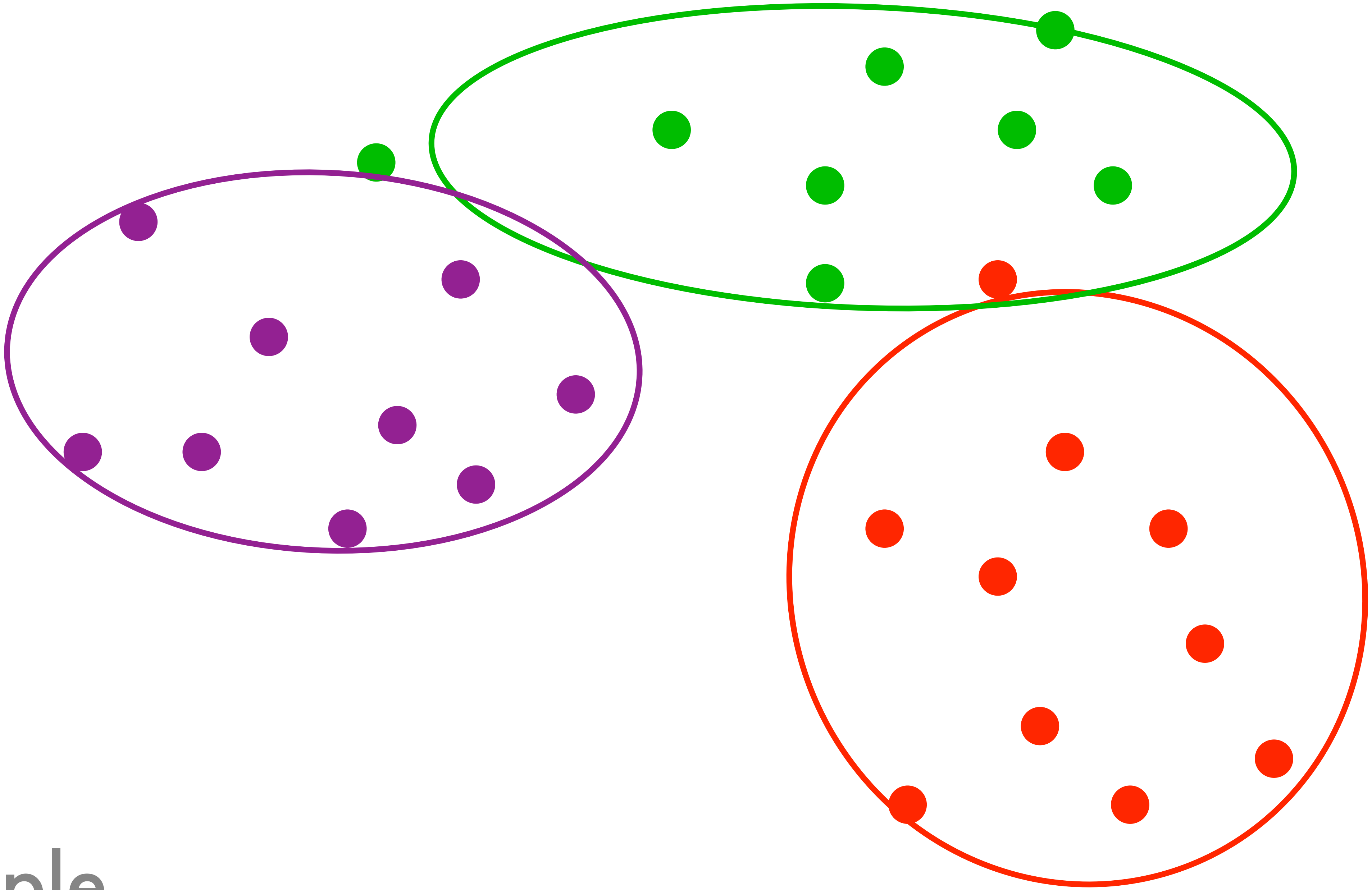
resample
cluster model

Gibbs sampling for clustering



resample
cluster labels

Gibbs sampling for clustering



resample
cluster model

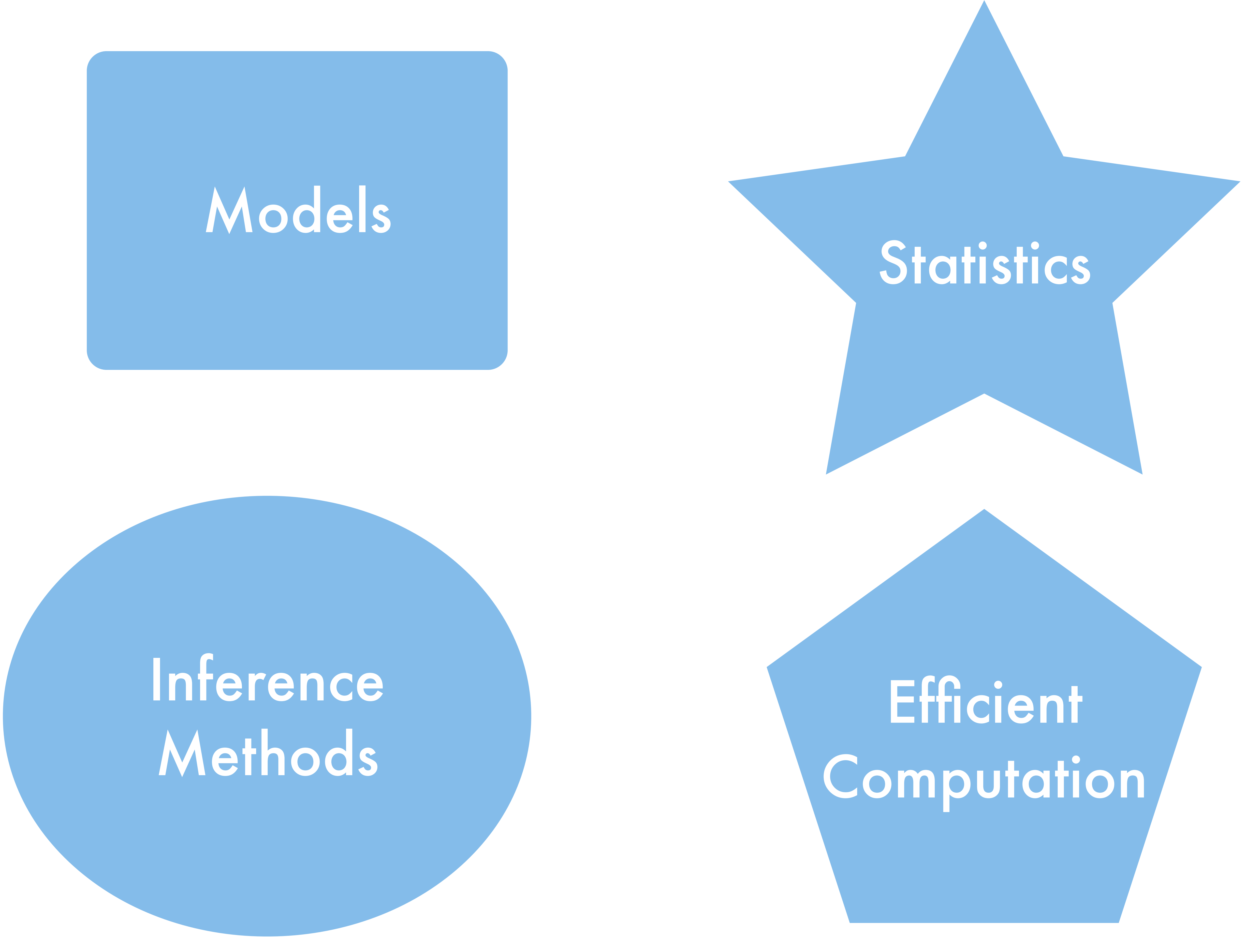
e.g. Mahout Dirichlet Process Clustering

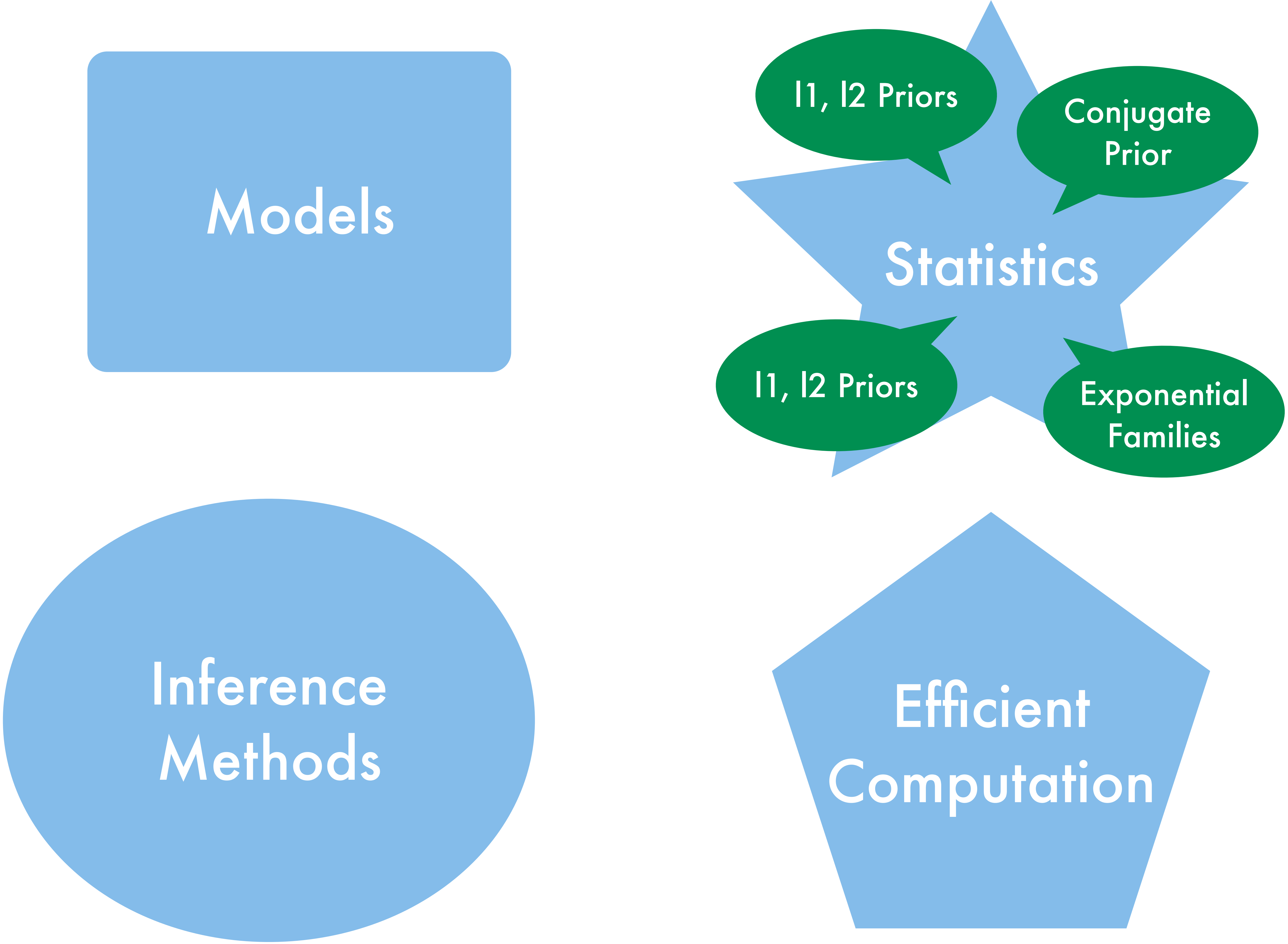
Inference Algorithm ≠ Model

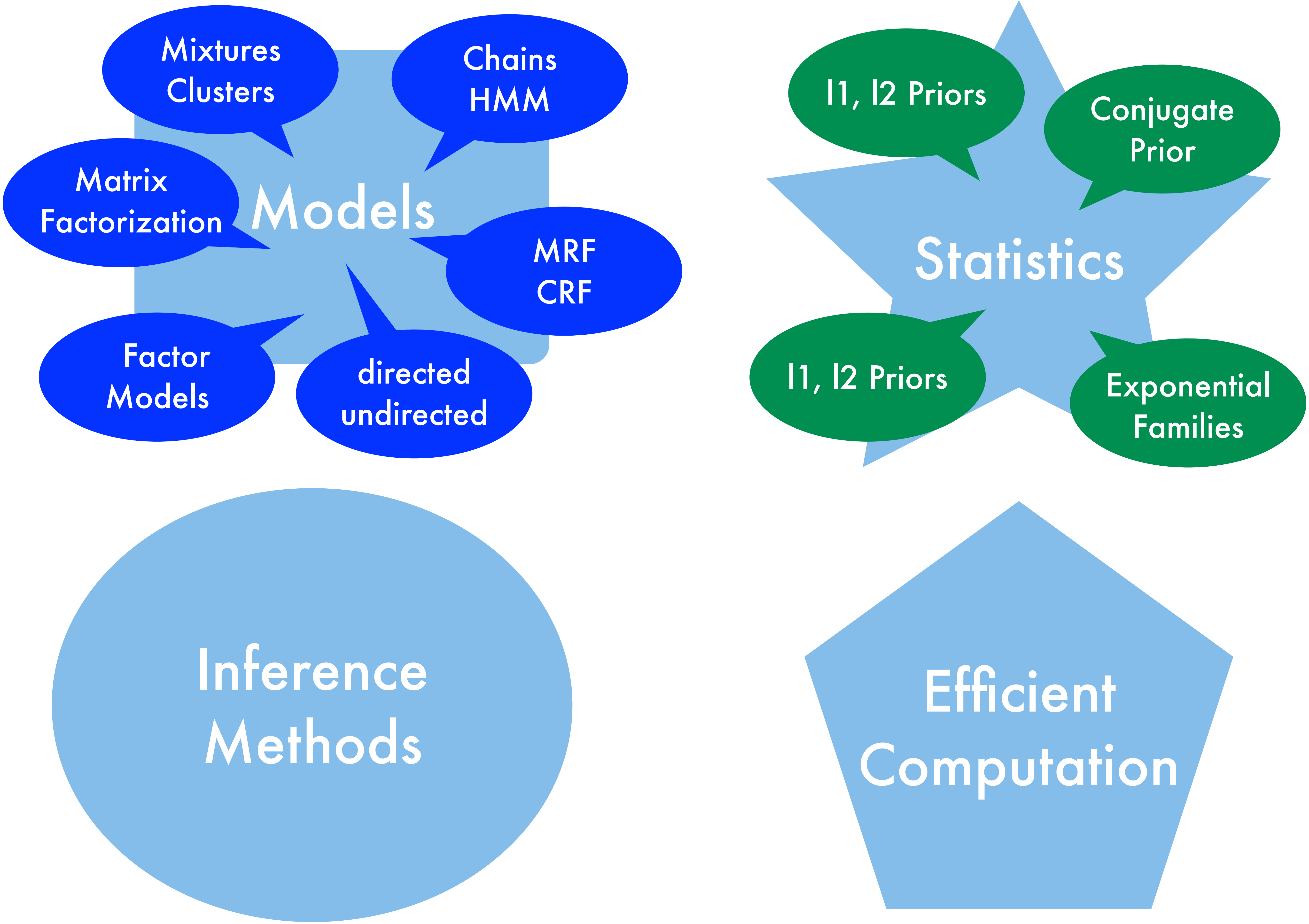
Inference Algorithm ≠ Model

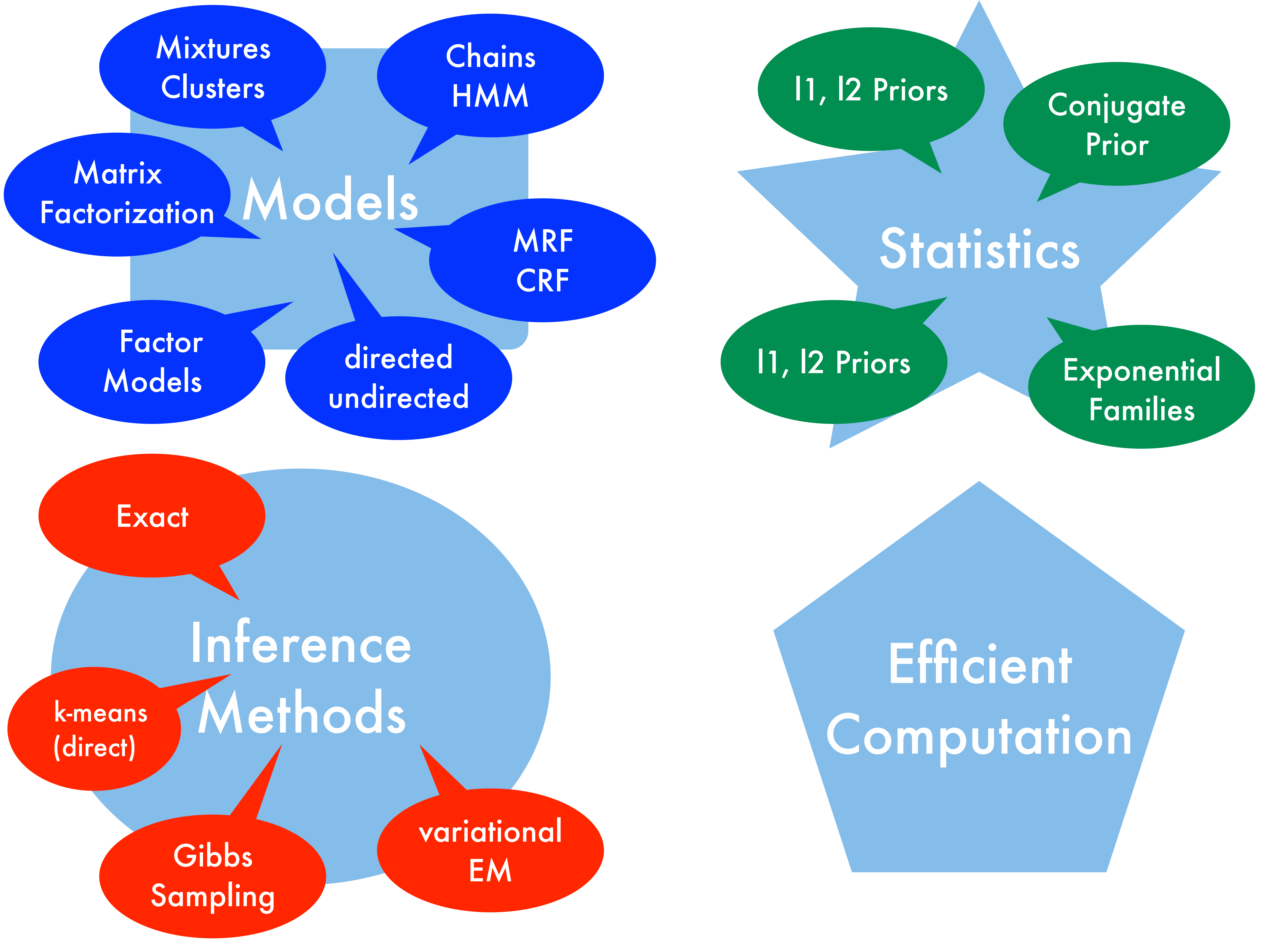
Corollary: EM ≠ Clustering

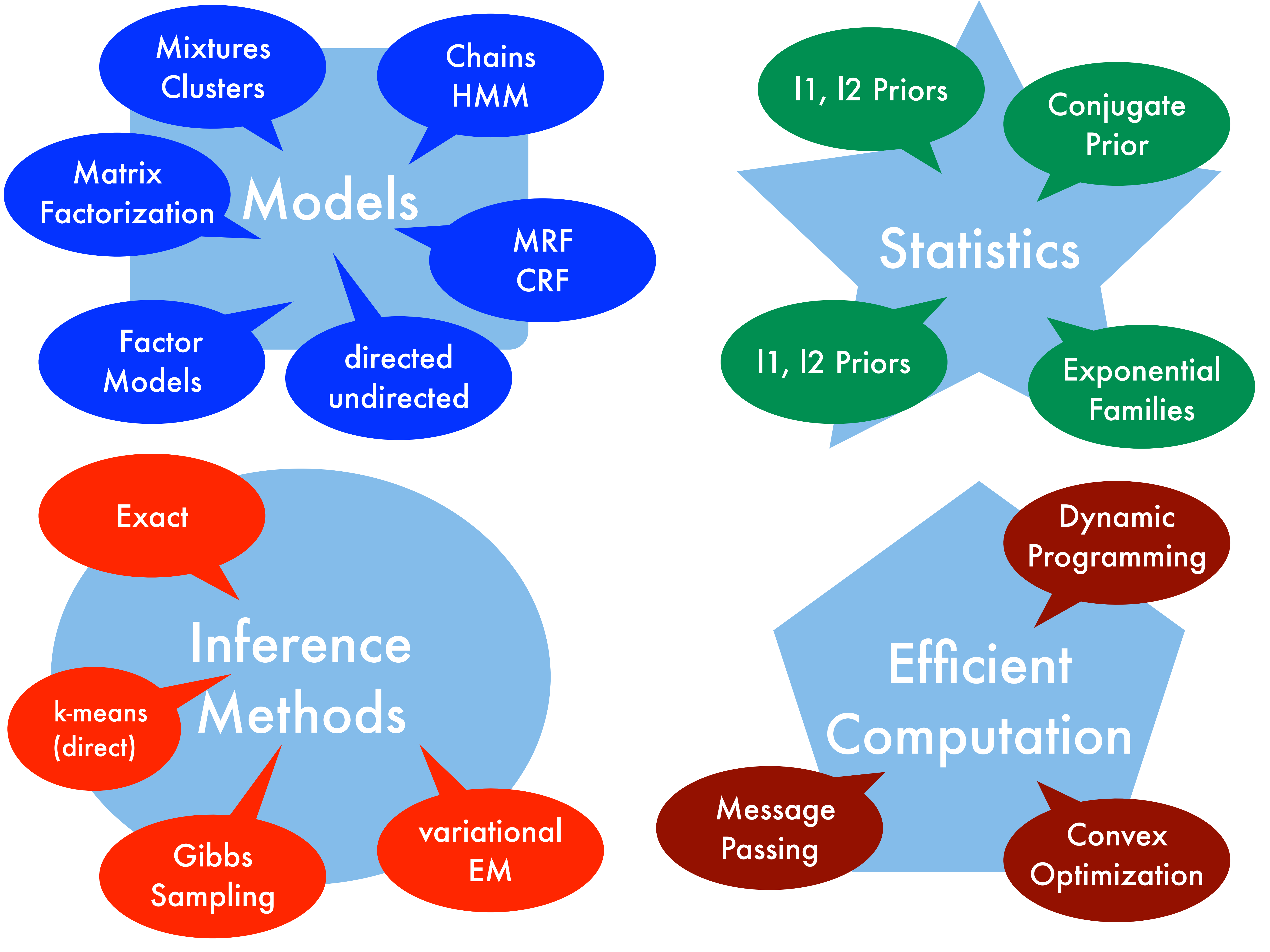
Graphical Models Zoology


Models
Statistics
Inference Methods
Efficient Computation


Models
l1, l2 Priors
Conjugate Prior
Statistics
l1, l2 Priors
Exponential Families
Inference Methods
Efficient Computation


Mixtures
Clusters
Chains
HMM
Matrix
Factorization
Models
MRF
CRF
Factor
Models
directed
undirected
l1, l2 Priors
Conjugate
Prior
Statistics
l1, l2 Priors
Exponential
Families
Inference
Methods
Efficient
Computation


Mixtures
Clusters
Chains
HMM
Matrix
Factorization
Models
MRF
CRF
Factor
Models
directed
undirected
l1, l2 Priors
Conjugate
Prior
Statistics
l1, l2 Priors
Exponential
Families
Exact
Inference
Methods
k-means
(direct)
Gibbs
Sampling
variational
EM
Efficient
Computation

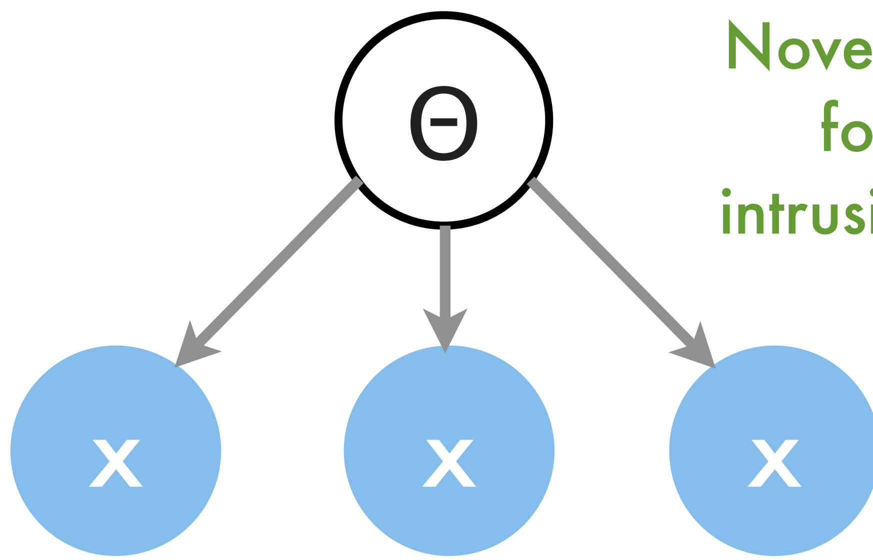

Mixtures
Clusters
Chains
HMM
Matrix
Factorization
Models
MRF
CRF
Factor
Models
directed
undirected
l1, l2 Priors
Conjugate
Prior
Statistics
l1, l2 Priors
Exponential
Families
Exact
Inference
Methods
k-means
(direct)
Gibbs
Sampling
variational
EM
Dynamic
Programming
Efficient
Computation
Message
Passing
Convex
Optimization

YAHOO!®

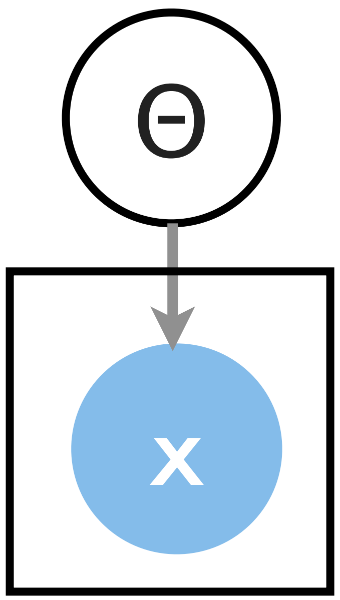

YAHOO!
Spam Filtering
Classification
Exploration
Segmentation
Annotation
System Design
Prediction (time series)
Clustering
Novelty Detection
Document Understanding
User Modeling
Advertising
Debugging
Performance Tuning

‘Unsupervised’ Models

Density Estimation



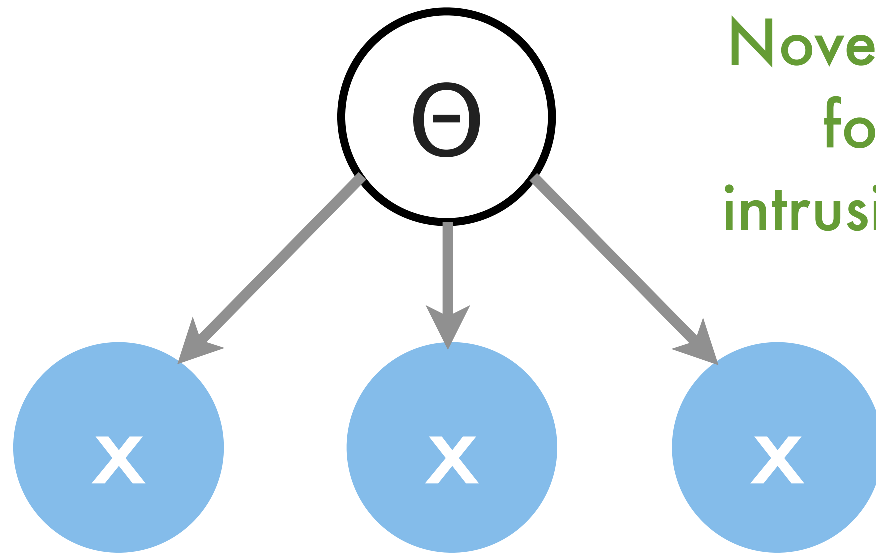
Novelty Detection
forecasting
intrusion detection



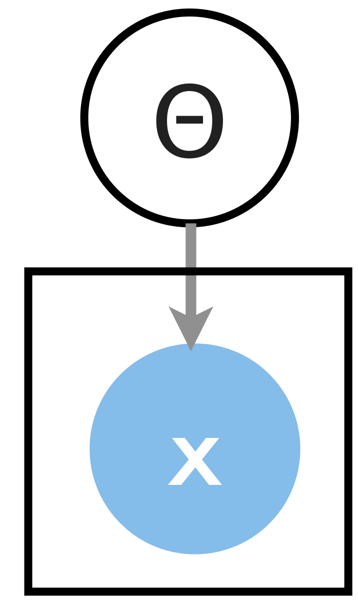
webpages
news
users
ads
queries
images

'Unsupervised' Models

Density Estimation

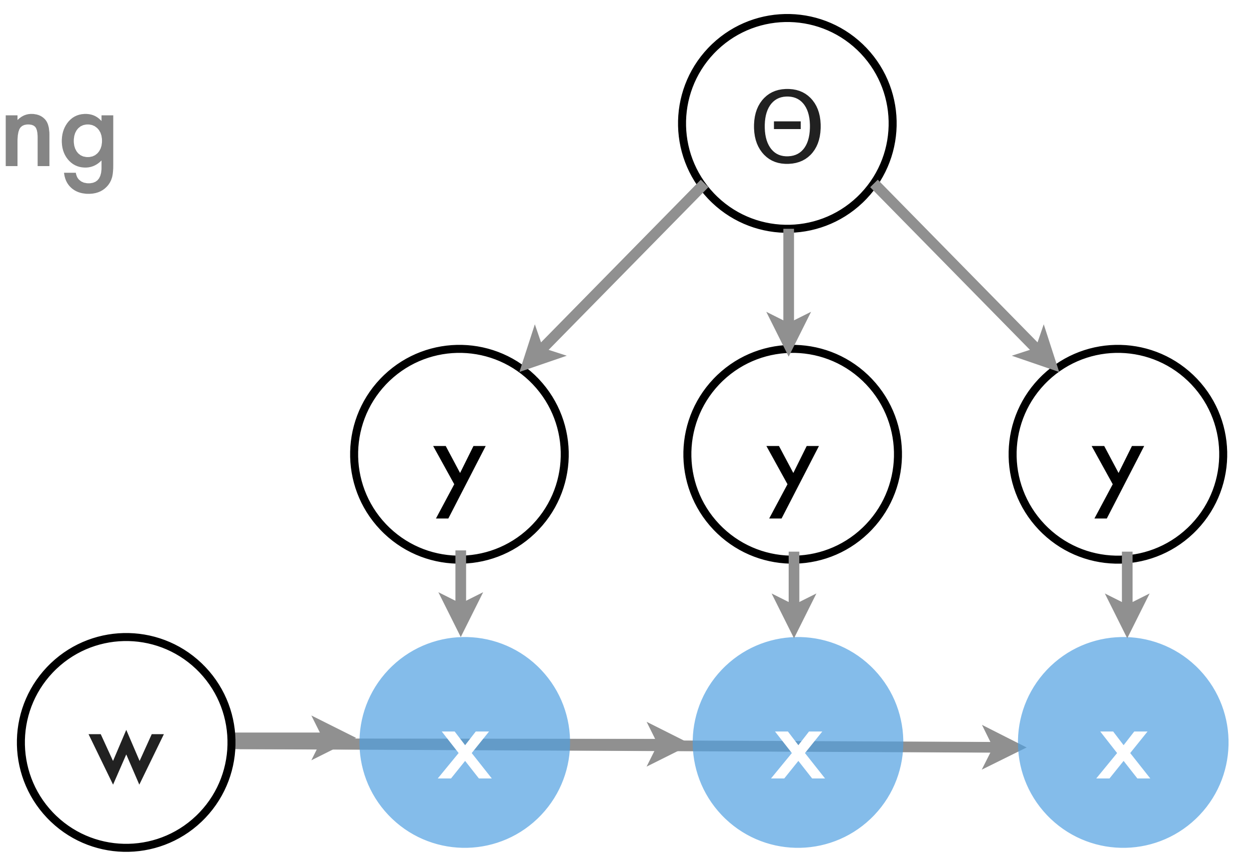


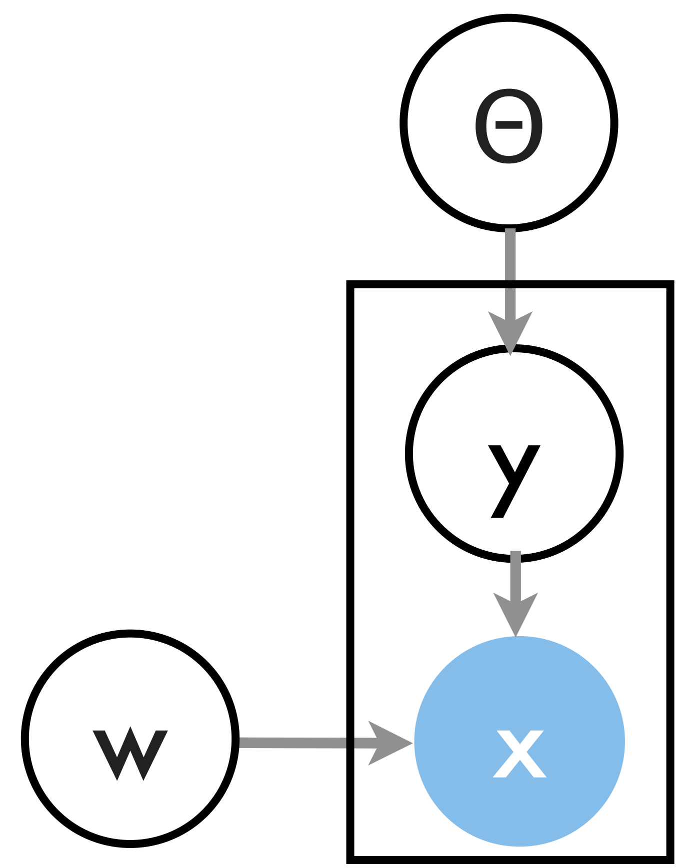
Novelty Detection
forecasting
intrusion detection



Clustering

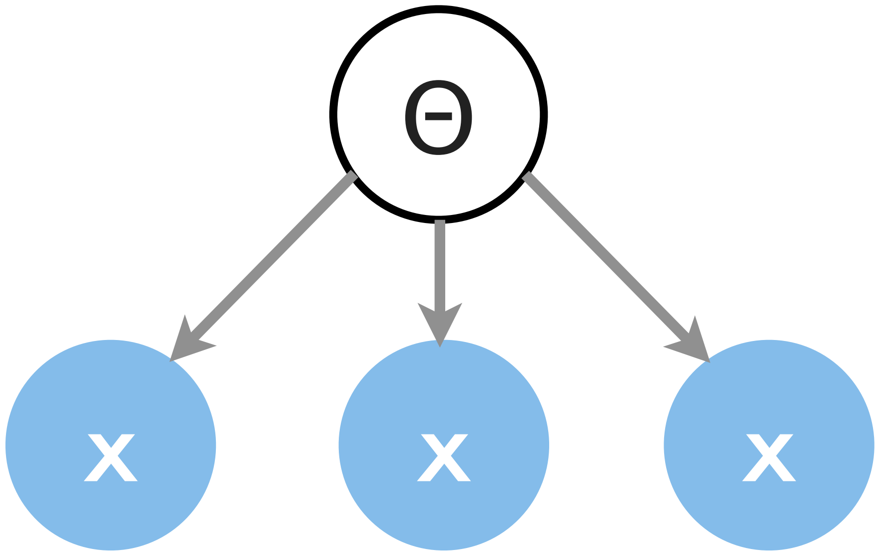
webpages
news
users
ads
queries
images

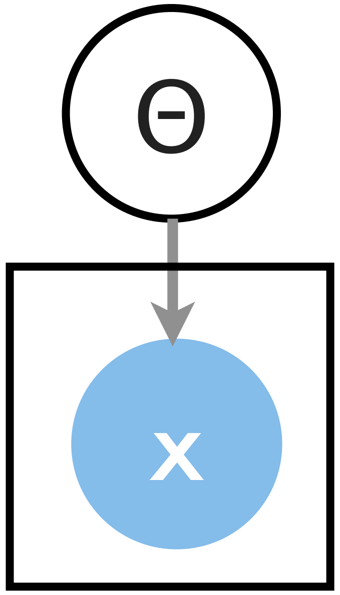




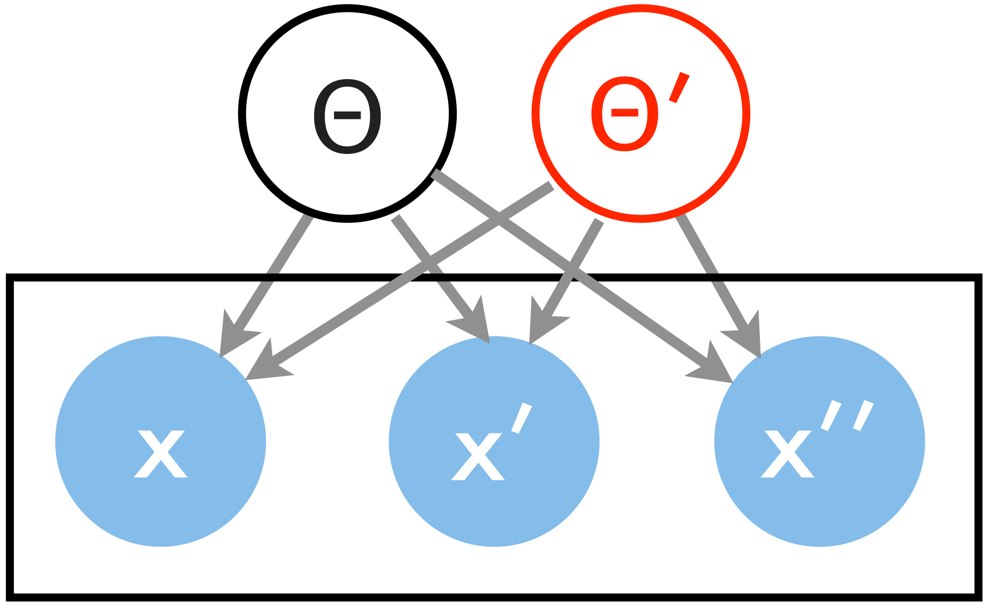
‘Unsupervised’ Models

Density Estimation





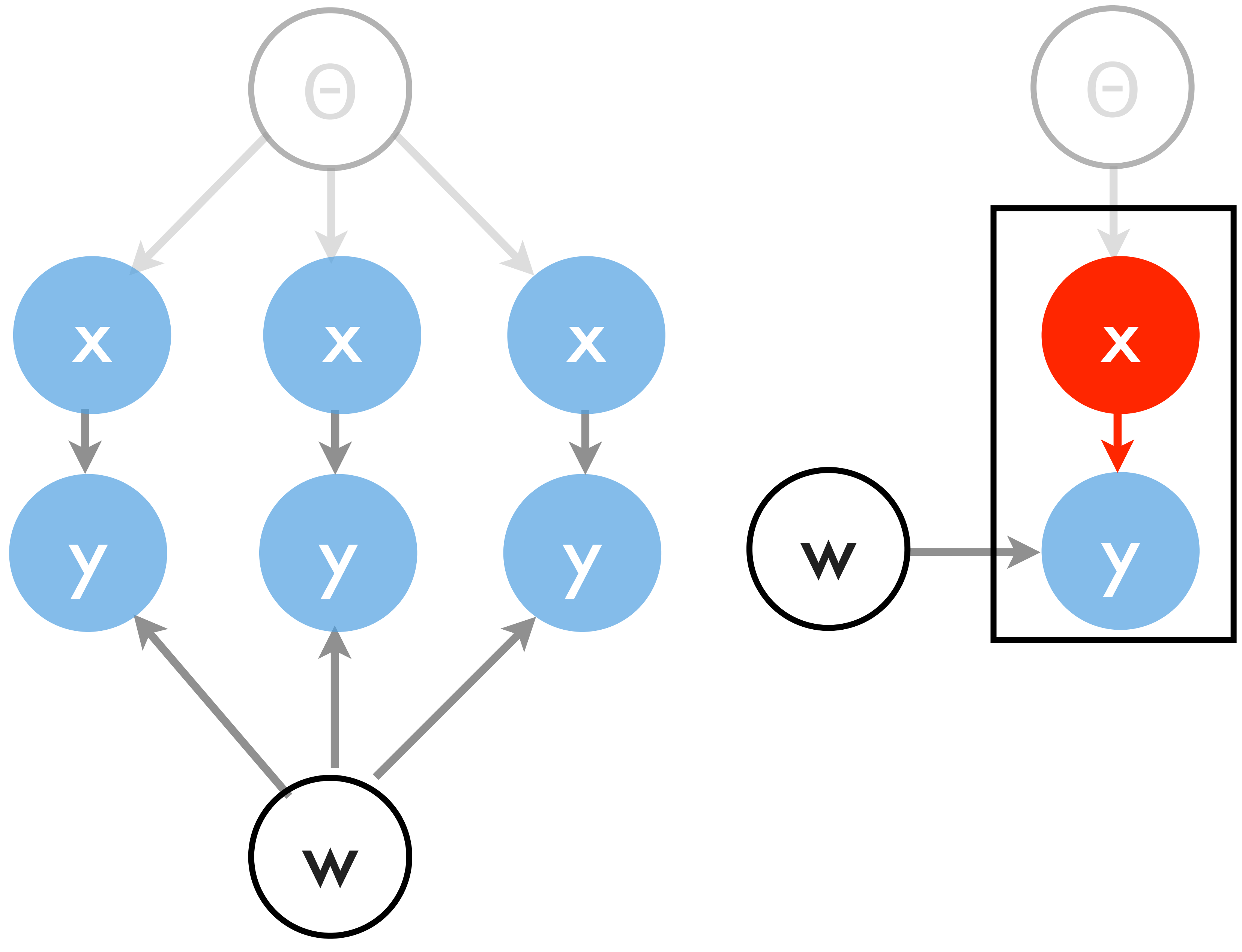
Factor Analysis



'Supervised' Models

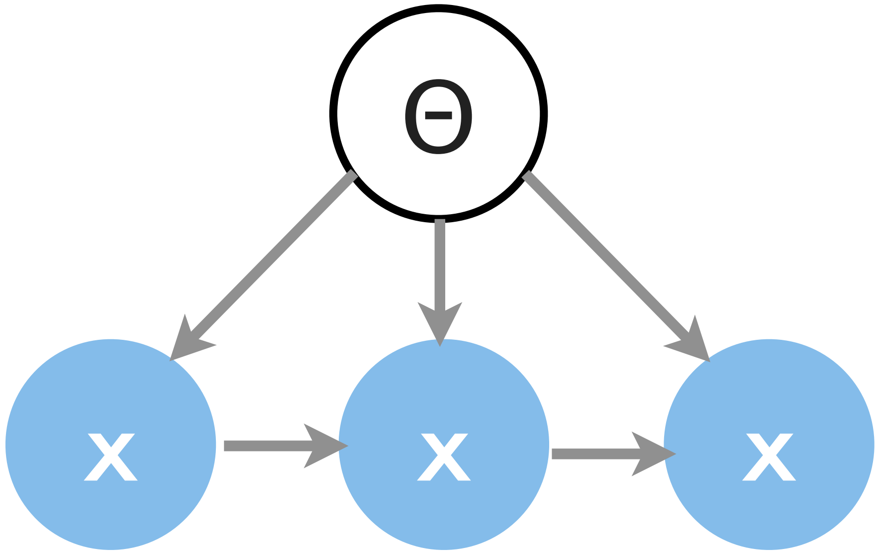
Classification
Regression

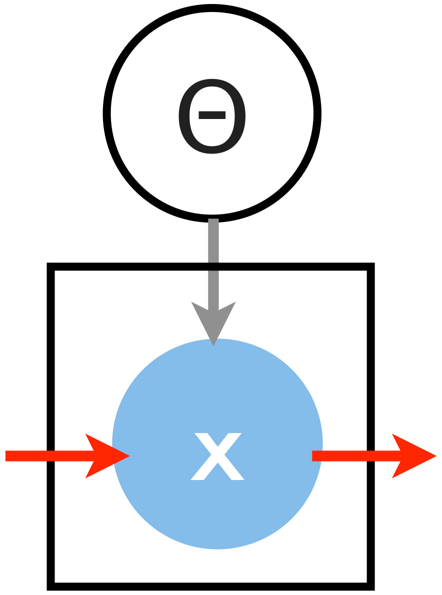
spam filtering
tiering
crawling
categorization
bid estimation
tagging



Chains

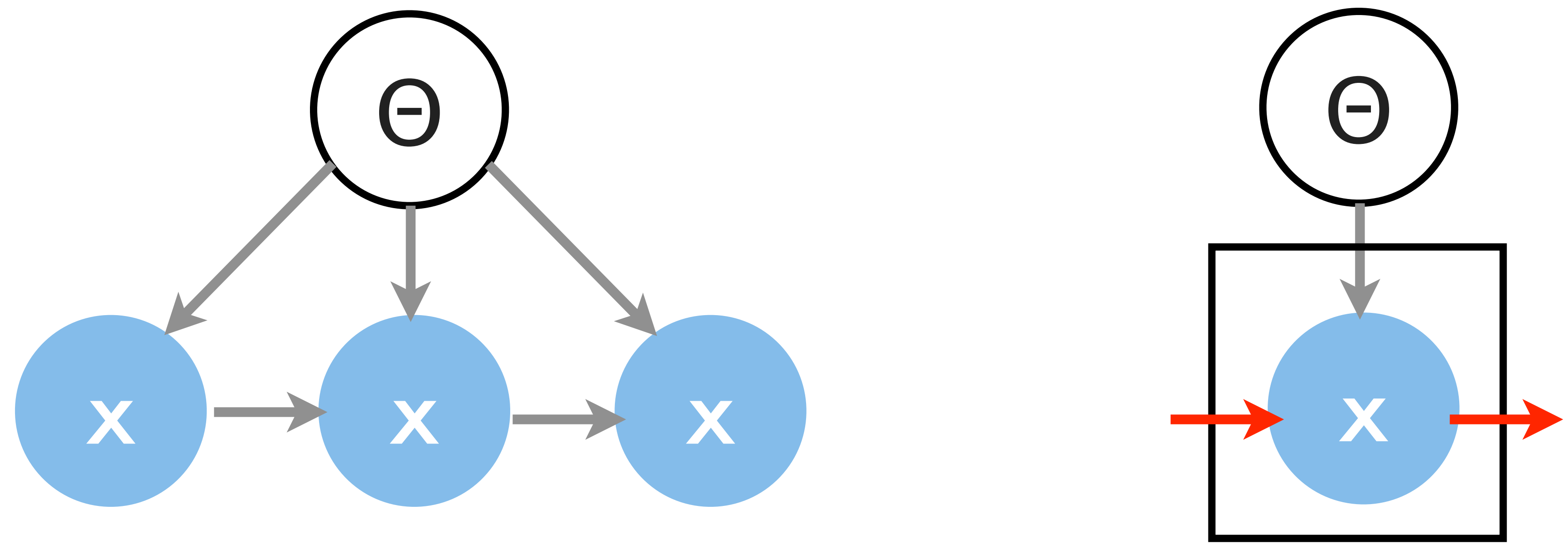
Markov
Chain



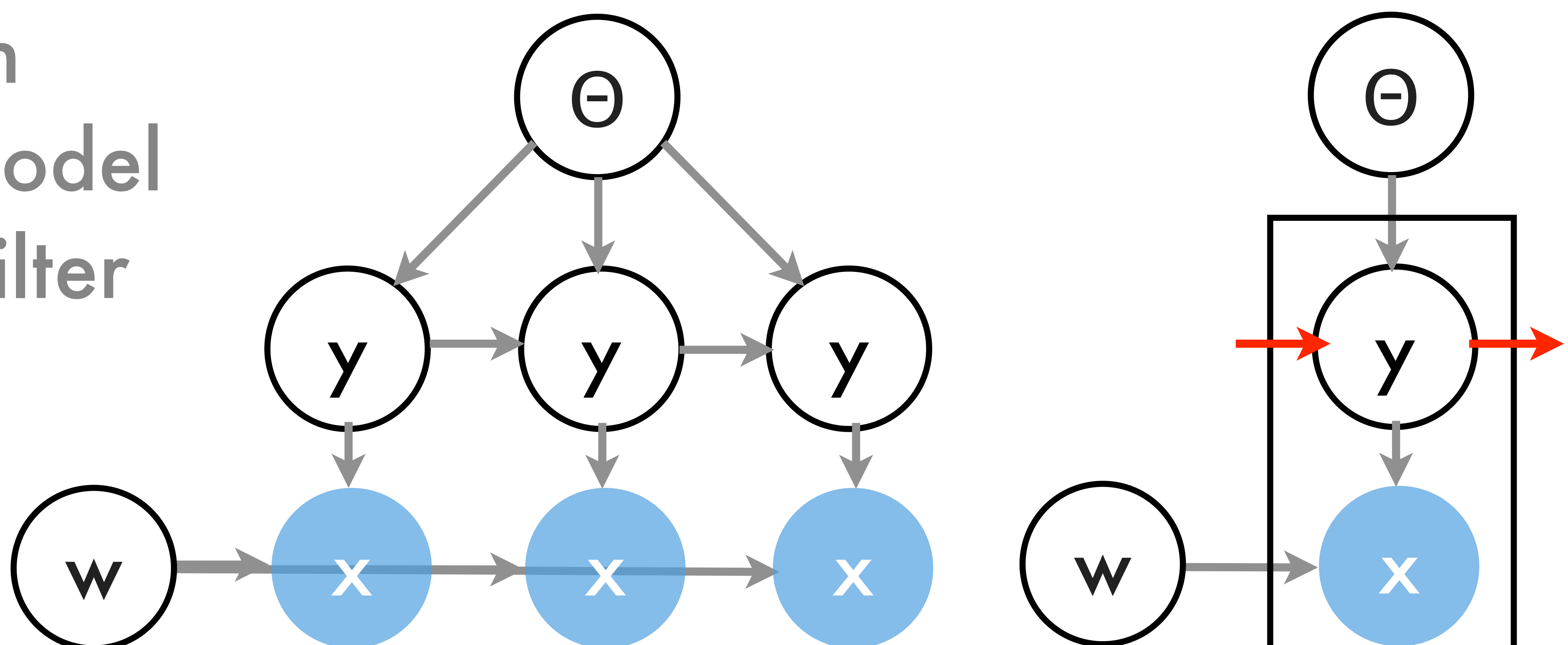


Chains

Markov
Chain



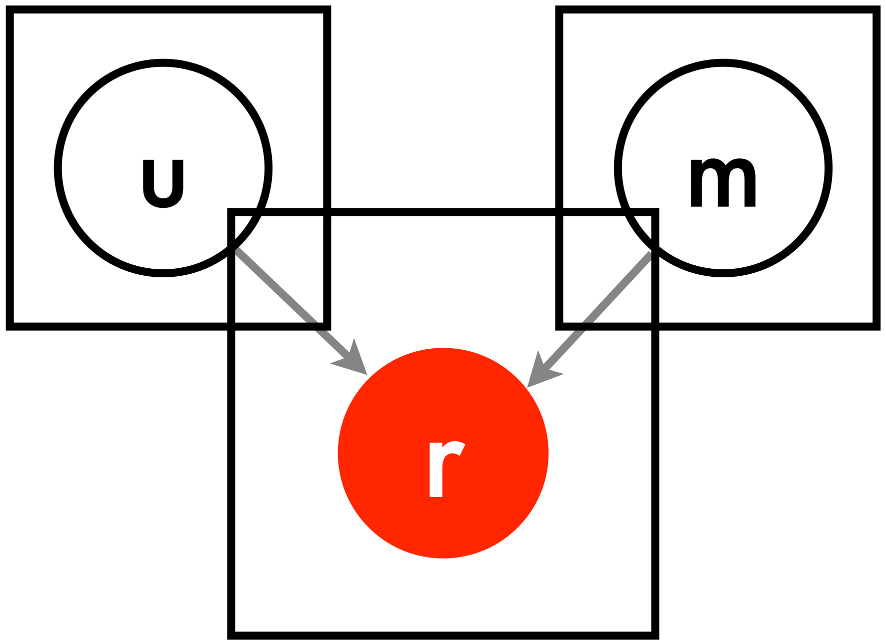
Hidden
Markov Model
Kalman Filter



Collaborative Models

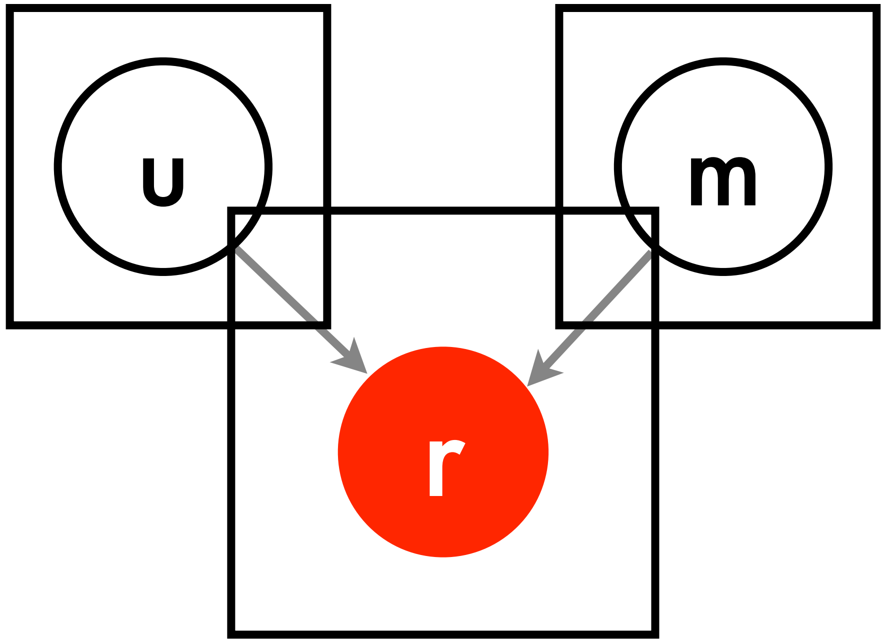
Collaborative Models

Collaborative Filtering

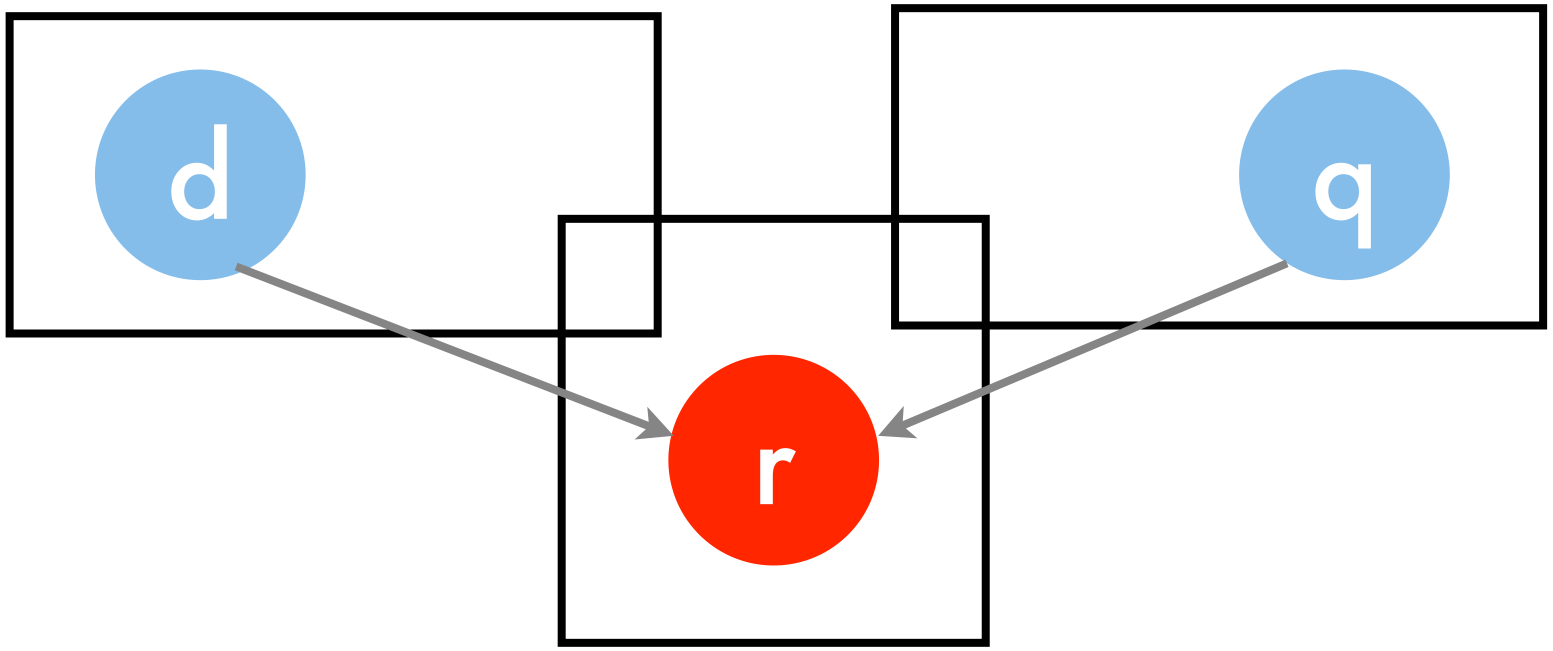


Collaborative Models

Collaborative Filtering

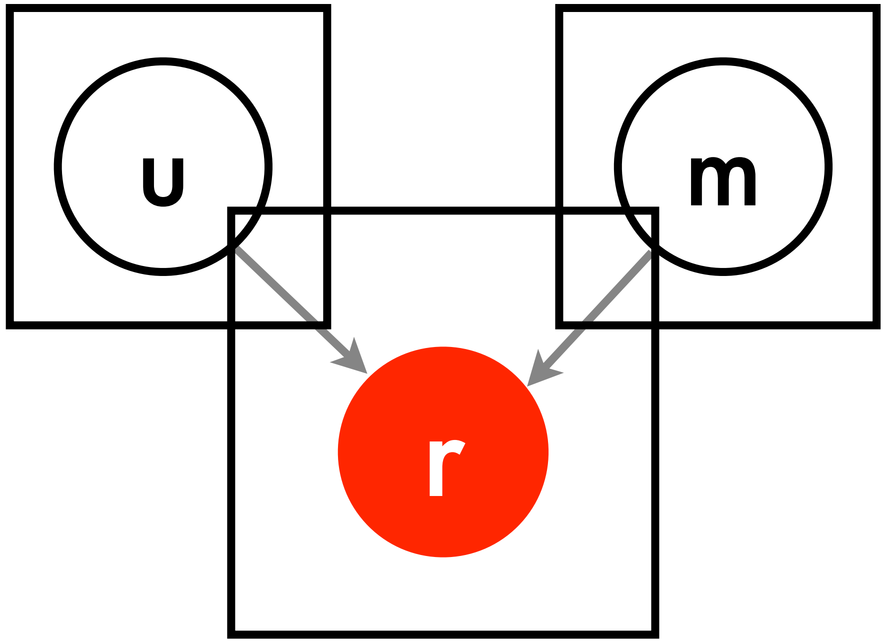


Current Webpage Ranking

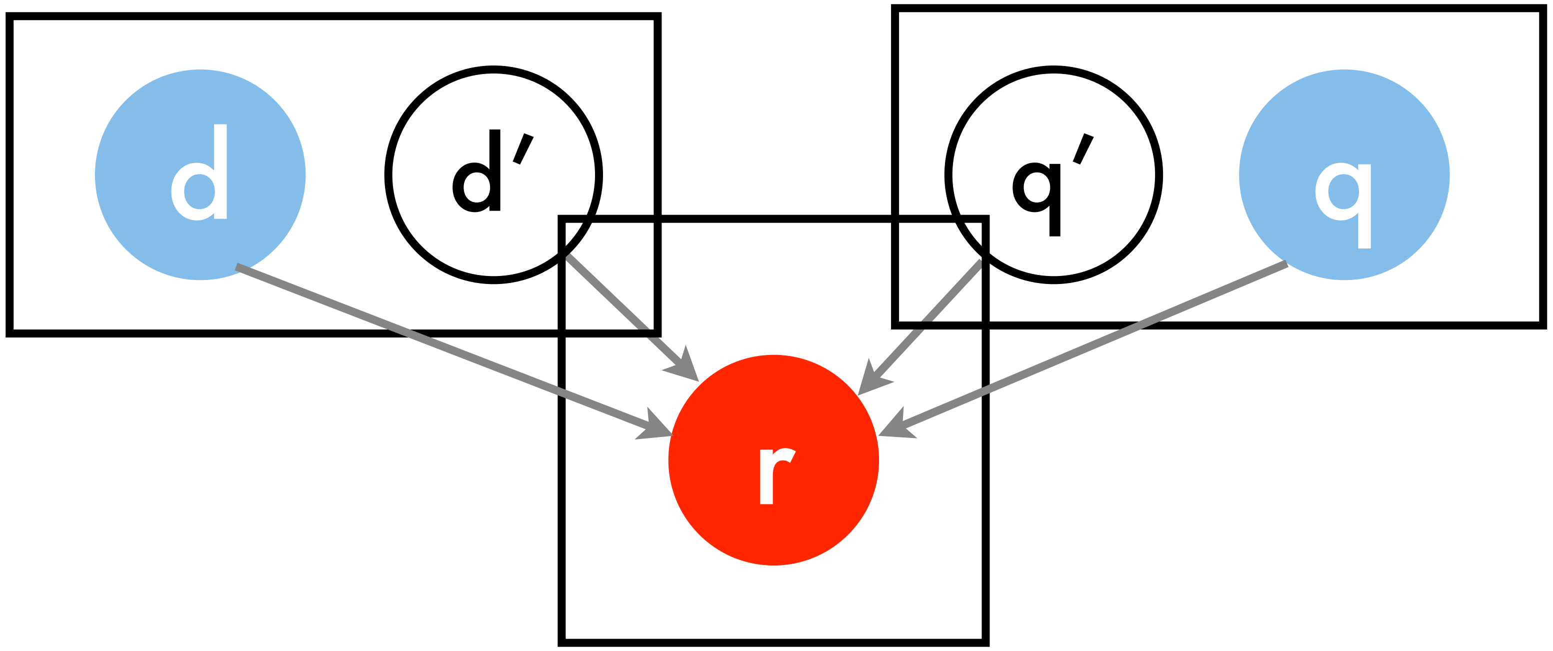


Collaborative Models

Collaborative Filtering

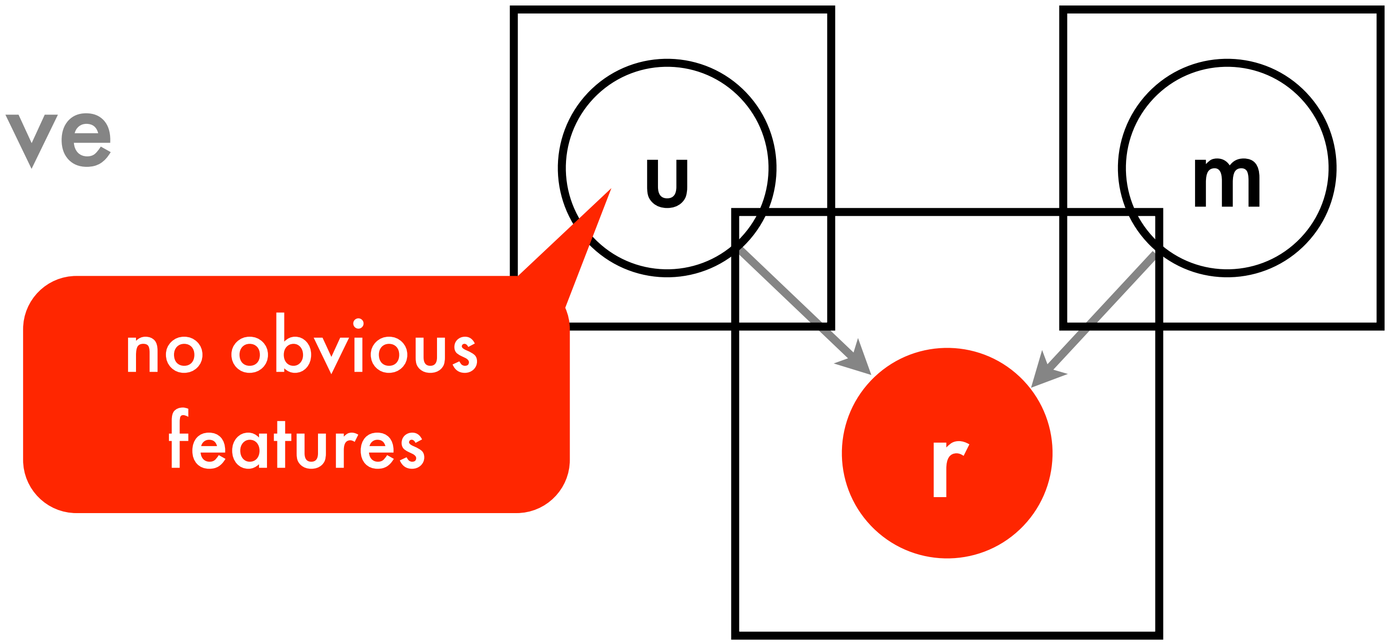


Webpage Ranking

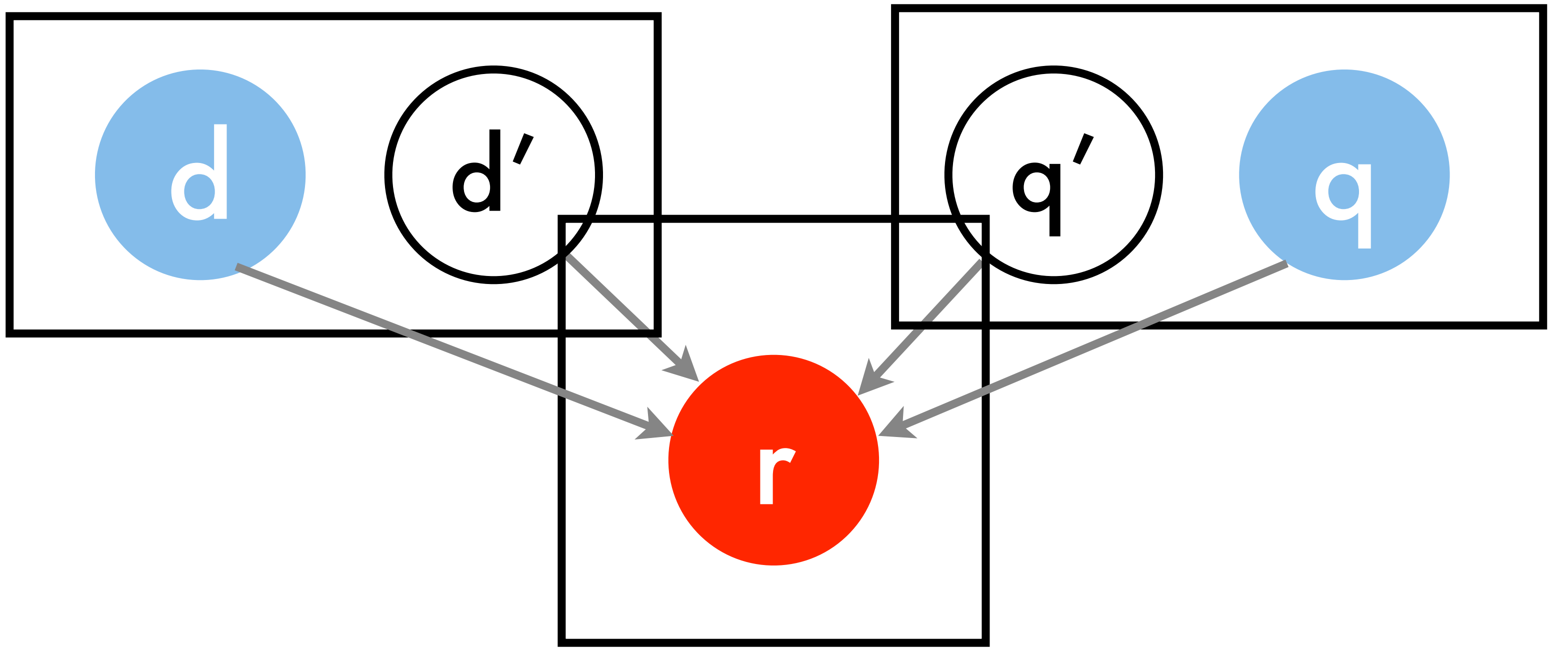


Collaborative Models

Collaborative Filtering



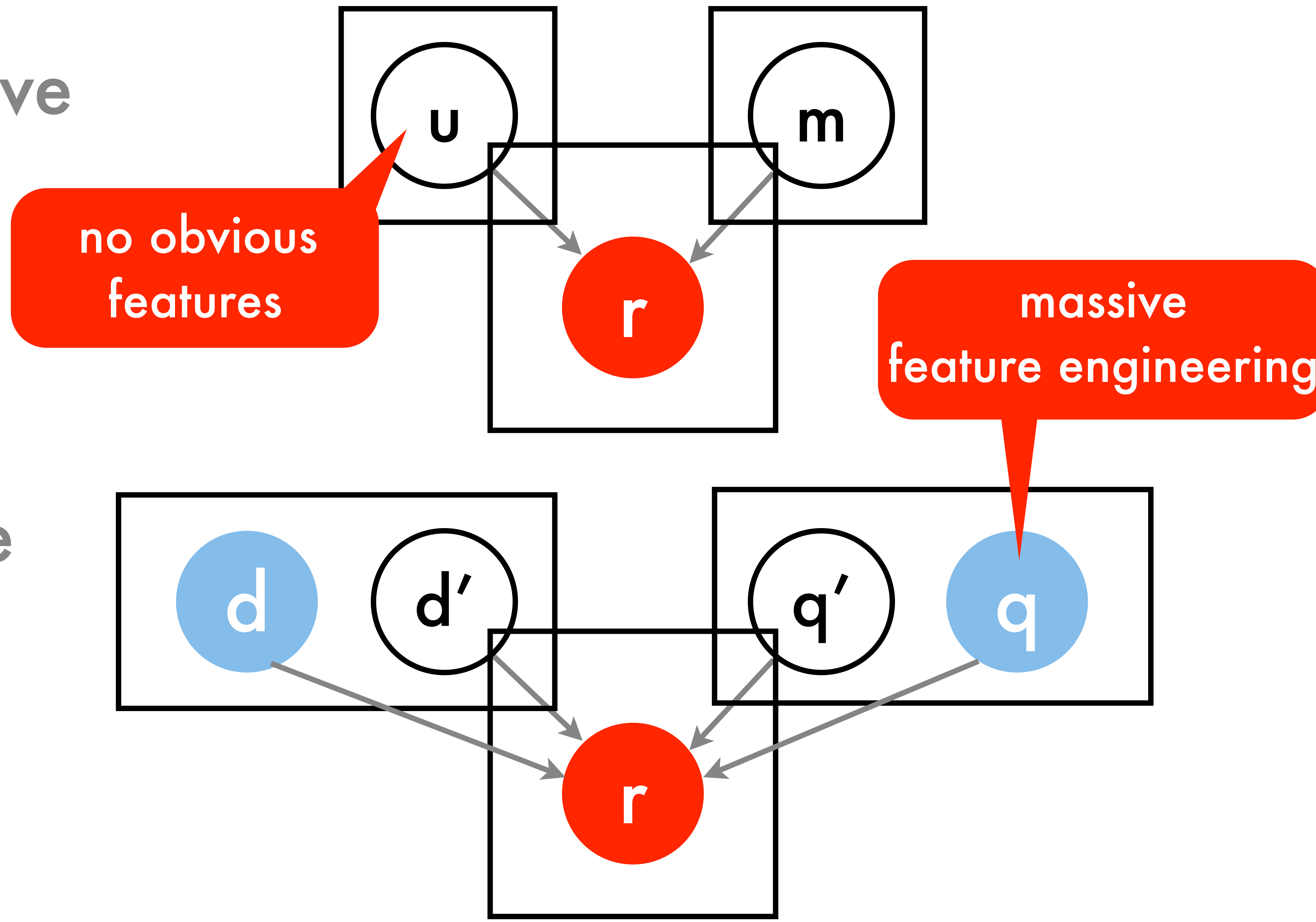
Webpage Ranking



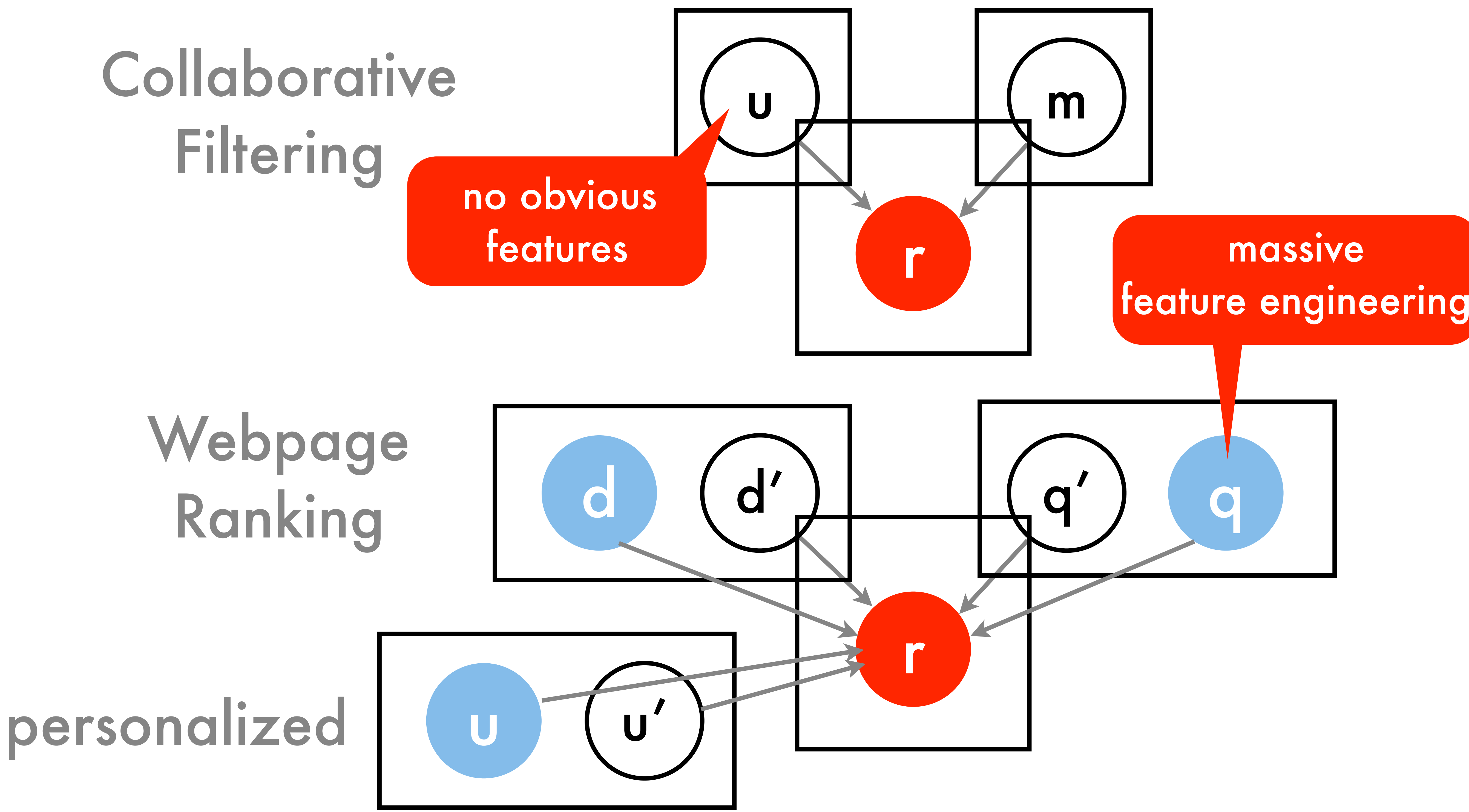
Collaborative Models

Collaborative Filtering

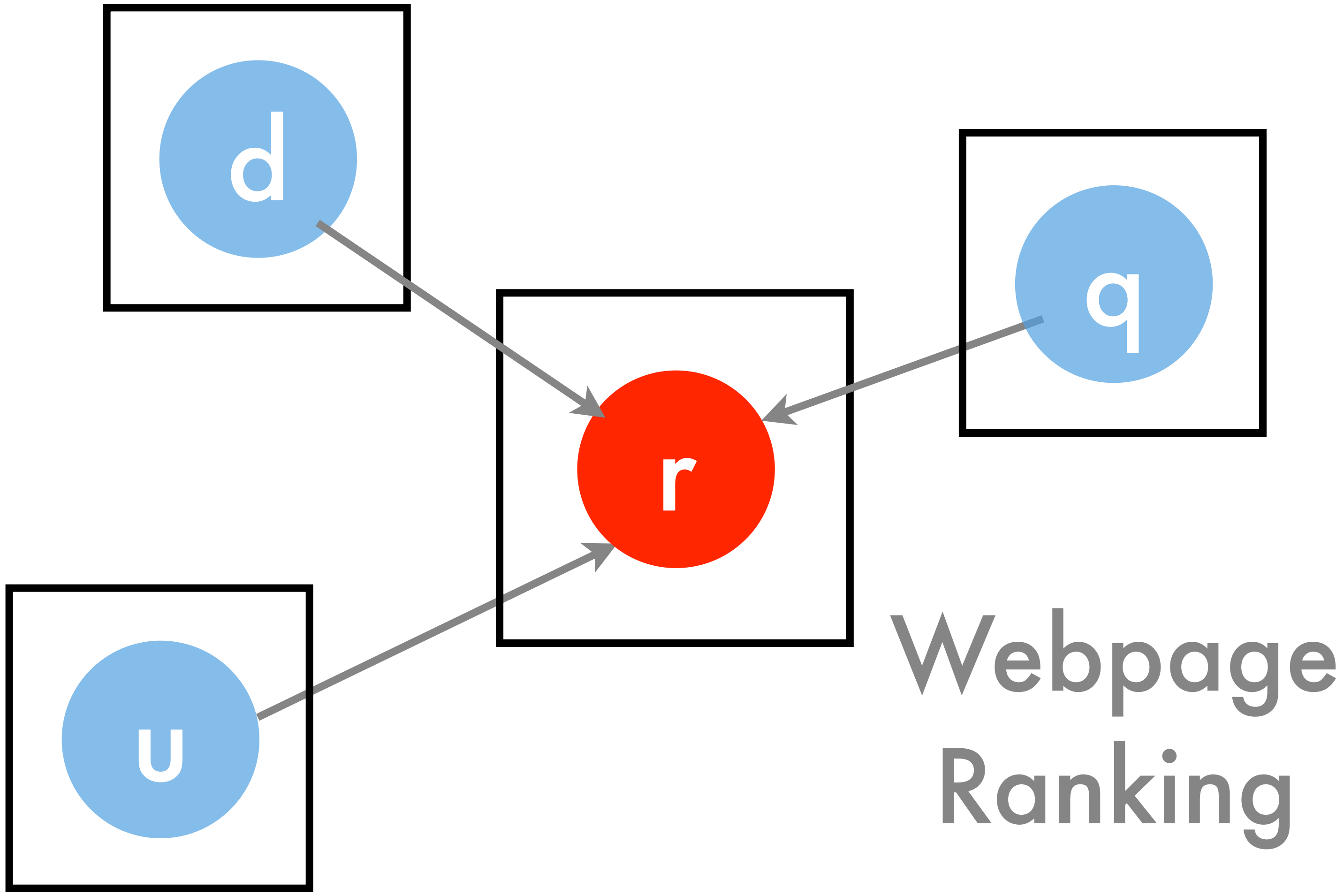
Webpage Ranking



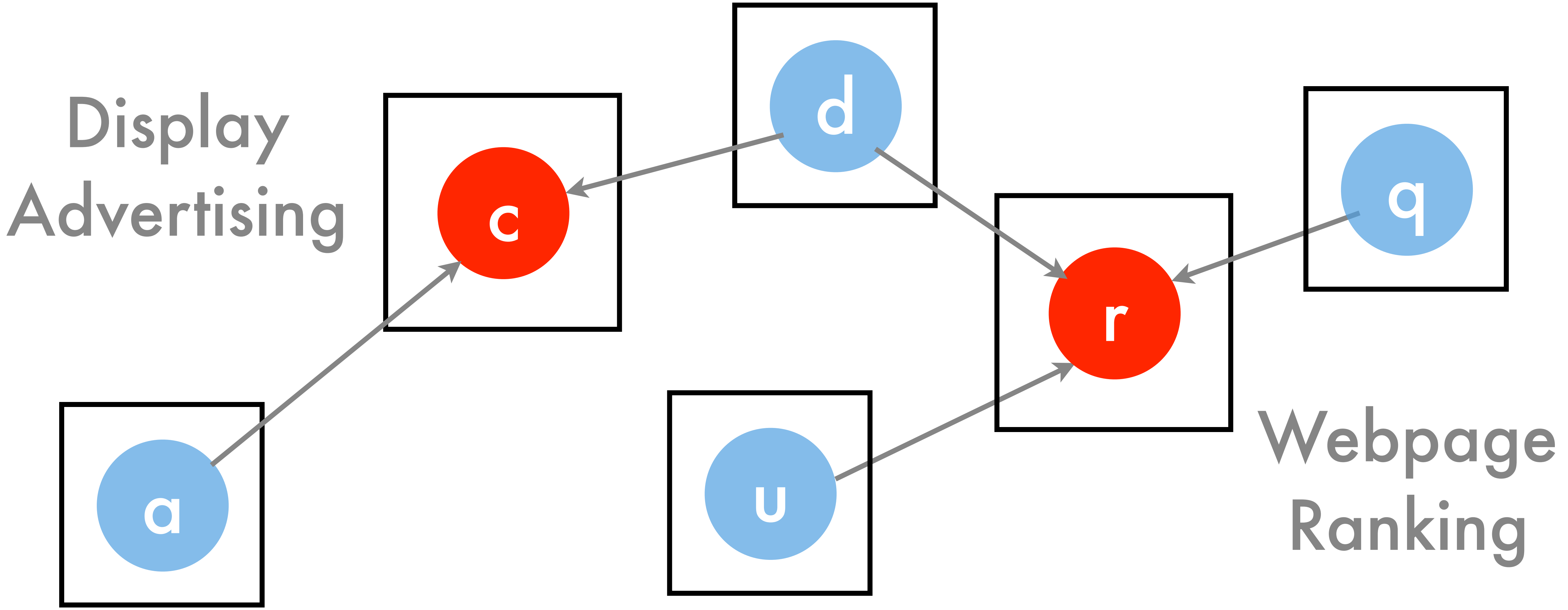
Collaborative Models



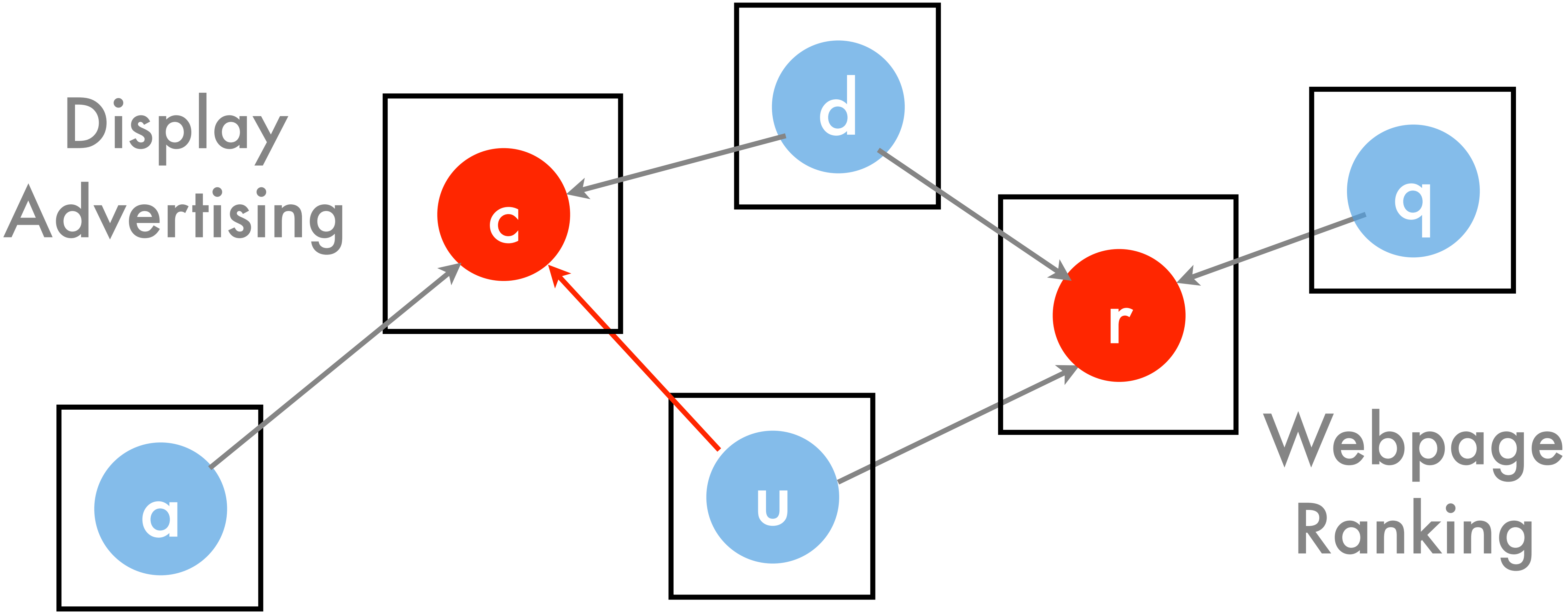
Data Integration



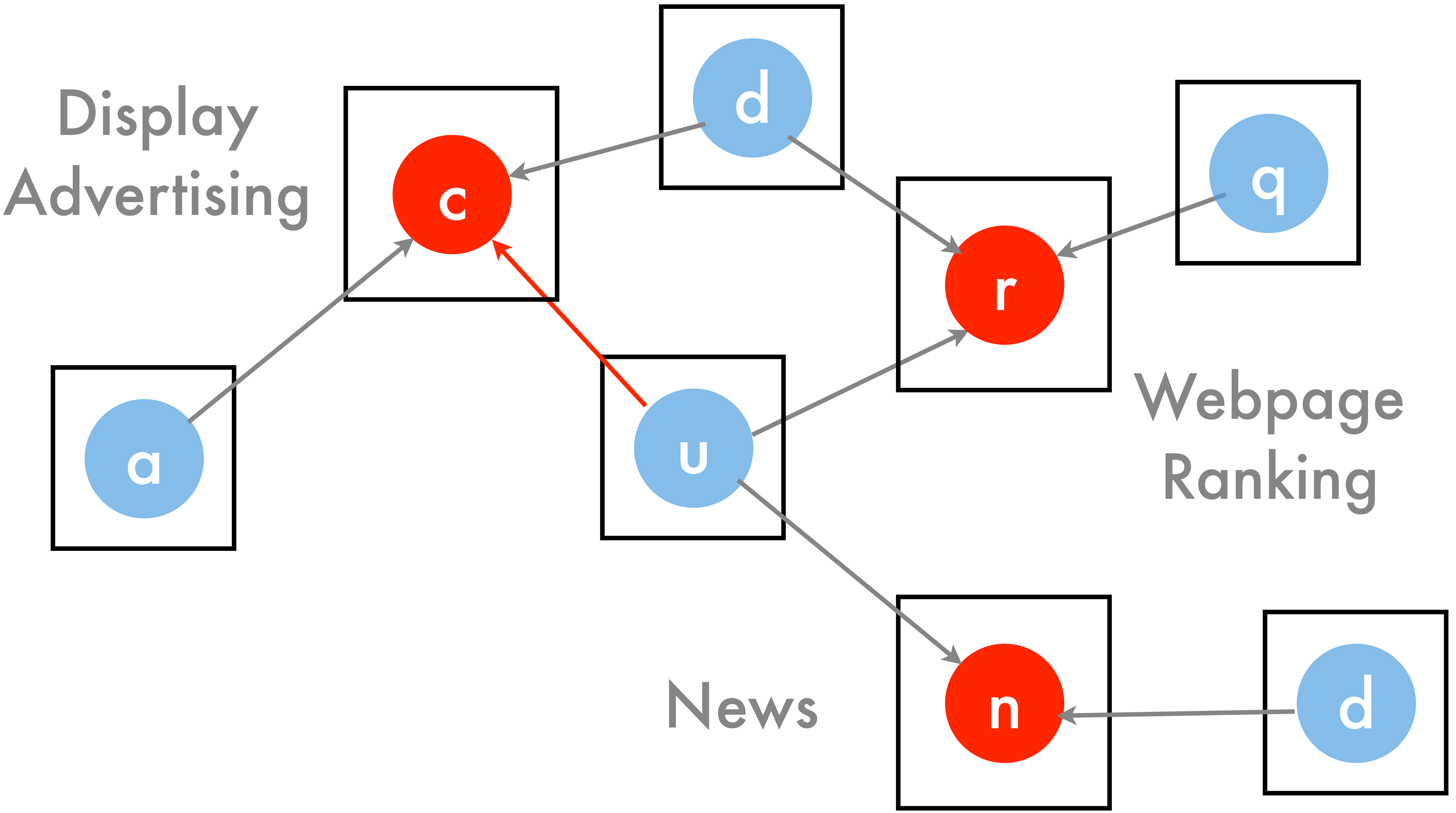
Data Integration



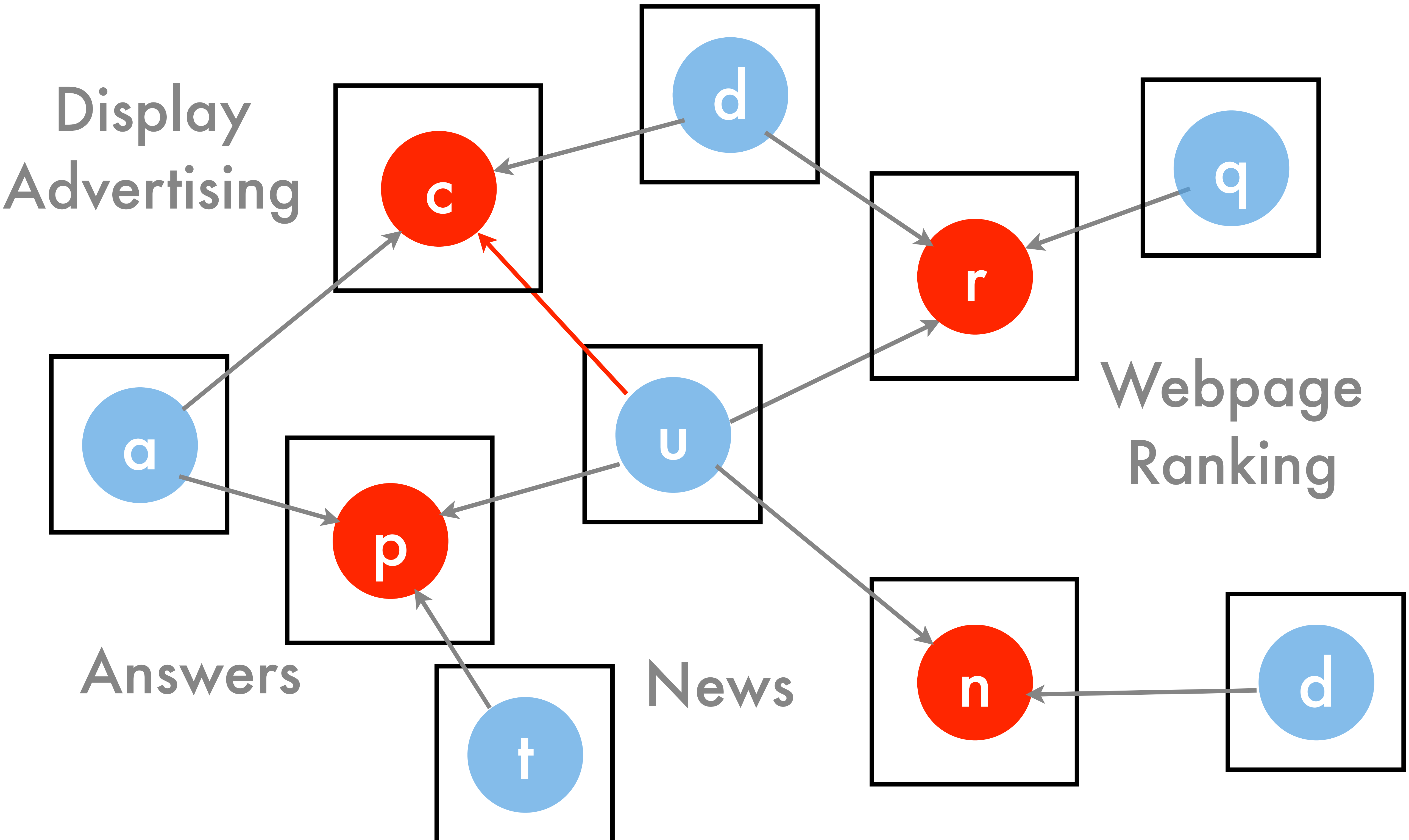
Data Integration



Data Integration



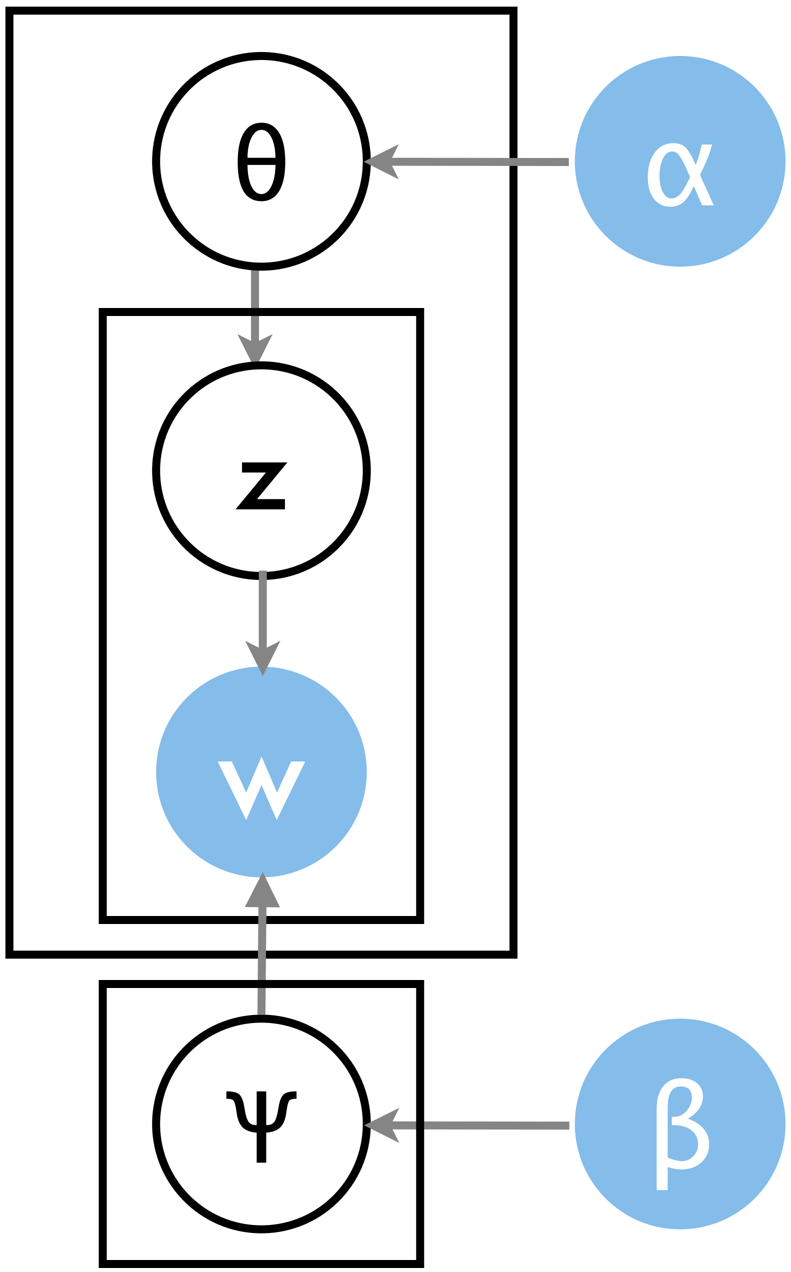
Data Integration



Topic Models

Topic Models

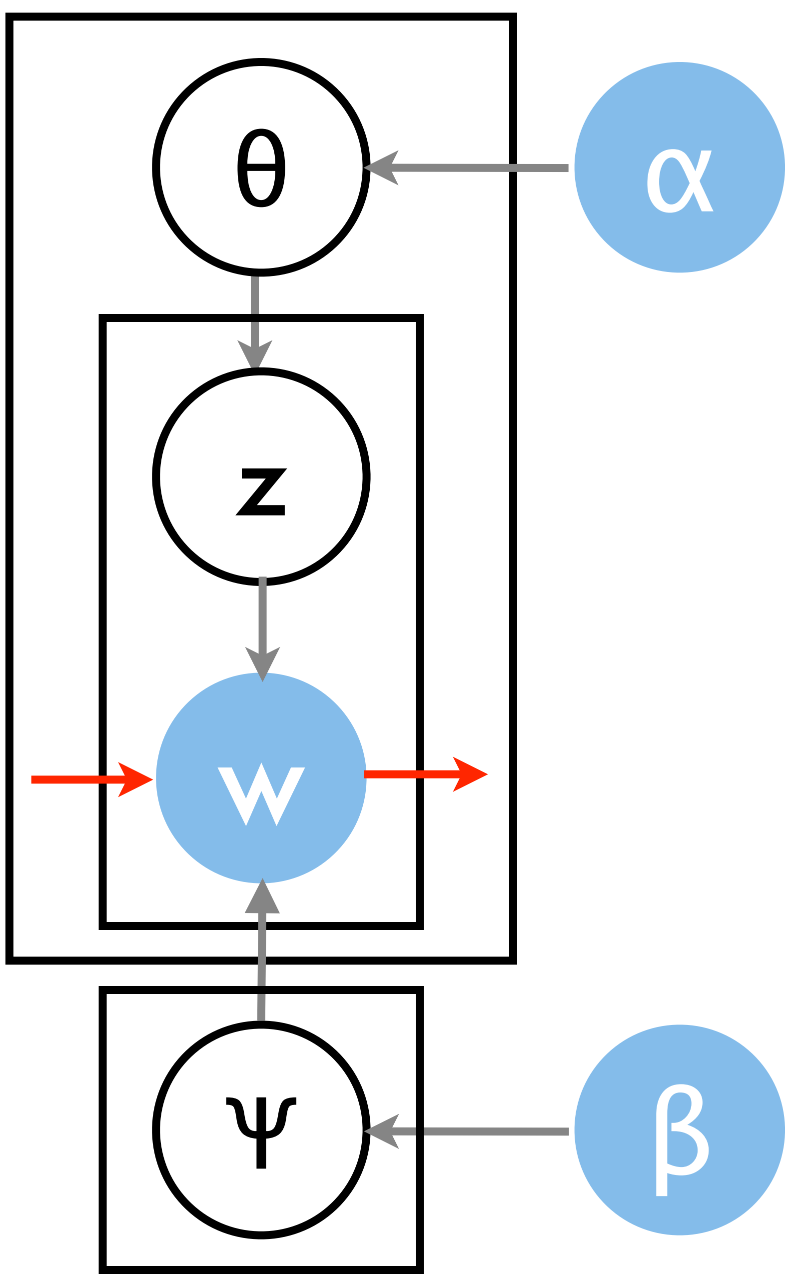
Topic Models



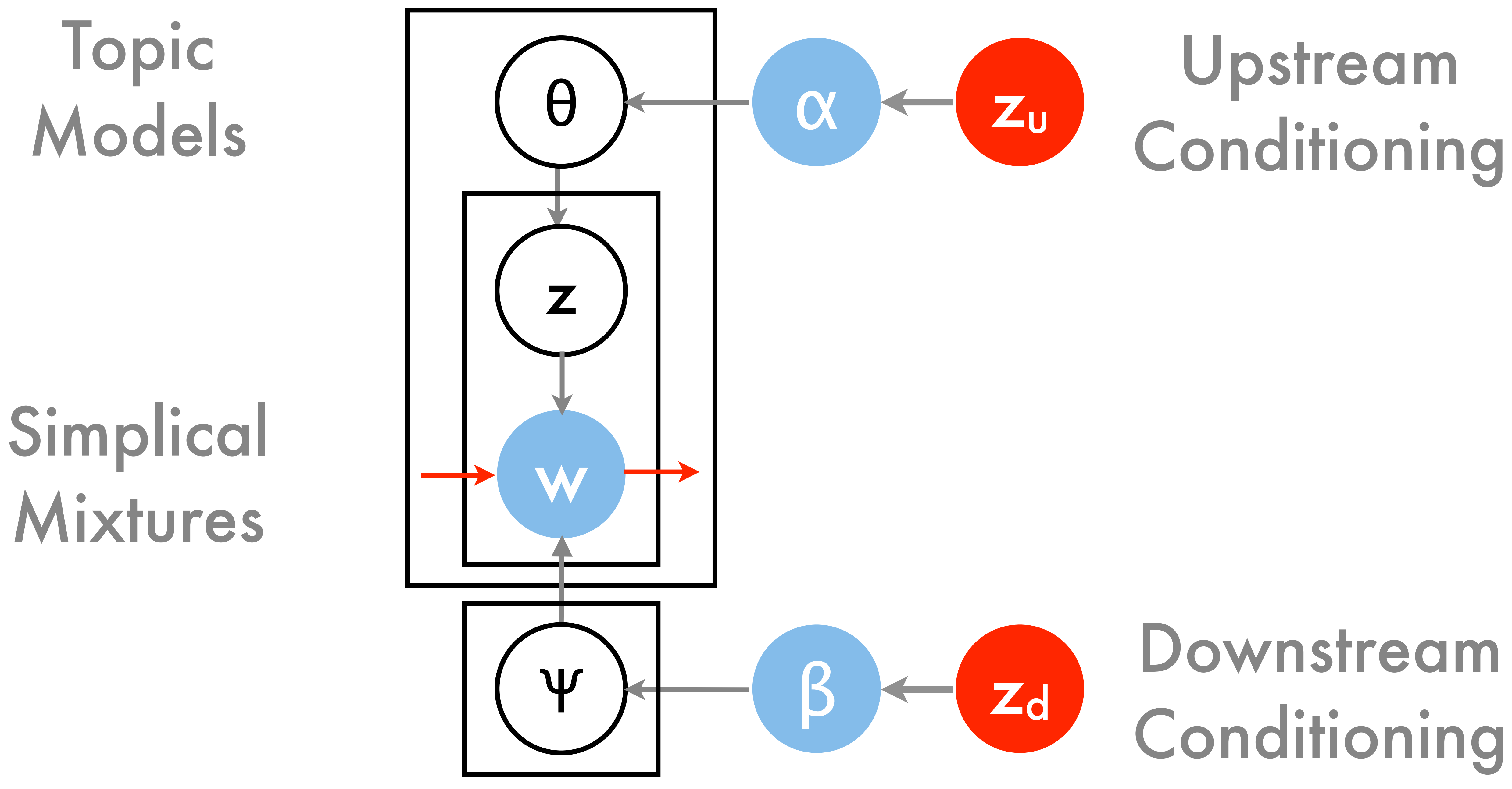
Topic Models

Topic Models

Simplical Mixtures



Topic Models

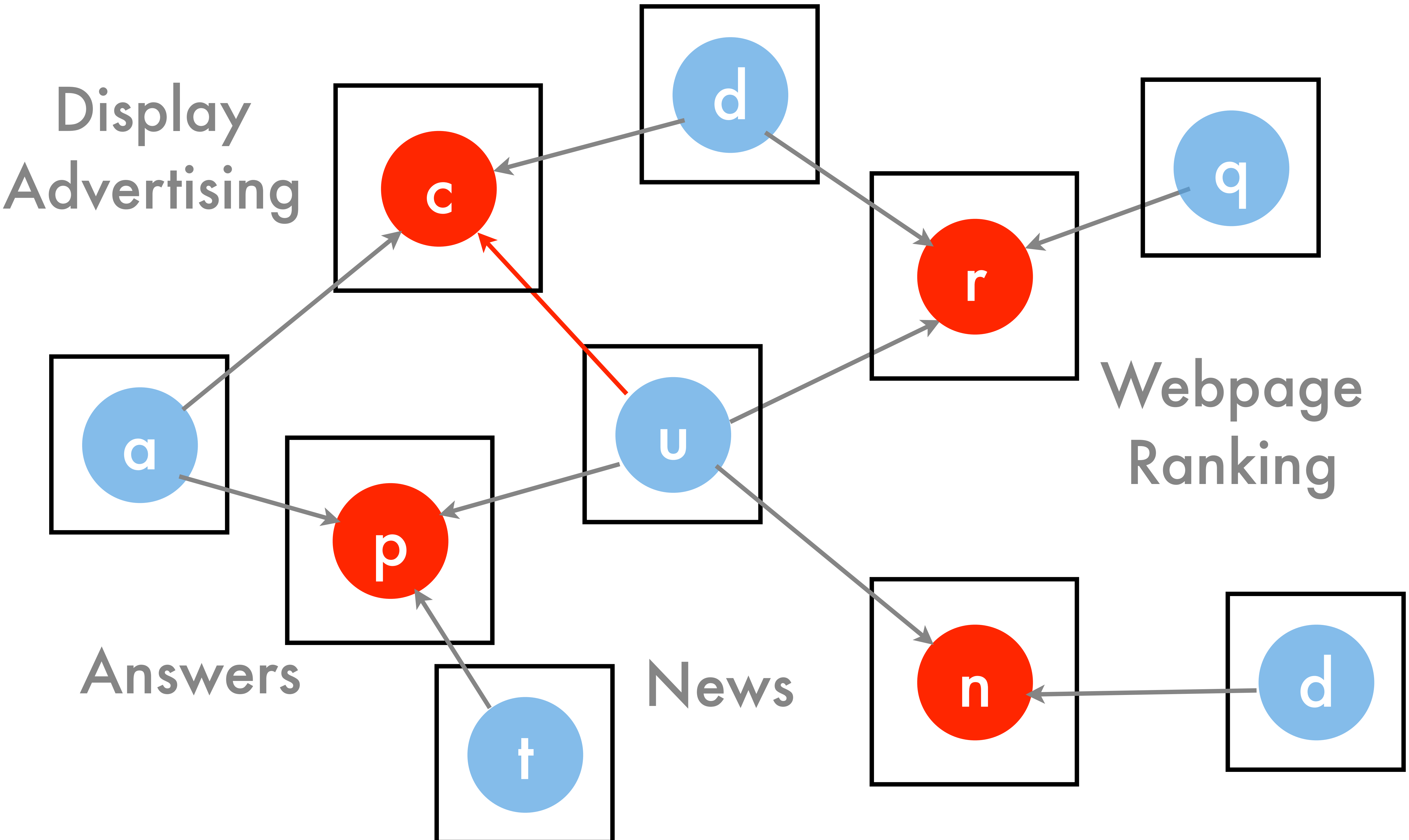


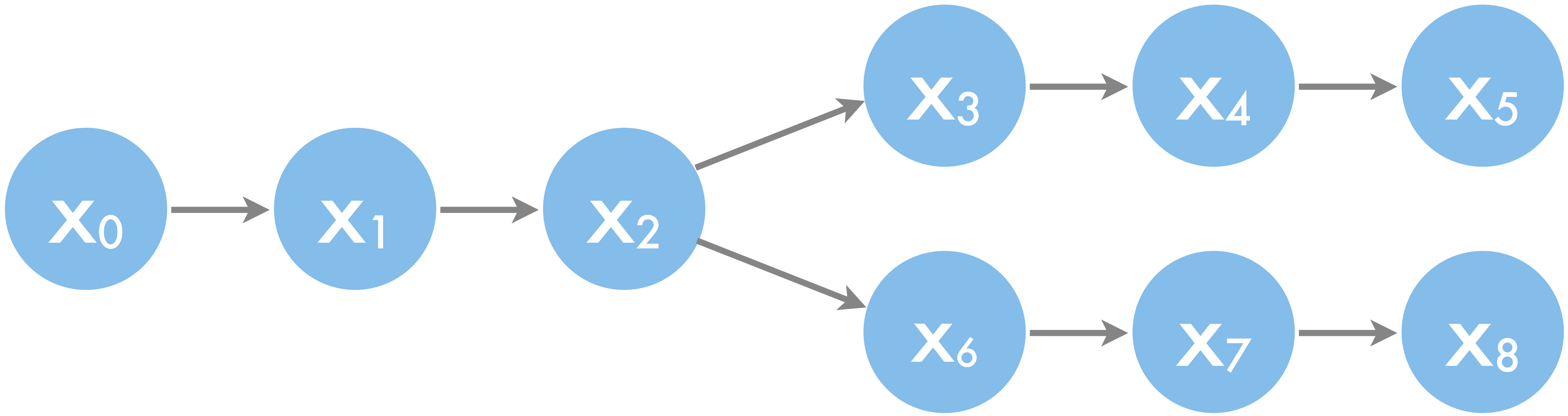
Undirected Graphical Models

Review

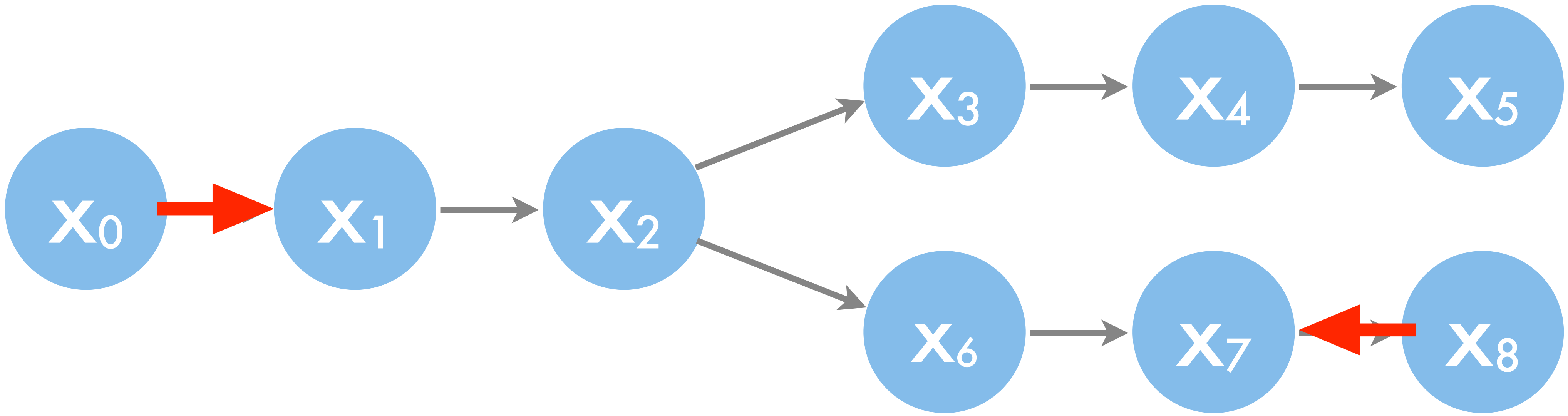

Spam Filtering
Classification
Exploration
Segmentation
Annotation
System Design
Prediction (time series)
Clustering
Novelty Detection
Document Understanding
Performance Tuning
User Modeling
Advertising
Debugging

Data Integration

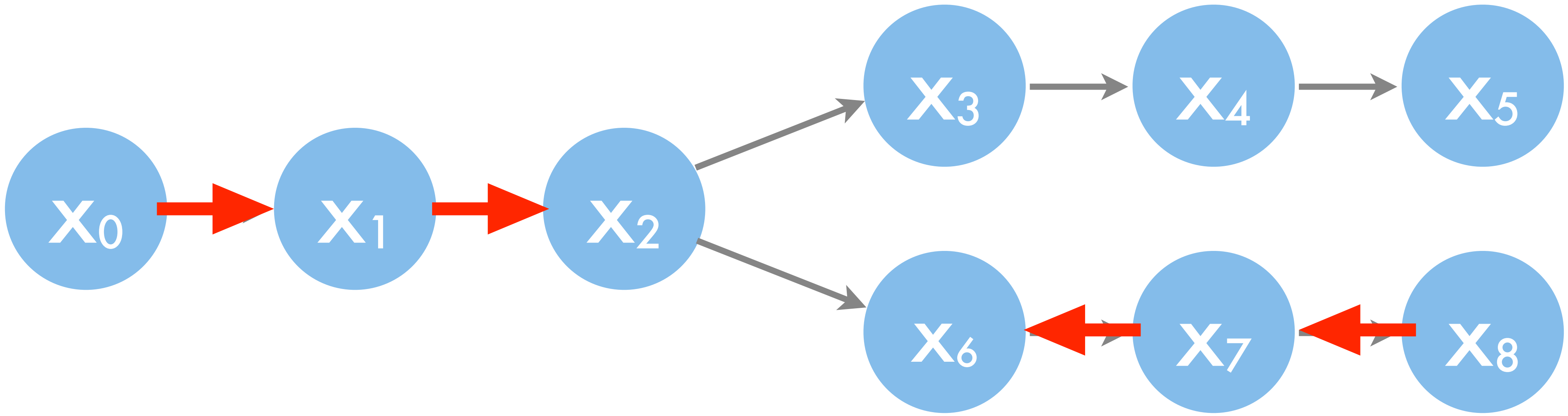




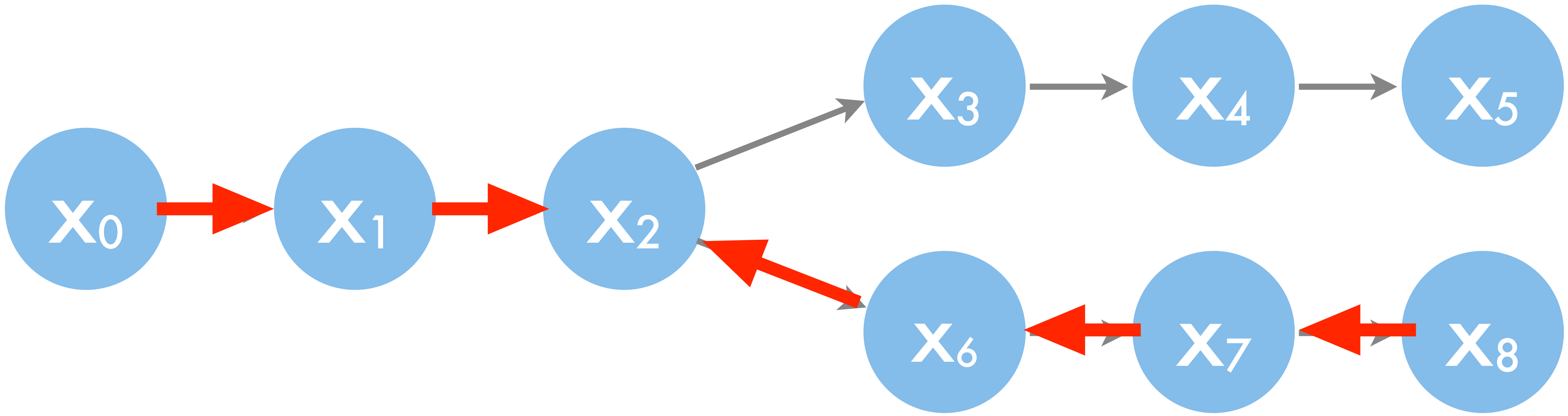
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether



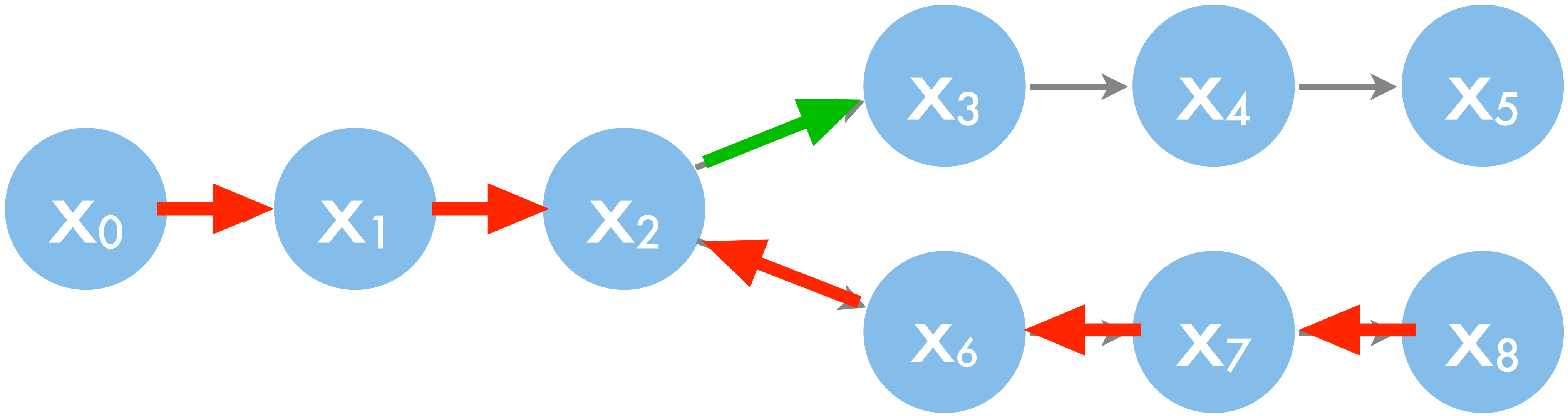
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether



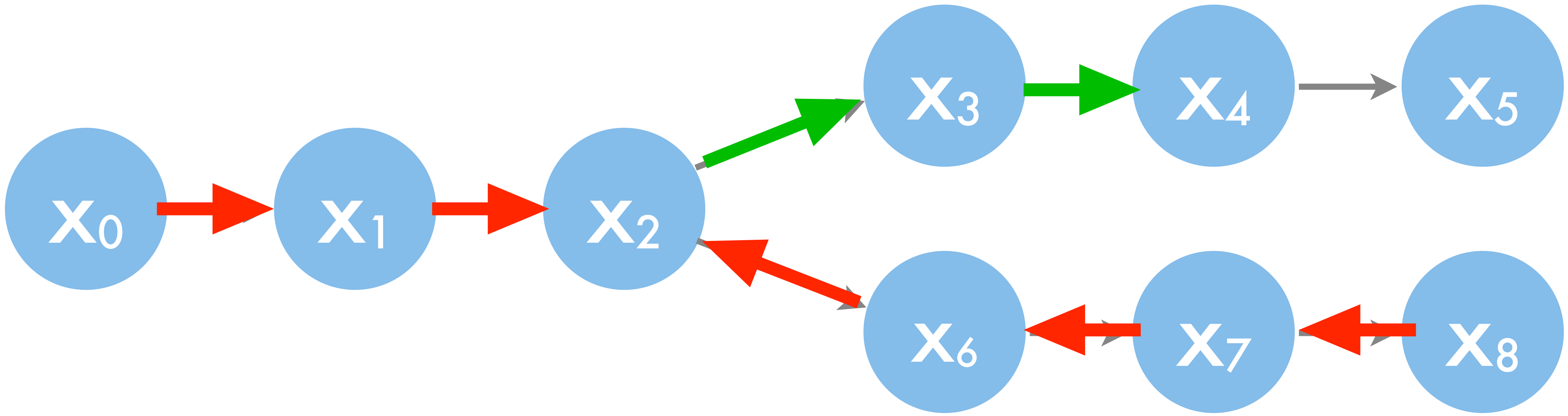
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether



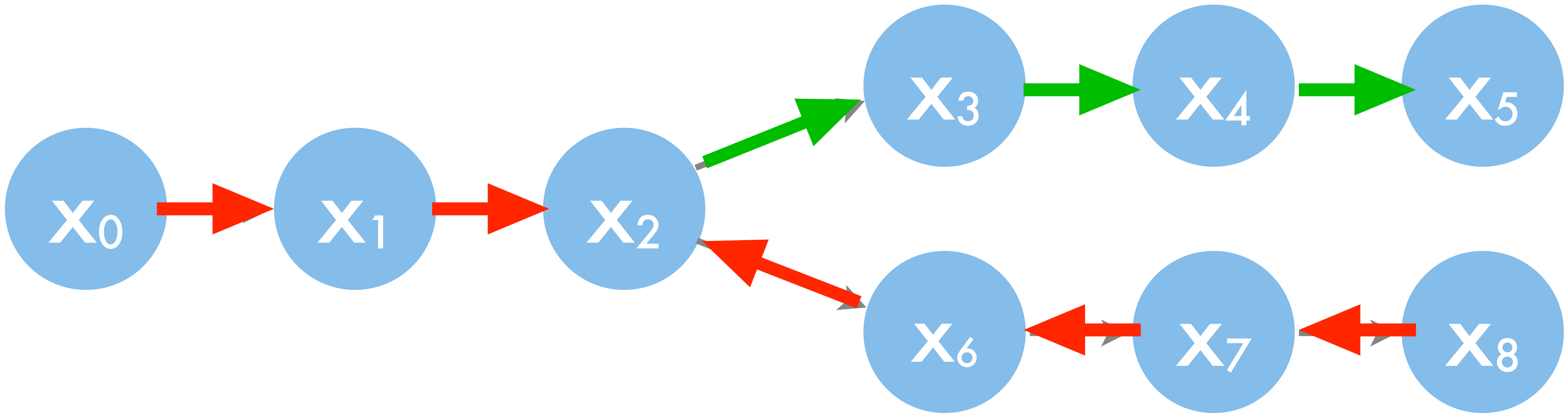
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether



- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether



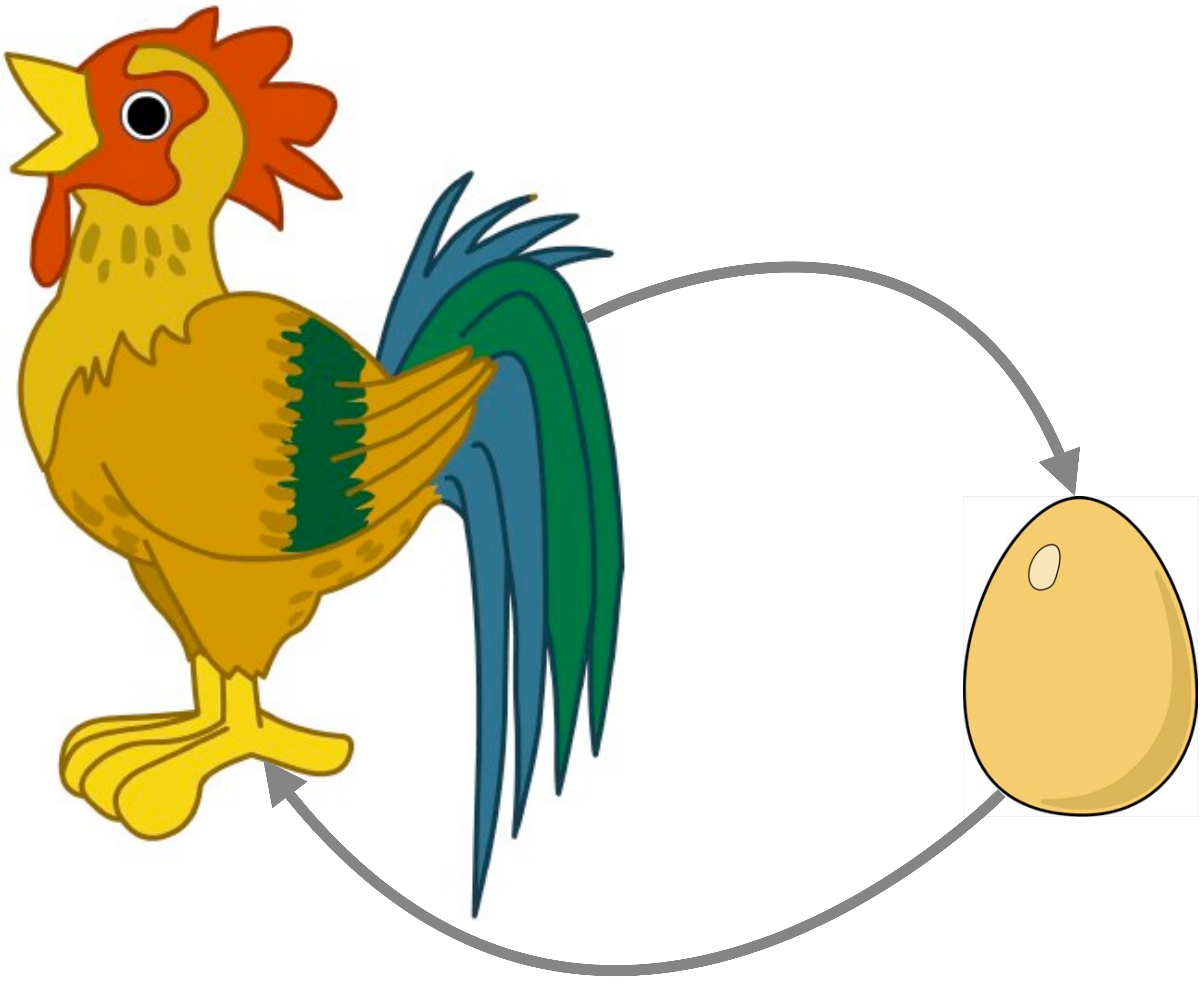
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether



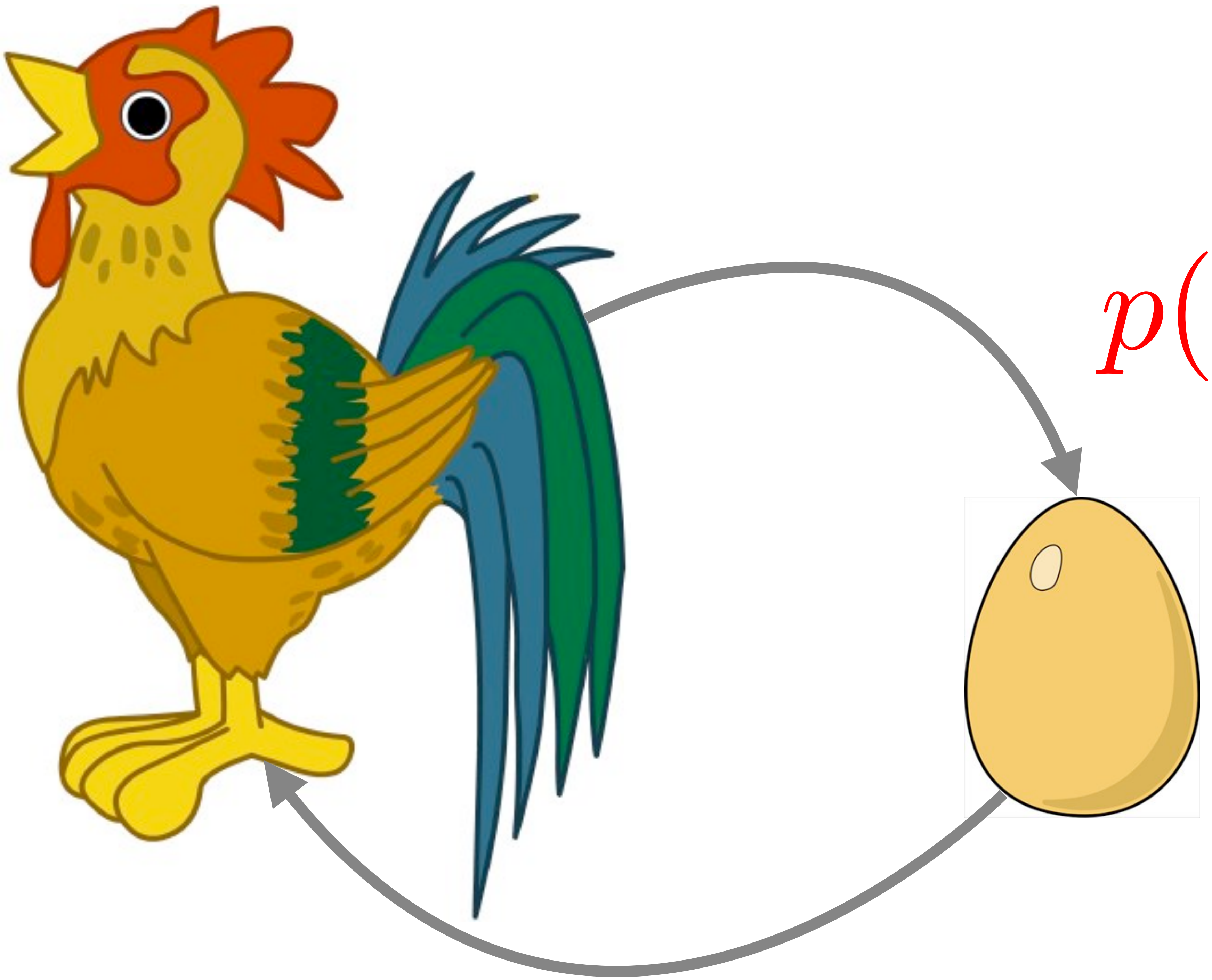
- Forward/Backward messages as normal for chain
- When we have more edges for a vertex use
  - For each outgoing message, send it once you have all other incoming messages
  - PRINCIPLED HACK
    If no message received yet, set it to 1 altogether

Blunting the arrows …

Chicken and Egg

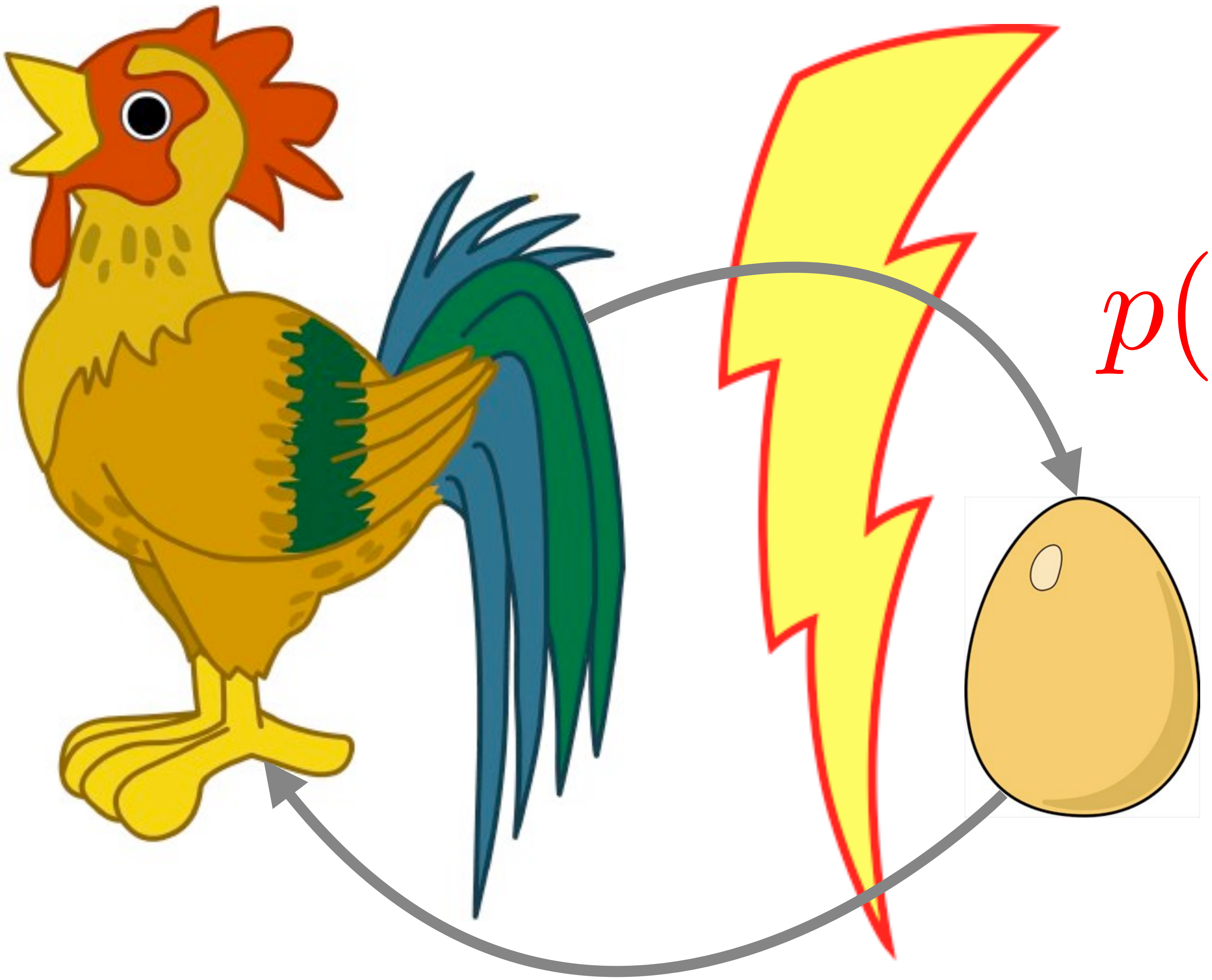


Chicken and Egg



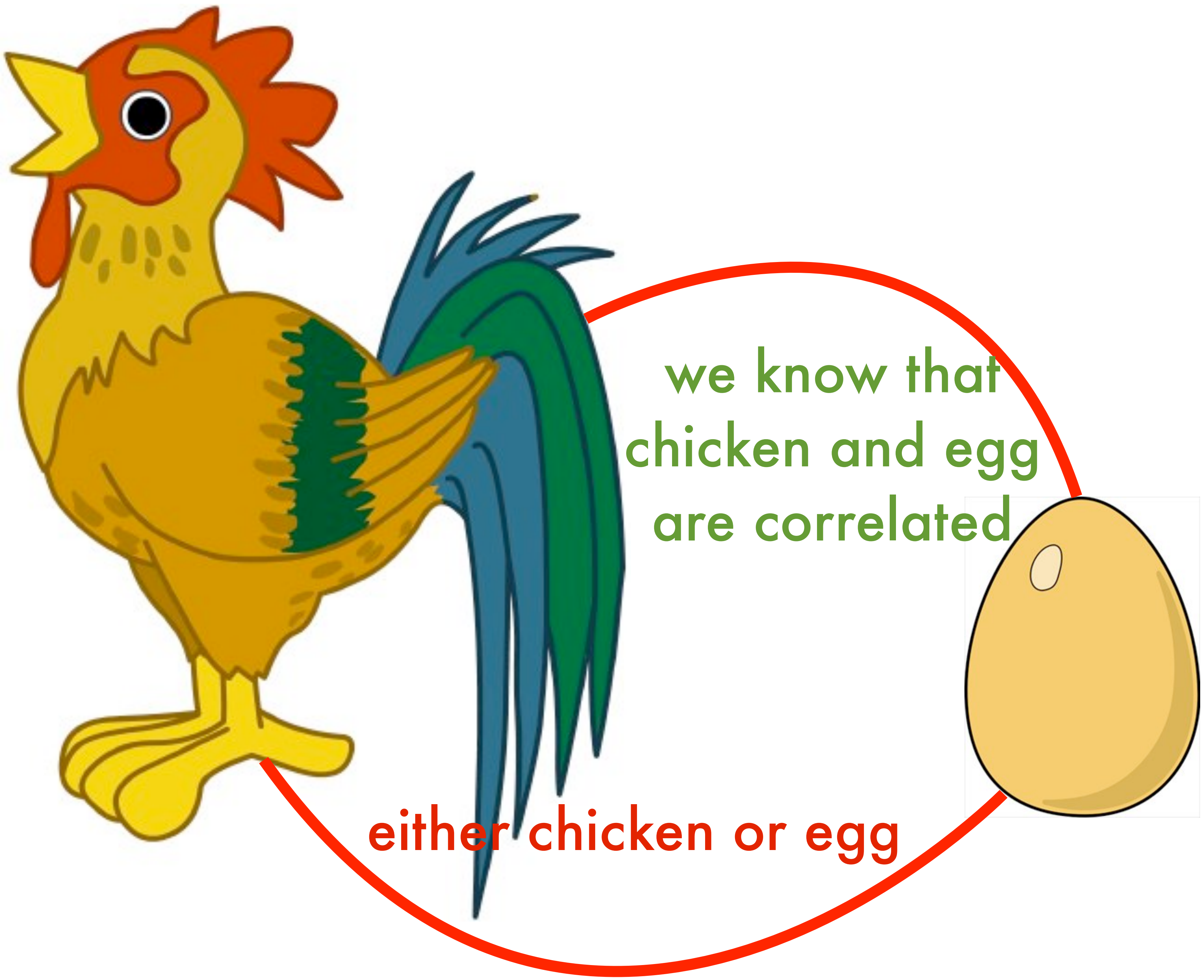
$$p(c|e)p(e|c)$$

Chicken and Egg

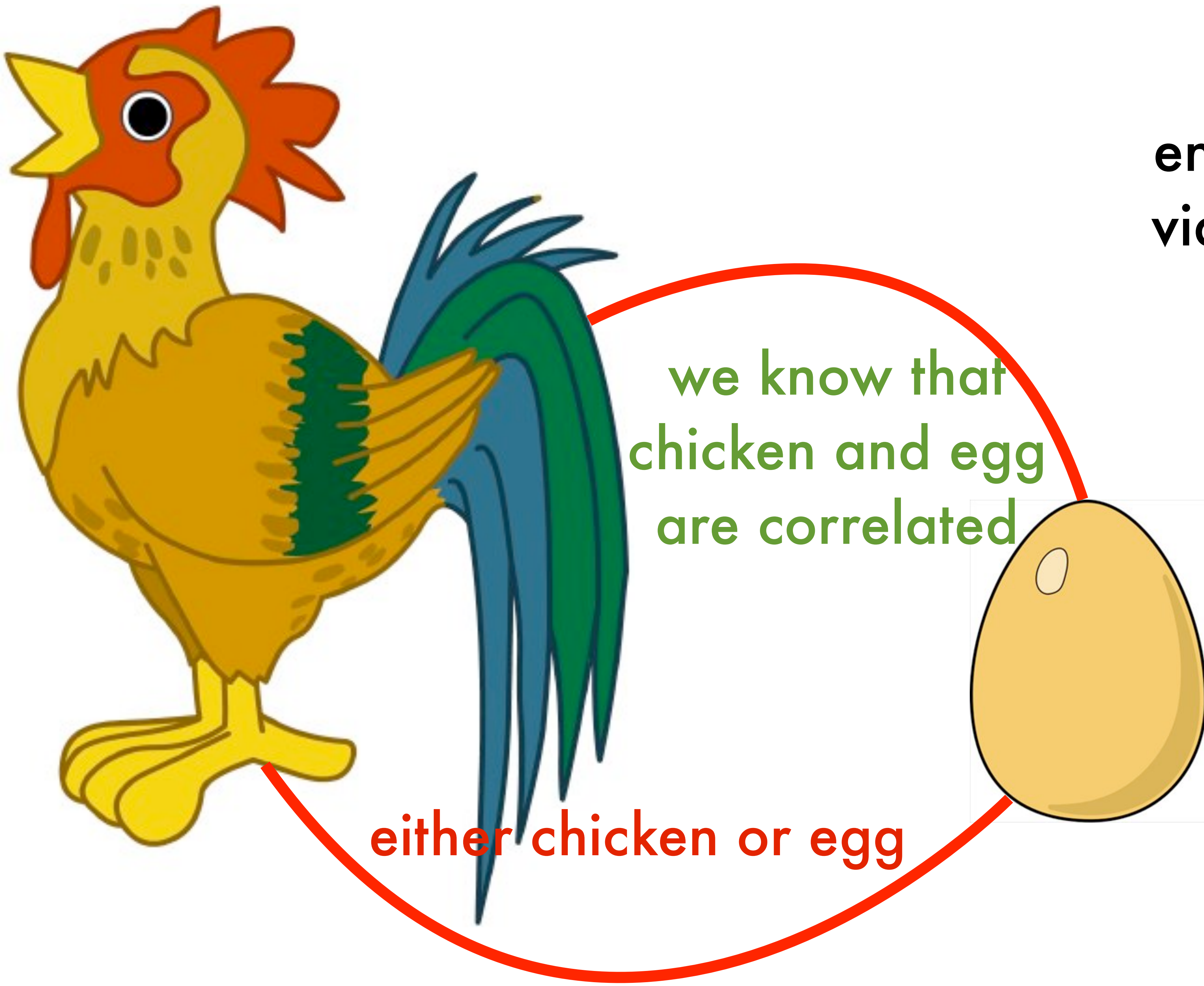


$$p(c|e)p(e|c)$$

Chicken and Egg



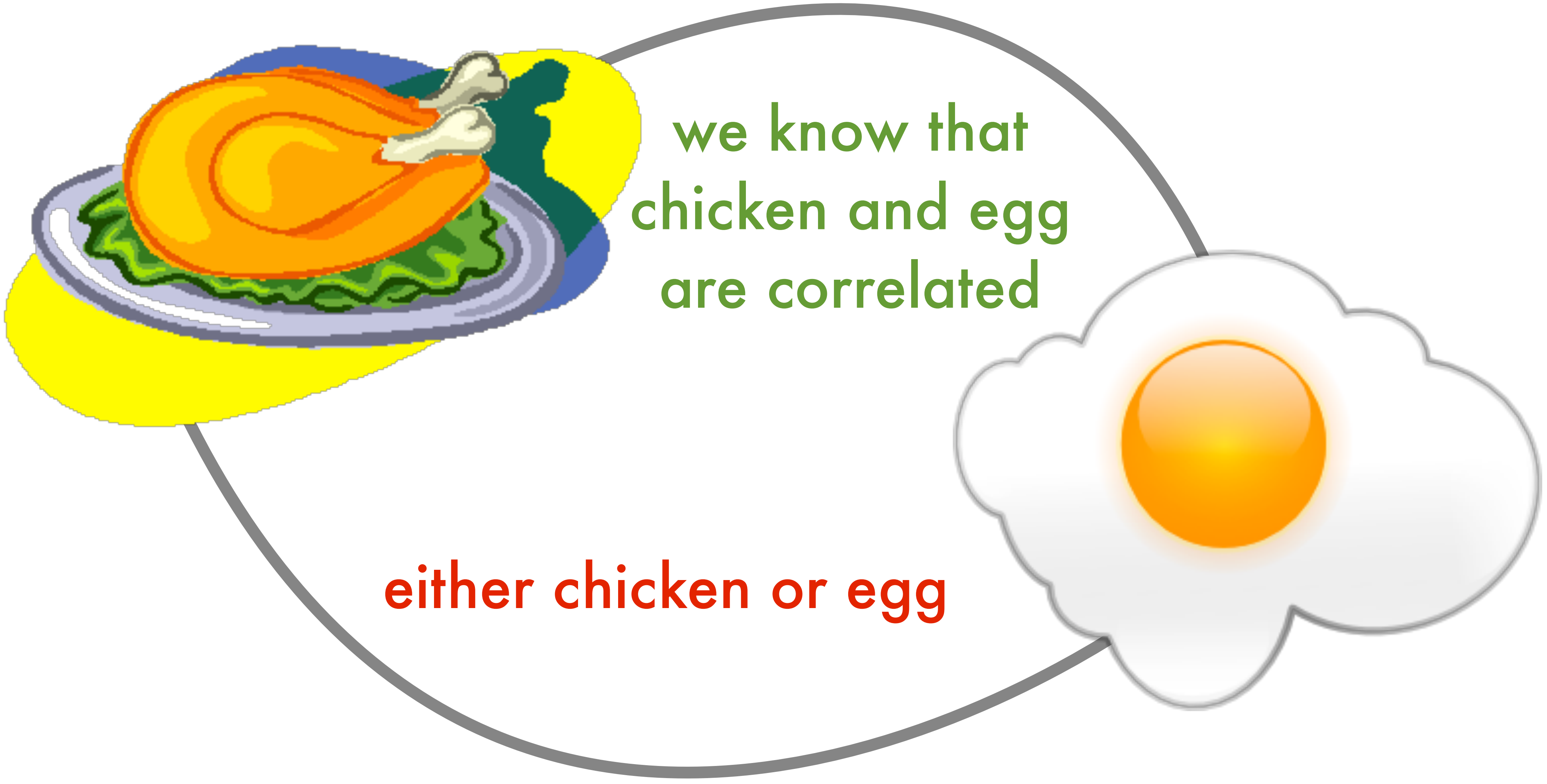
Chicken and Egg



encode the correlation via the clique potential between c and e

$$p(c, e) \propto \exp \psi(c, e)$$

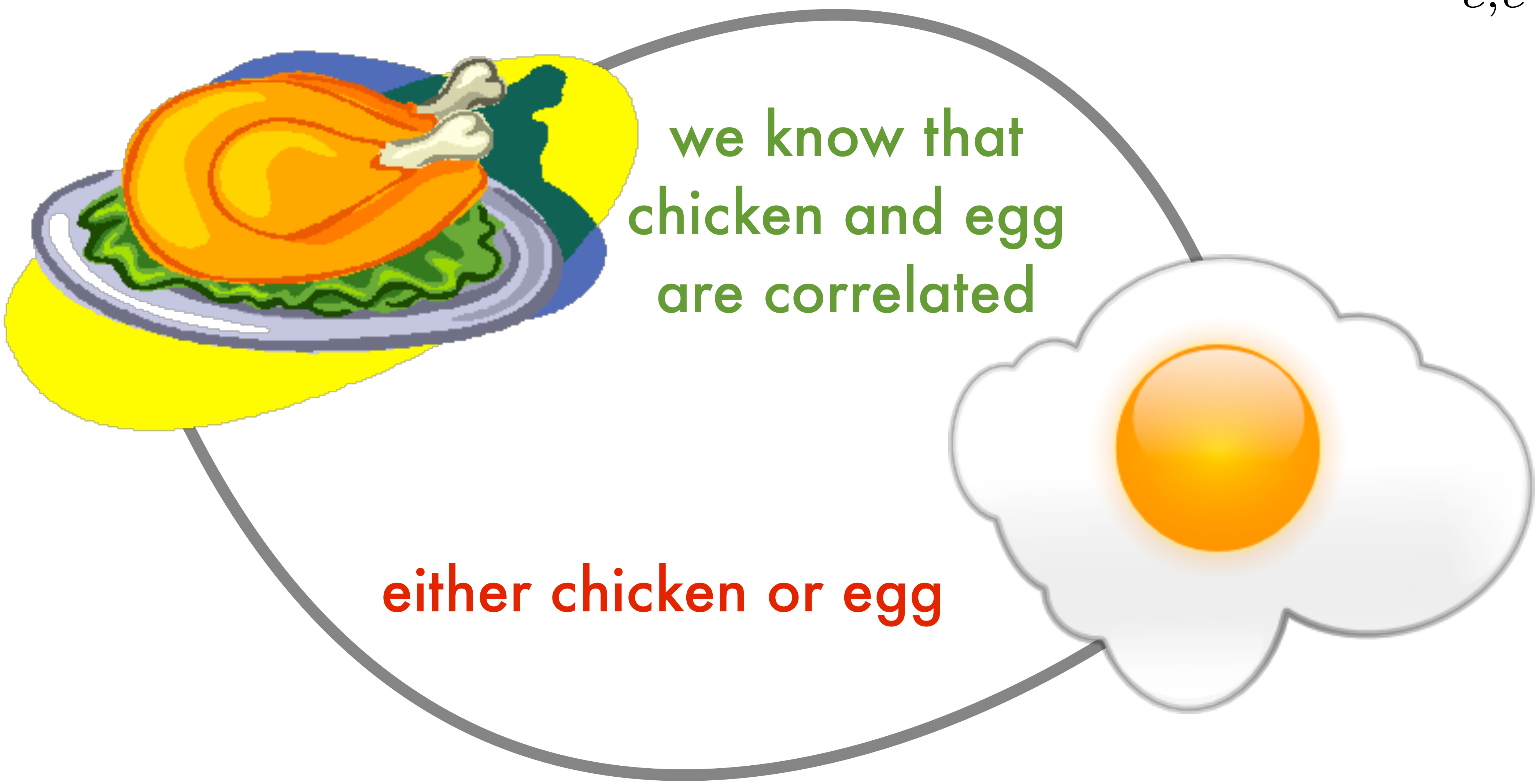
Chicken and Egg



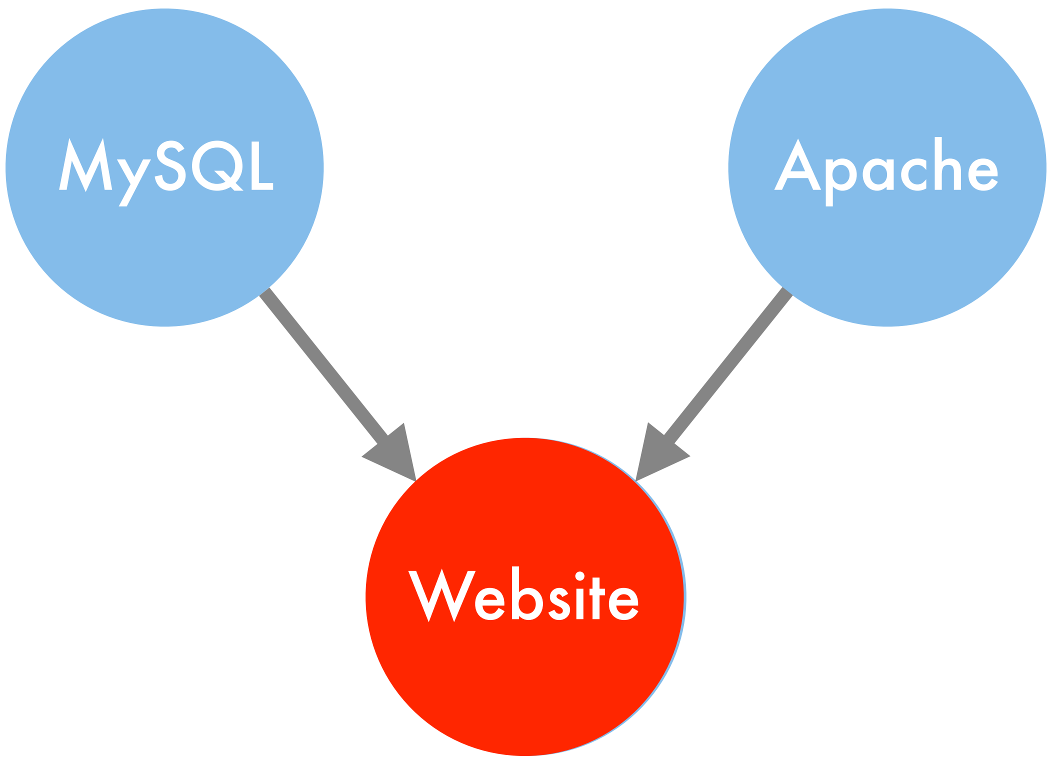
Chicken and Egg

$$p(c,e) = \frac{\exp \psi(c,e)}{\sum_{c',e'} \exp \psi(c',e')}$$

$$= \exp\left[\psi(c,e) - g(\psi)\right] \text{ where } g(\psi) = \log \sum_{c,e} \exp \psi(c,e)$$

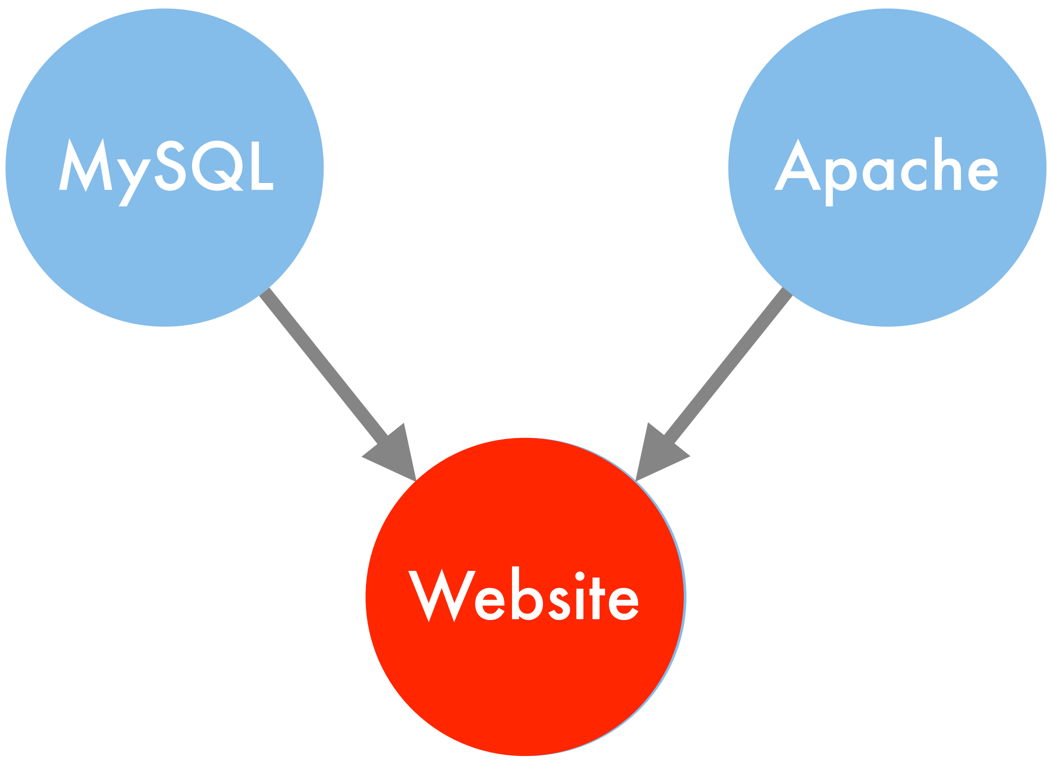


... some Yahoo service

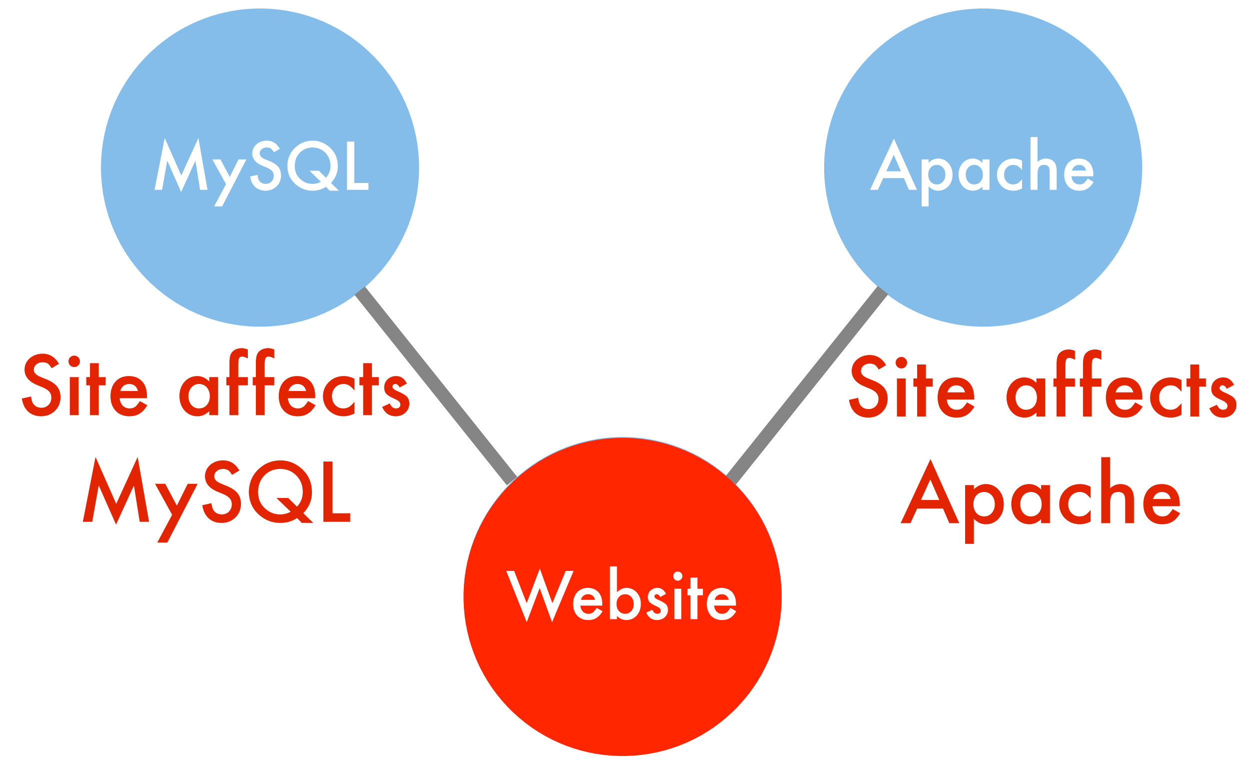


$$p(w|m, a)p(m)p(a)$$
$$m \not\perp\!\!\!\perp a | w$$

… some Yahoo service

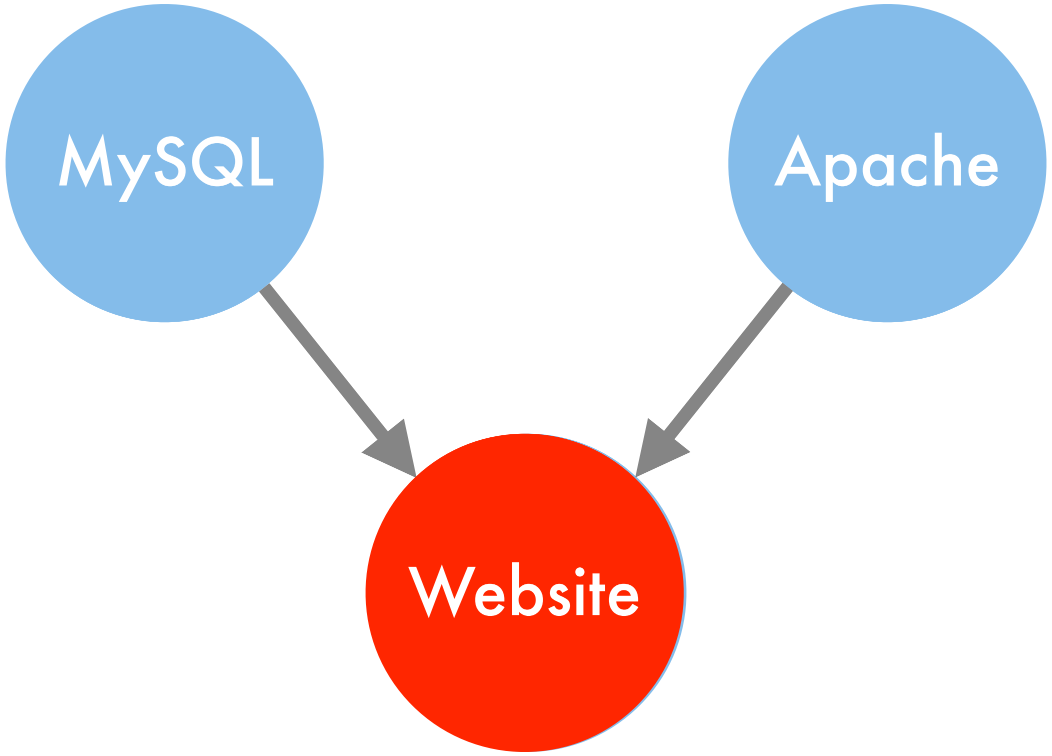


$$p(w|m,a)p(m)p(a)$$
$$m \not\perp\!\!\!\perp a|w$$



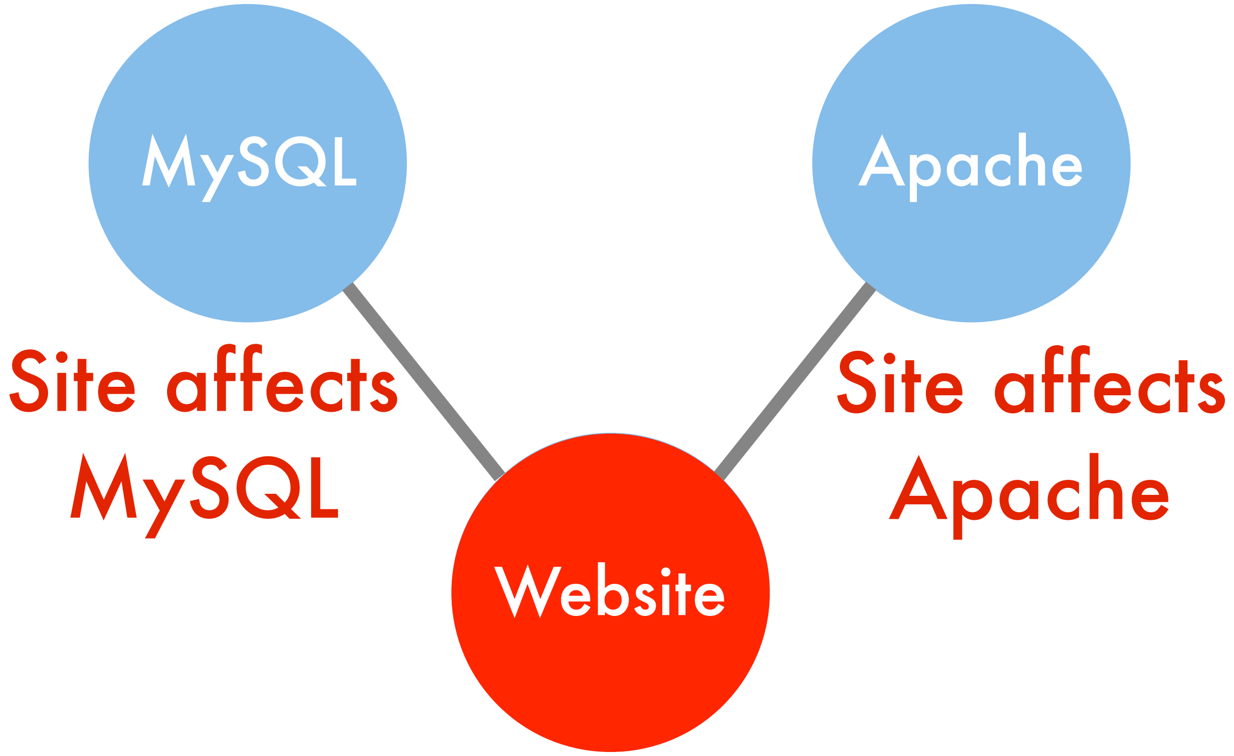
$$p(m,w,a) \propto \phi(m,w)\phi(w,a)$$
$$m \perp\!\!\!\perp a|w$$

… some Yahoo service



$$p(w|m,a)p(m)p(a)$$
$$m \not\perp\!\!\!\perp a|w$$

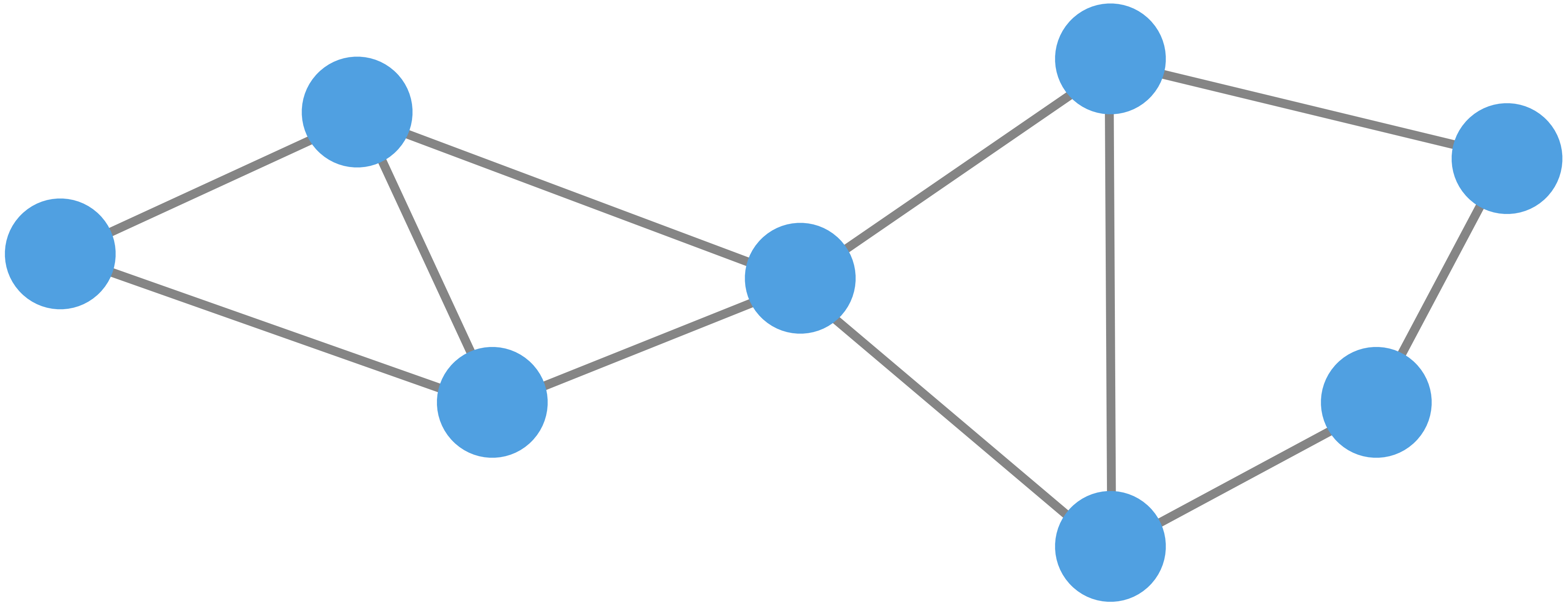
easier
"debugging"



$$p(m,w,a) \propto \phi(m,w)\phi(w,a)$$
$$m \perp\!\!\!\perp a|w$$

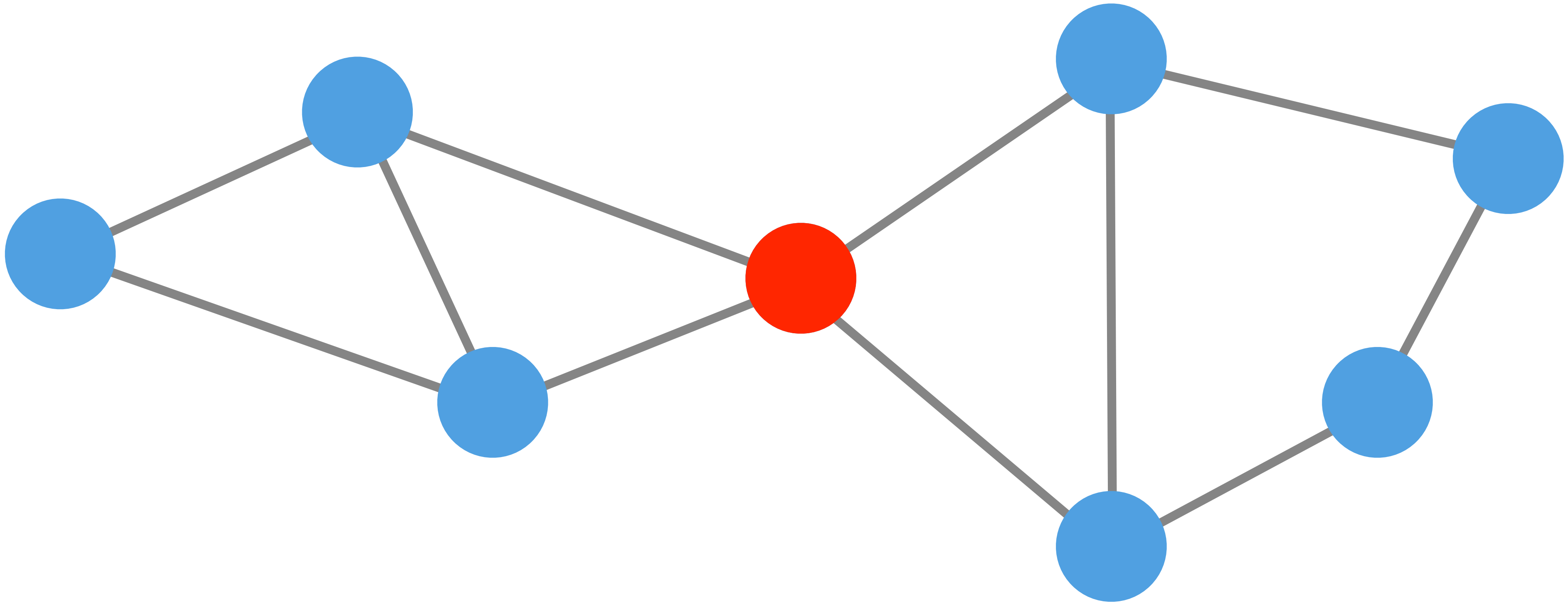
easier
"modeling"

Undirected Graphical Models



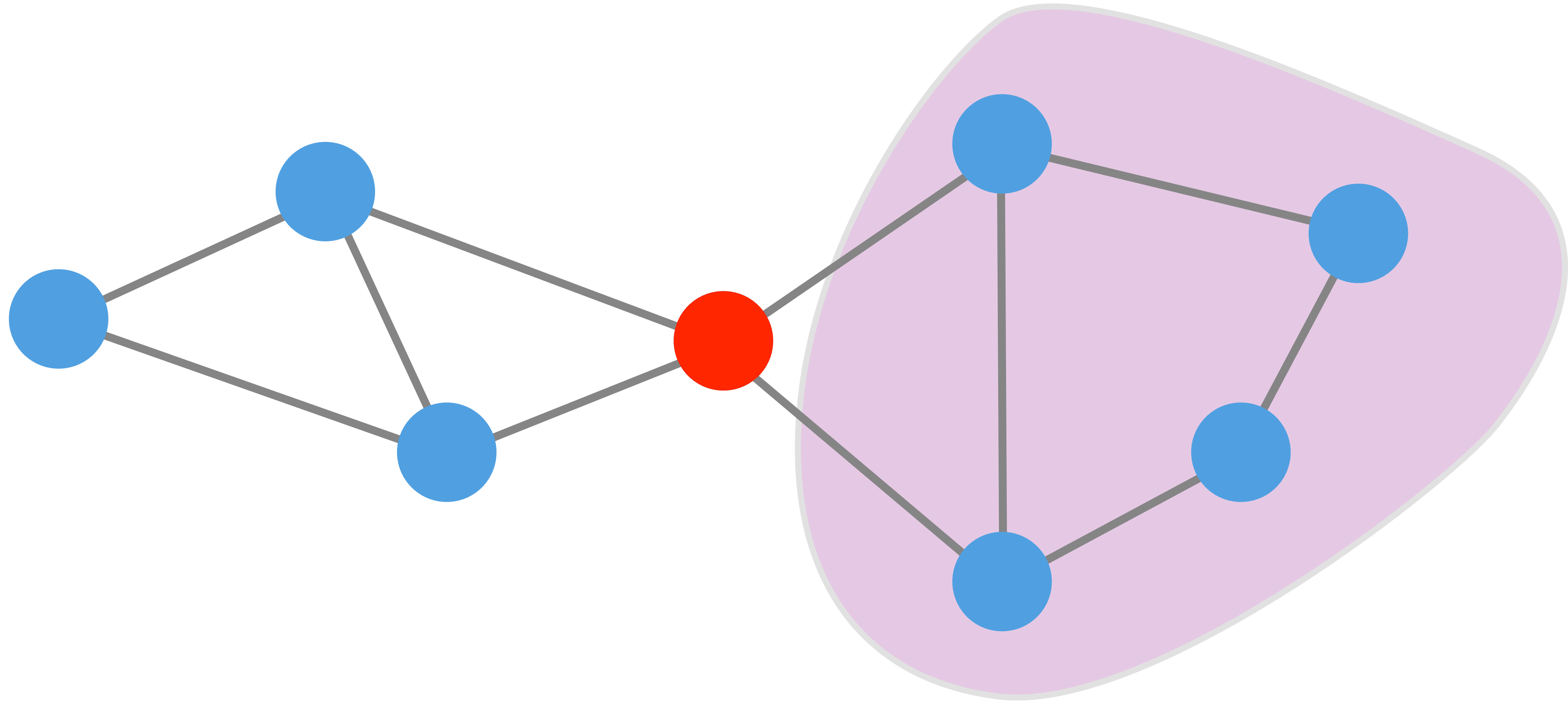
Key Concept
Observing nodes makes remainder conditionally independent

Undirected Graphical Models



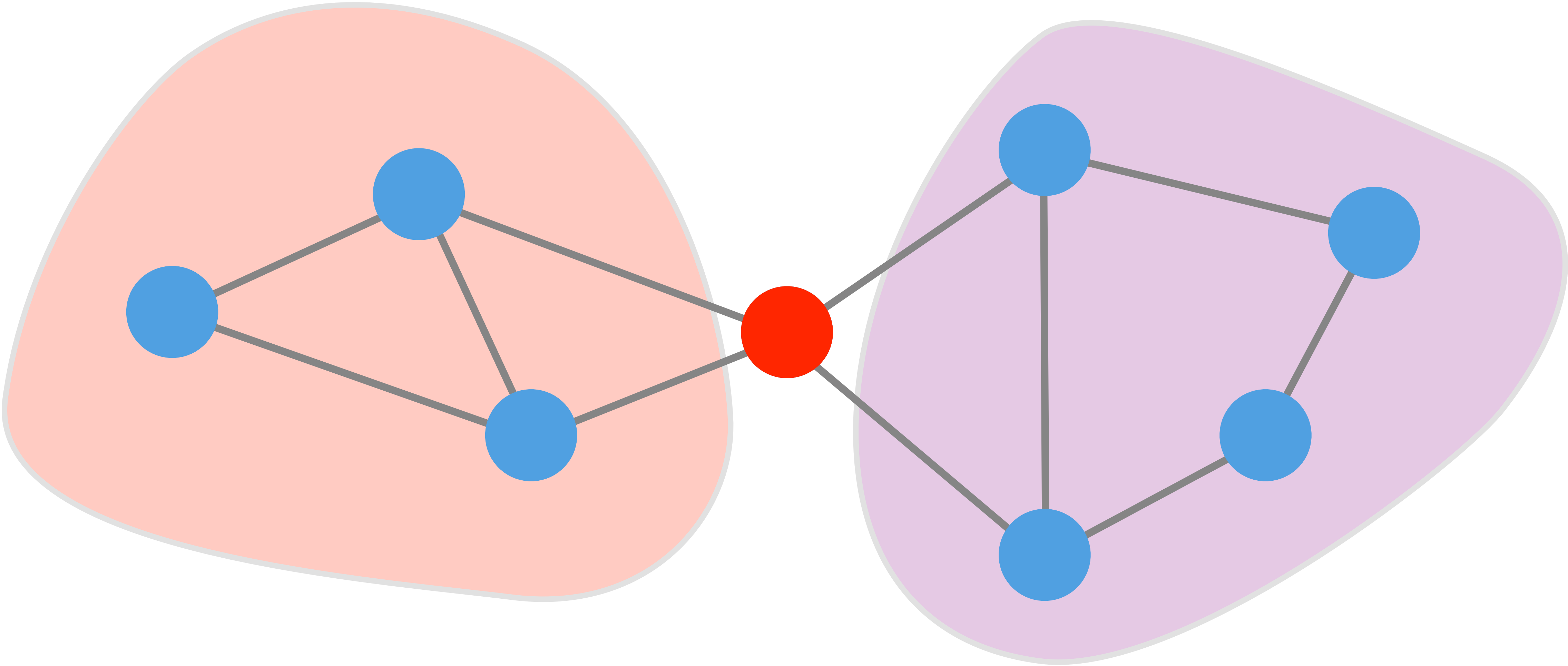
Key Concept
Observing nodes makes remainder
conditionally independent

Undirected Graphical Models



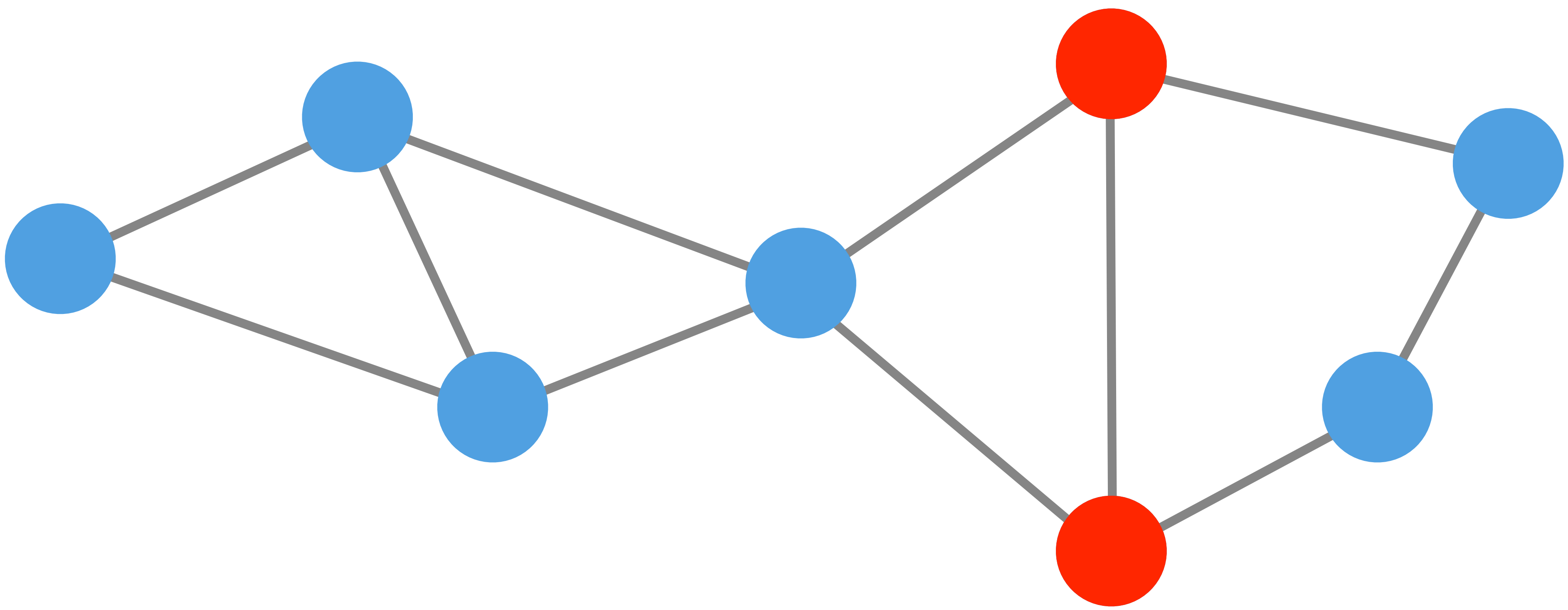
Key Concept
Observing nodes makes remainder conditionally independent

Undirected Graphical Models



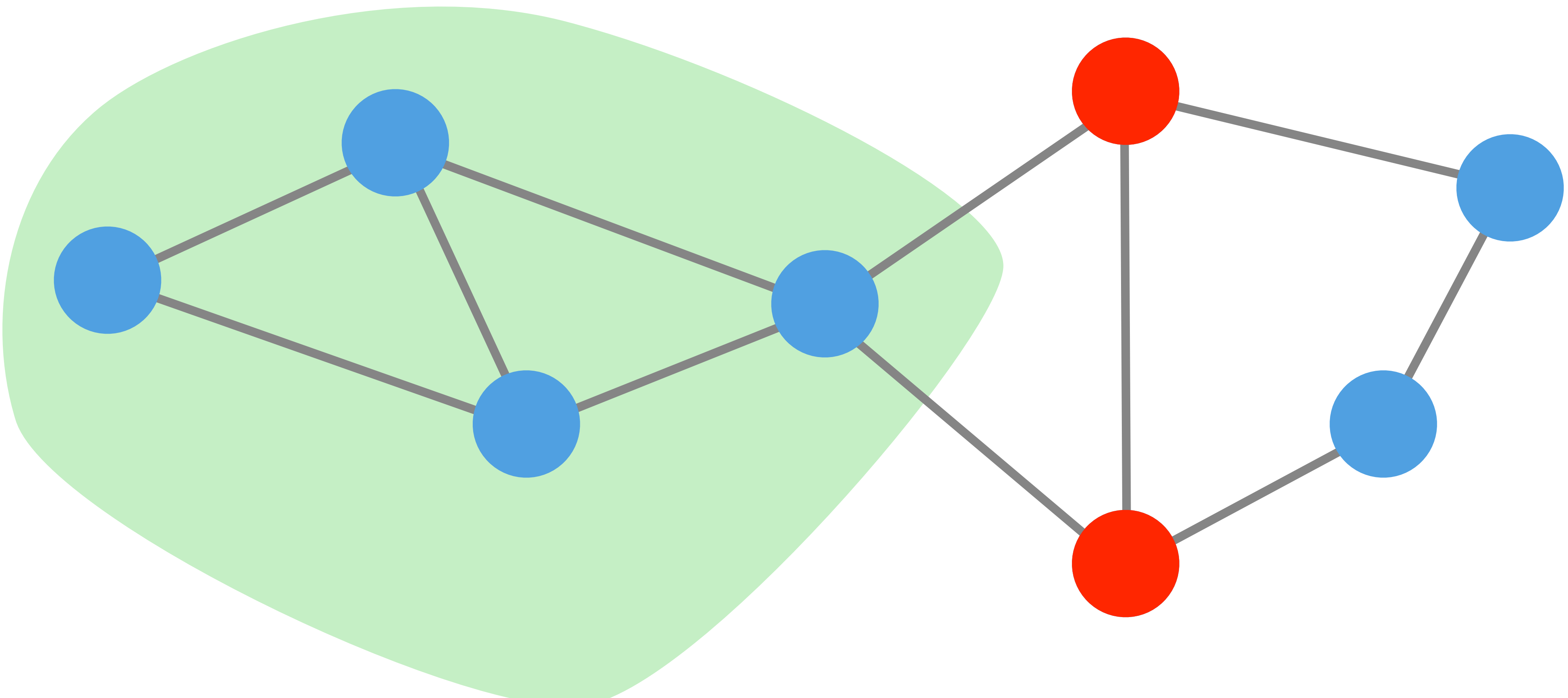
Key Concept
Observing nodes makes remainder conditionally independent

Undirected Graphical Models



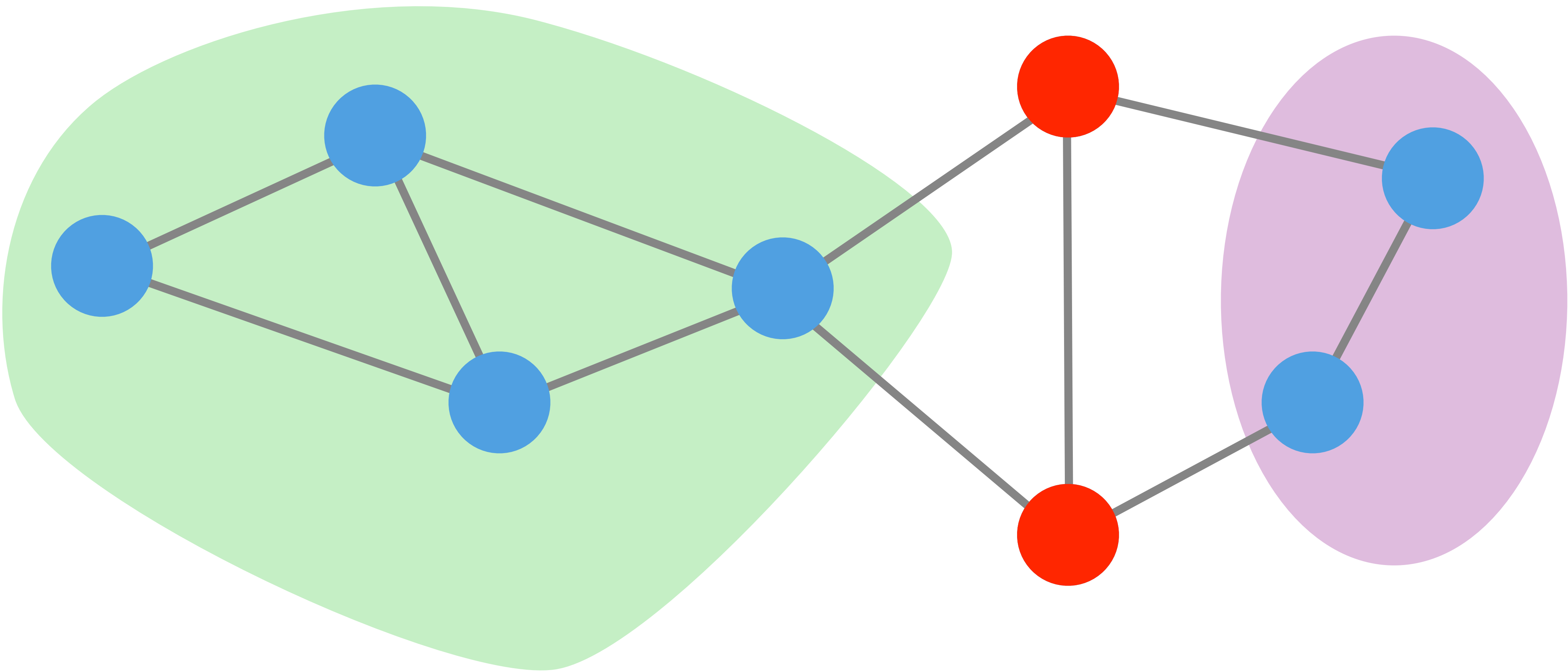
Key Concept
Observing nodes makes remainder
conditionally independent

Undirected Graphical Models



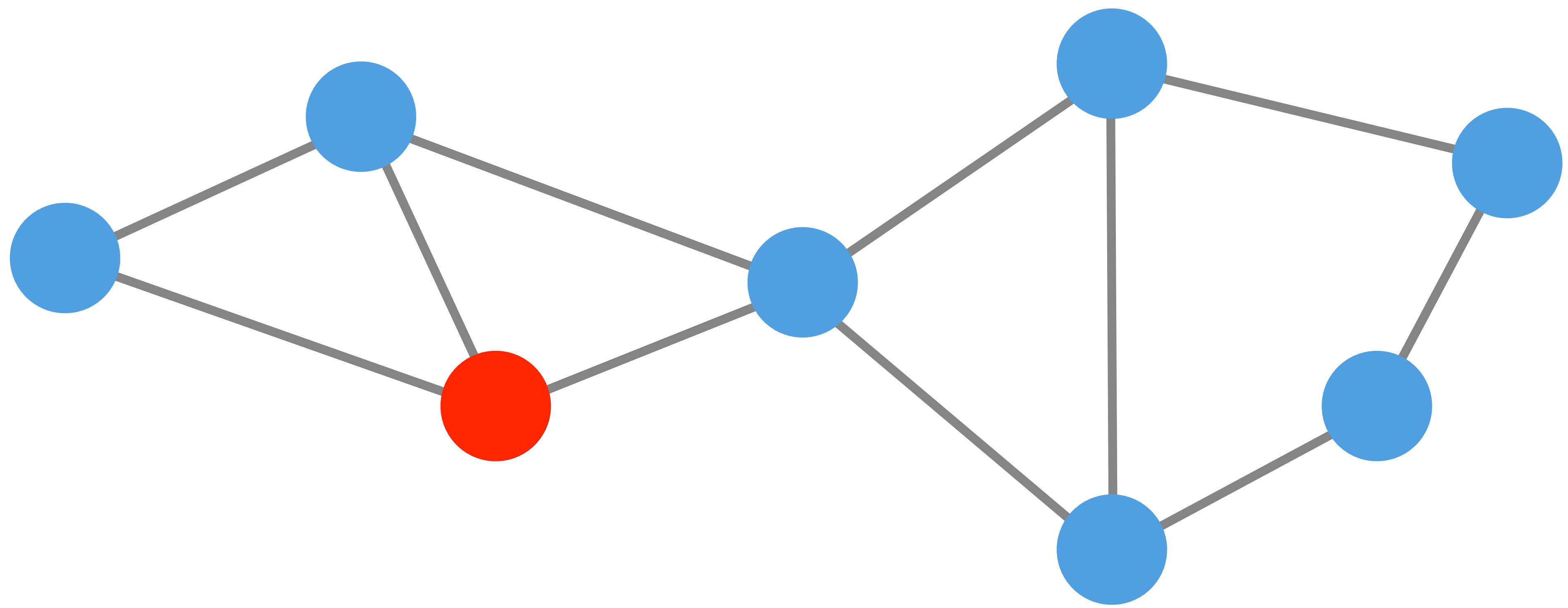
Key Concept
Observing nodes makes remainder conditionally independent

Undirected Graphical Models



Key Concept
Observing nodes makes remainder
conditionally independent

Undirected Graphical Models



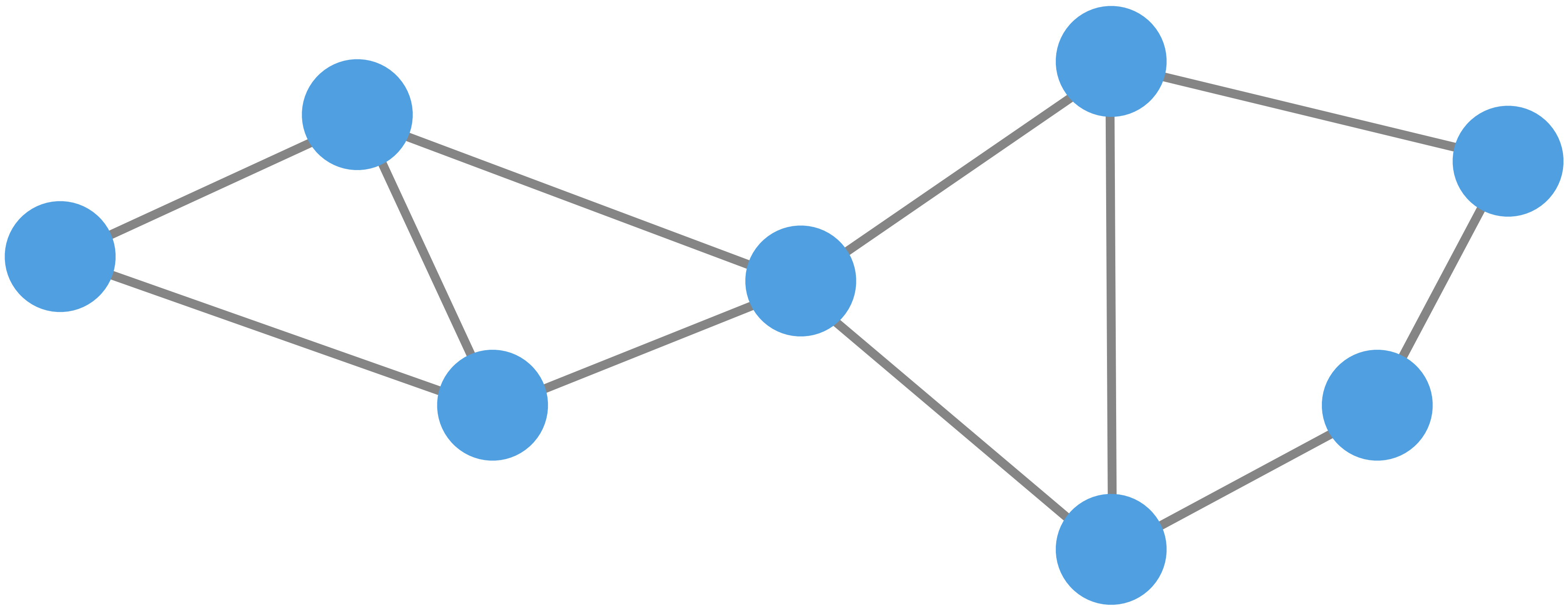
Key Concept
Observing nodes makes remainder conditionally independent

Cliques

Cliques

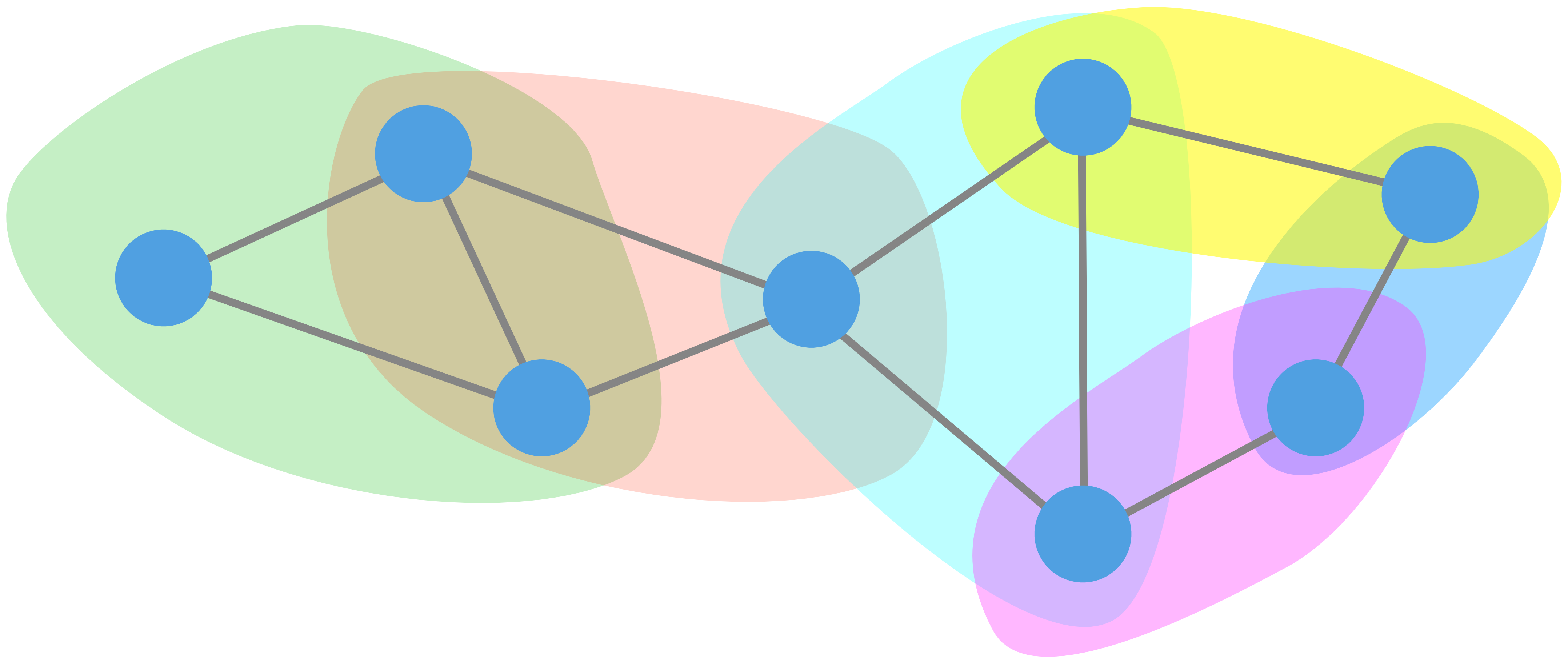
maximal fully connected subgraph

Cliques



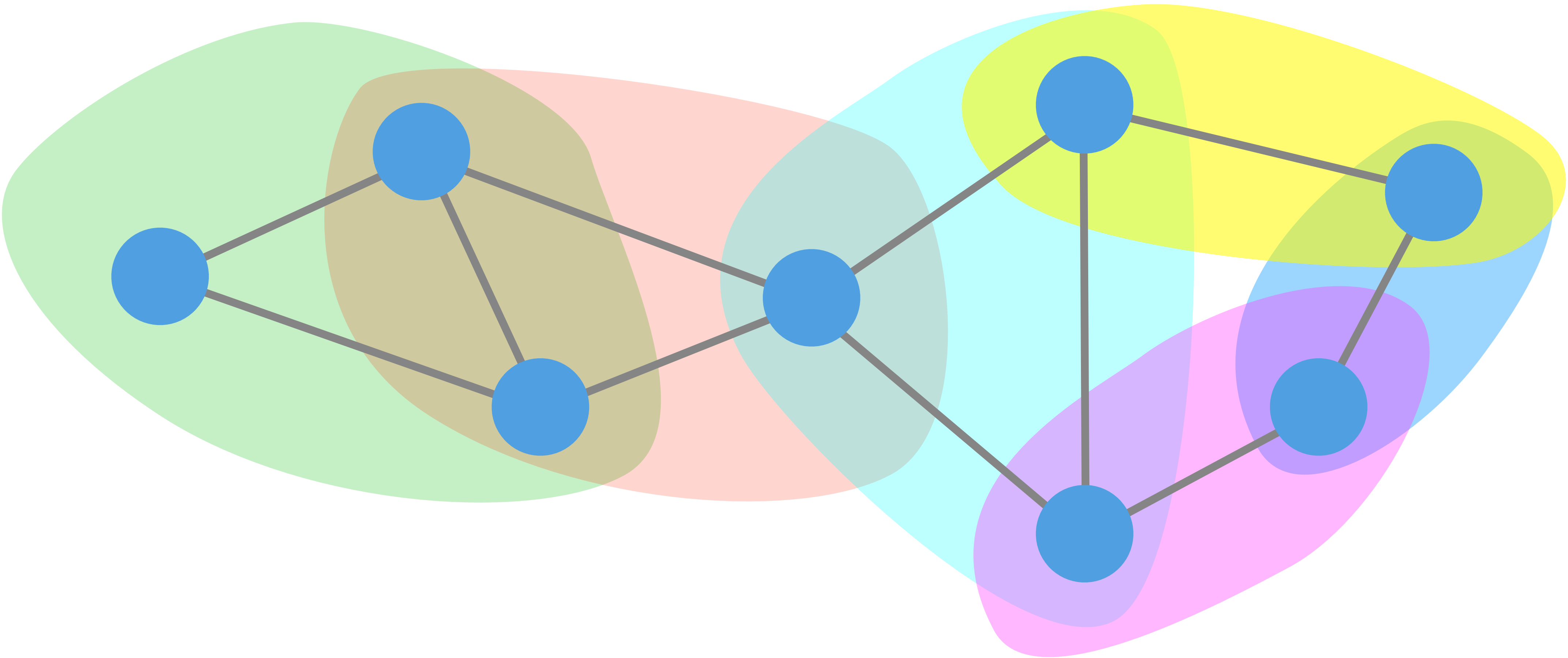
maximal fully connected subgraph

Cliques



maximal fully connected subgraph

Hammersley Clifford Theorem



If density has full support then it decomposes into products of clique potentials

$$p(x) = \prod_c \psi_c(x_c)$$

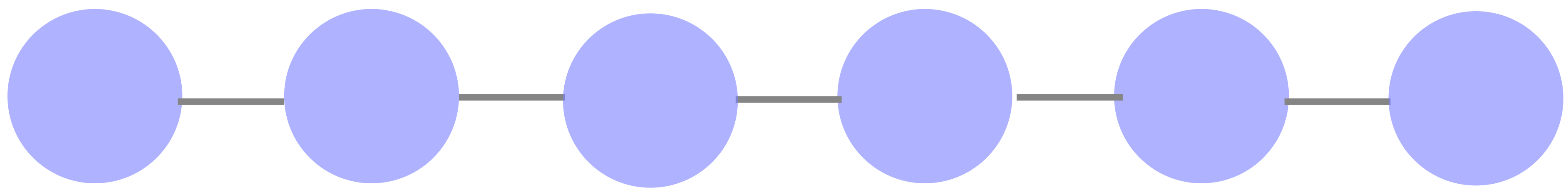
Directed vs. Undirected

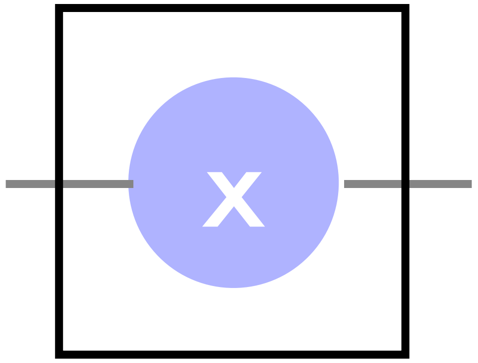
- Causal description
- Normalization automatic
- Intuitive
- Requires knowledge of dependencies
- Conditional independence tricky (Bayes Ball algorithm)

- Noncausal description (correlation only)
- Intuitive
- Easy modeling
- Normalization difficult
- Conditional independence easy to read off (graph connectivity)

Examples

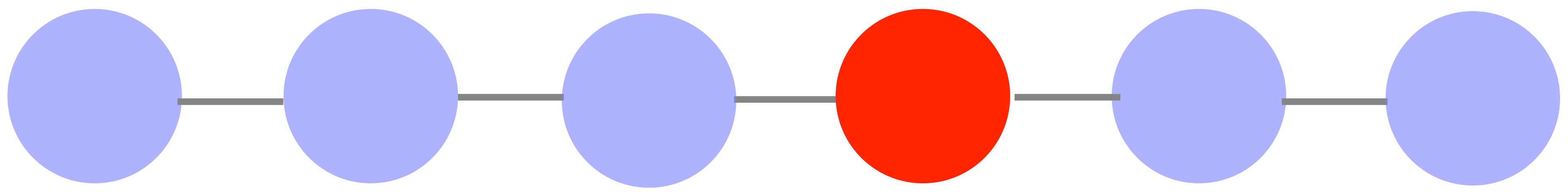
Chains

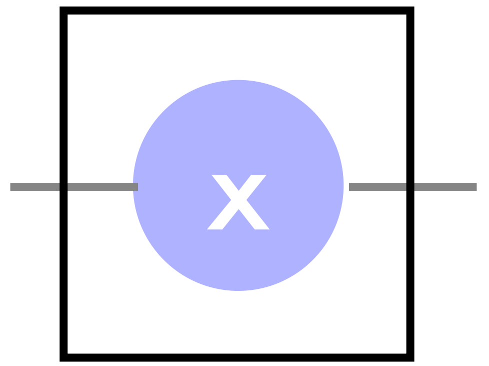




$$p(x) = \prod_i \psi_i(x_i, x_{i+1})$$

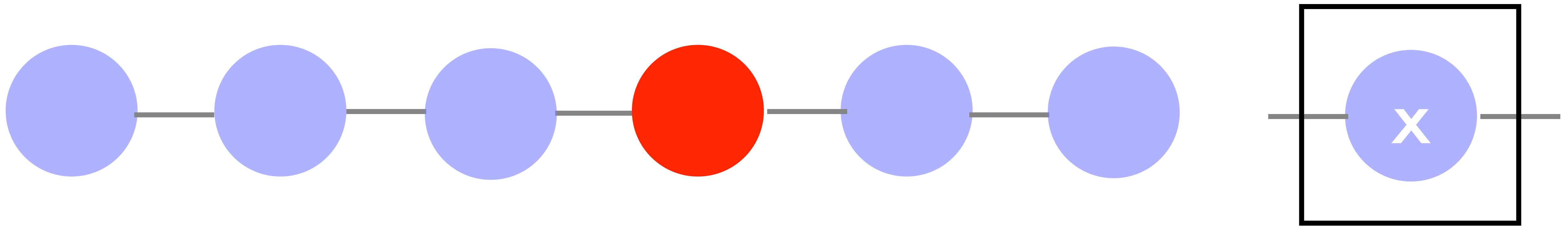
Chains



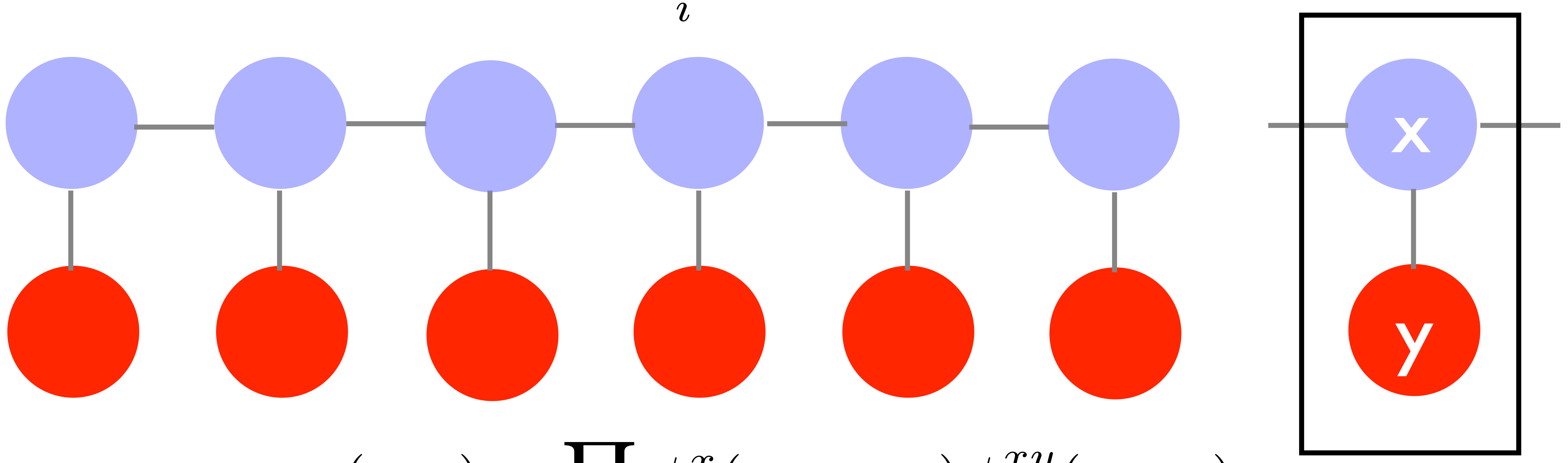


$$p(x) = \prod_i \psi_i(x_i, x_{i+1})$$

Chains

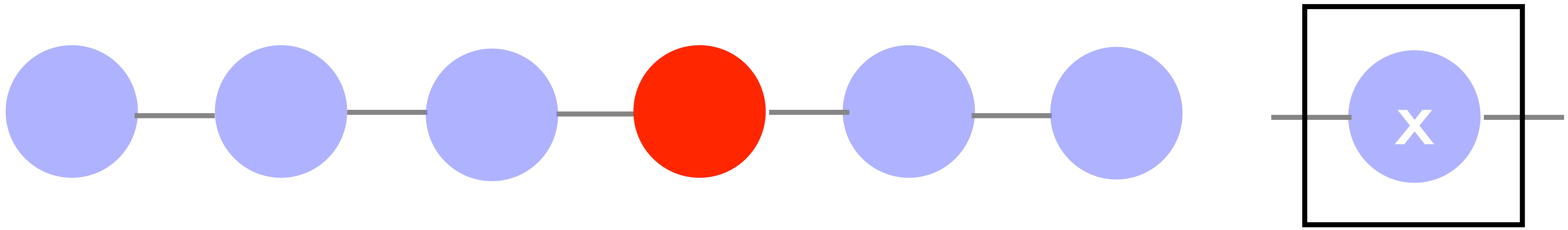


$$p(x) = \prod_i \psi_i(x_i, x_{i+1})$$

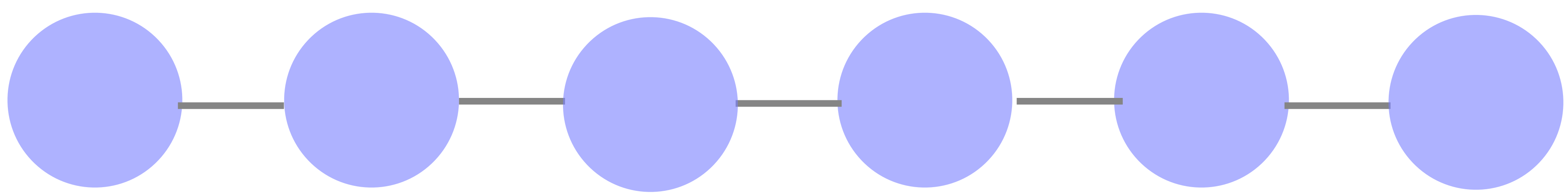


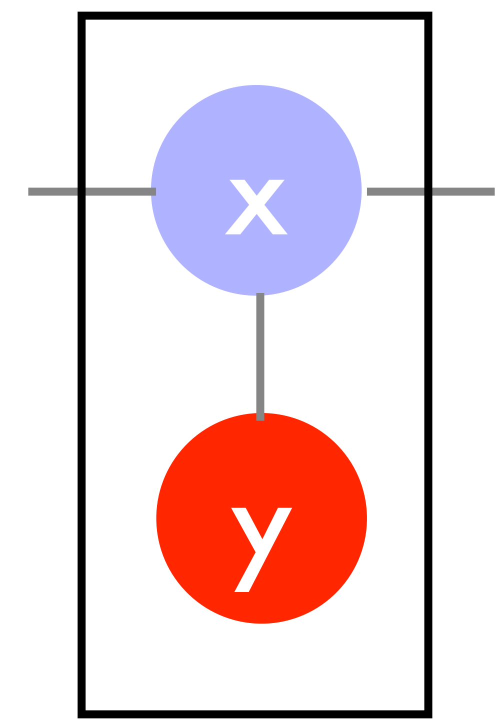
$$p(x, y) = \prod_i \psi_i^x(x_i, x_{i+1}) \psi_i^{xy}(x_i, y_i)$$

Chains



$$p(x) = \prod_i \psi_i(x_i, x_{i+1})$$

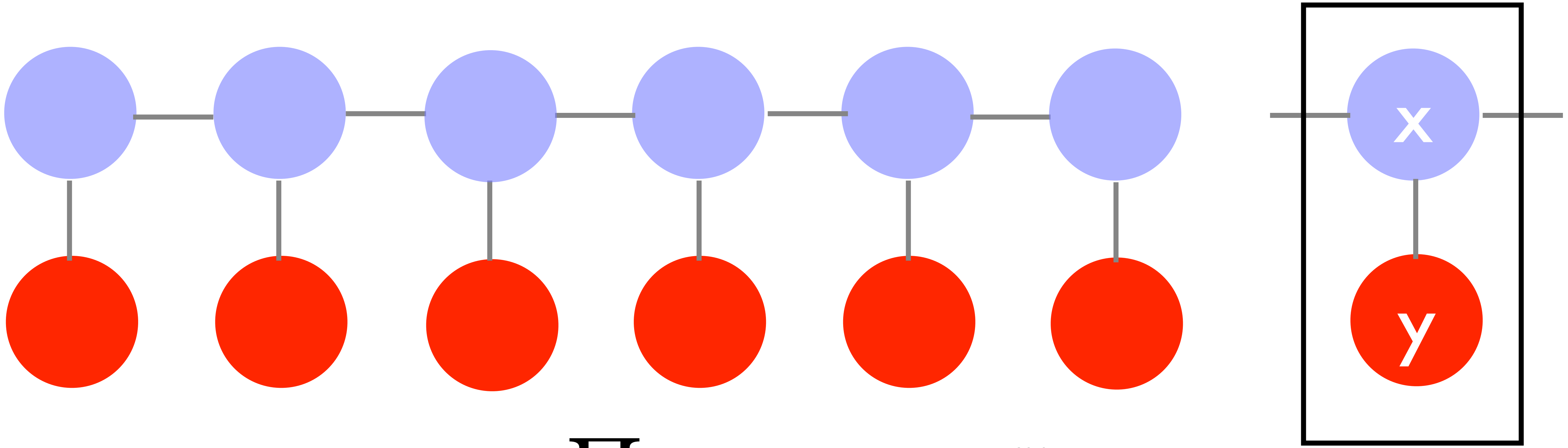




$$p(x|y) \propto \prod_i \underbrace{\psi_i^x(x_i, x_{i+1})\psi_i^{xy}(x_i, y_i)}_{=:f_i(x_i, x_{i+1})}$$

$$p(x, y) = \prod_i \psi_i^x(x_i, x_{i+1})\psi_i^{xy}(x_i, y_i)$$

Chains



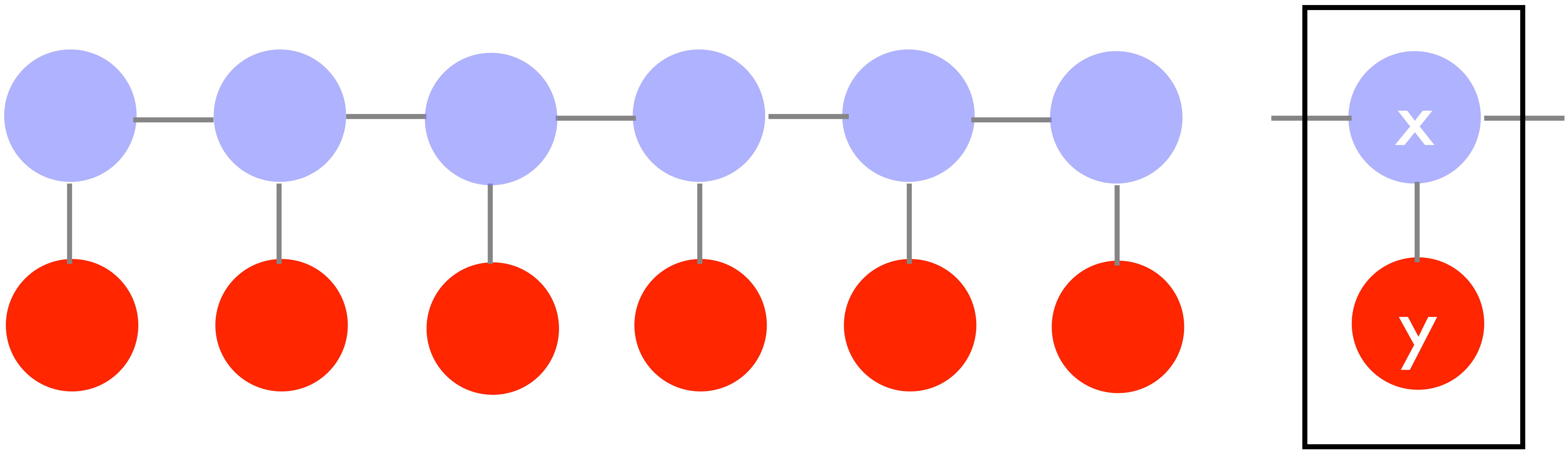
$$p(x|y) \propto \prod_i \underbrace{\psi_i^x(x_i, x_{i+1})\psi_i^{xy}(x_i, y_i)}_{=: f_i(x_i, x_{i+1})}$$

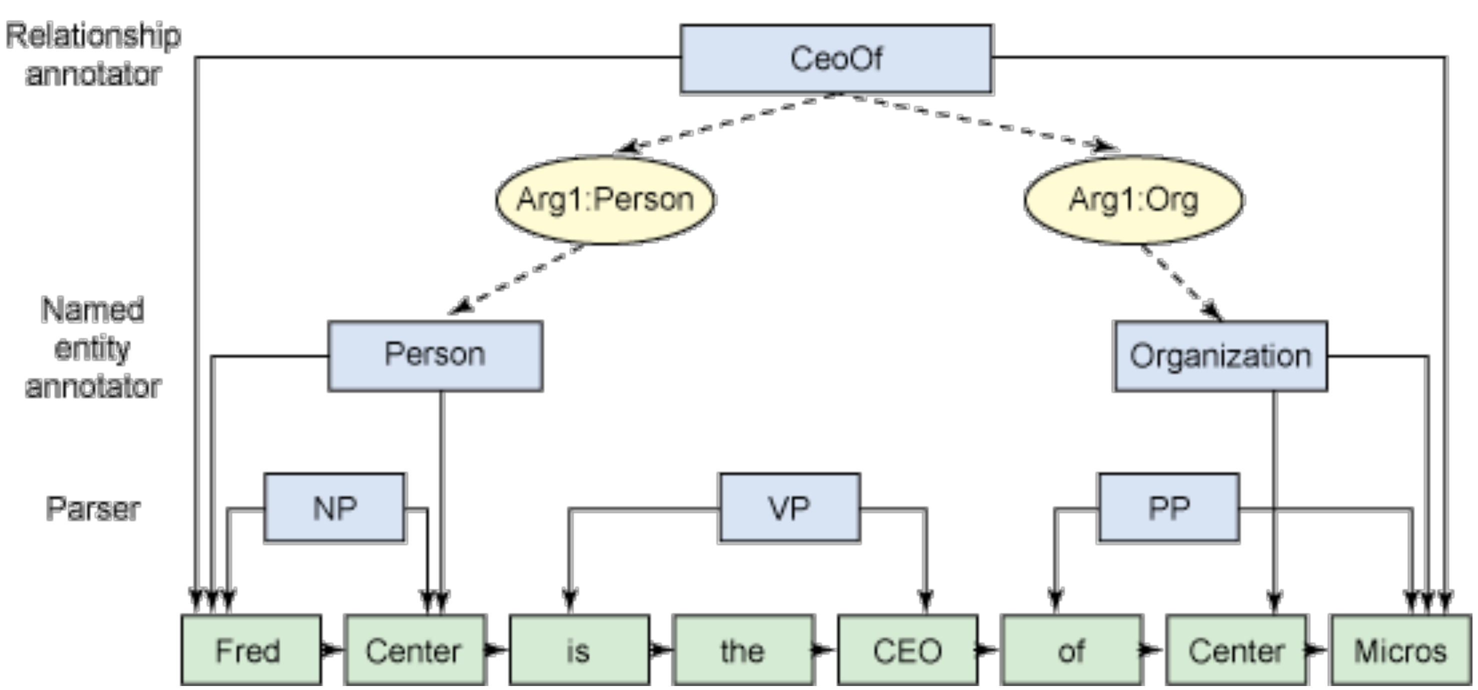
Dynamic Programming

$$l_1(x_1) = 1 \text{ and } l_{i+1}(x_{i+1}) = \sum_{x_i} l_i(x_i) f_i(x_i, x_{i+1})$$

$$r_n(x_n) = 1 \text{ and } r_i(x_i) = \sum_{x_{i+1}} r_{i+1}(x_{i+1}) f_i(x_i, x_{i+1})$$

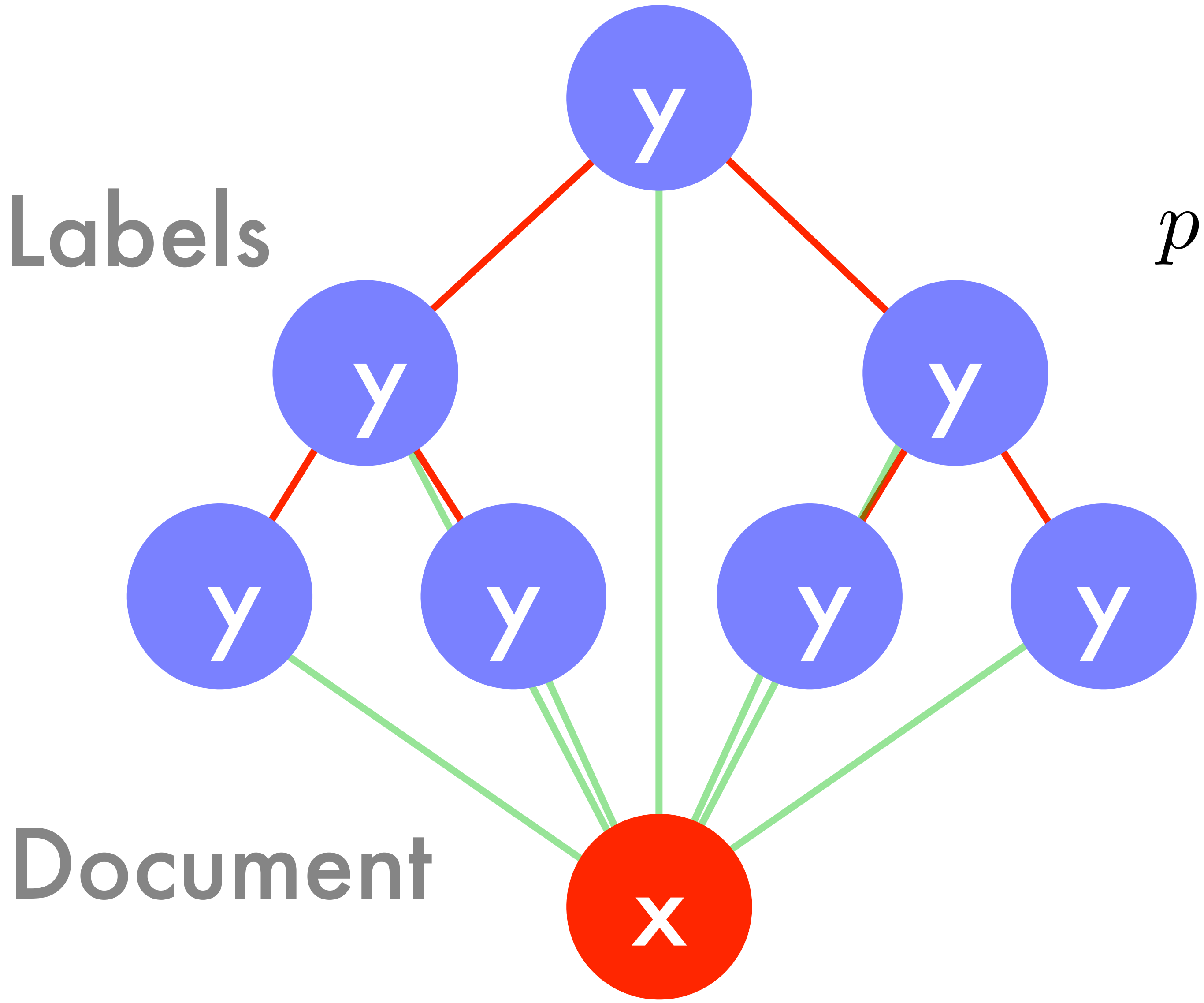
Named Entity Tagging





$$p(x|y) \propto \prod_i \underbrace{\psi_i^x(x_i, x_{i+1})\psi_i^{xy}(x_i, y_i)}_{=:f_i(x_i, x_{i+1})}$$

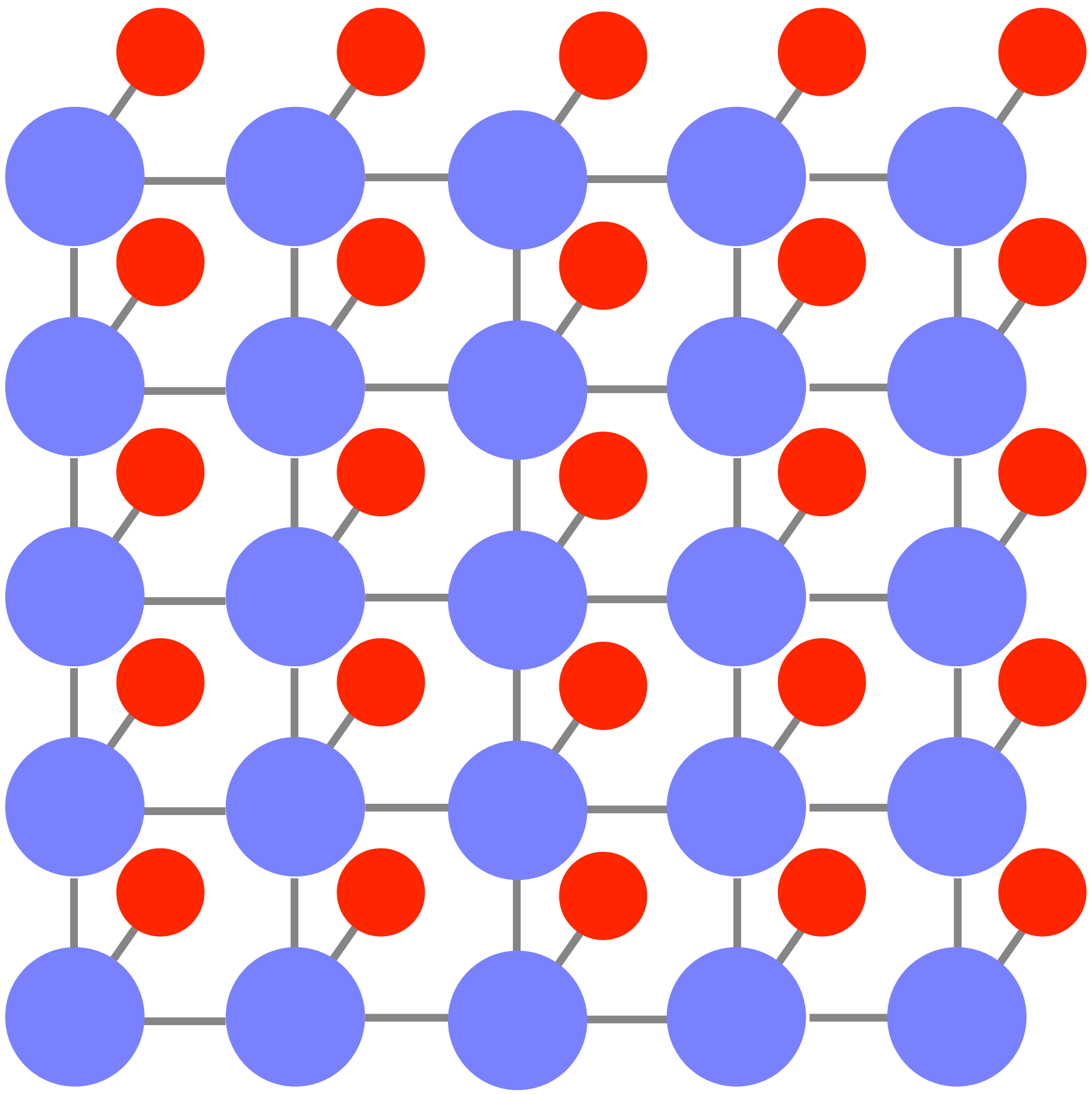
Trees + Ontologies

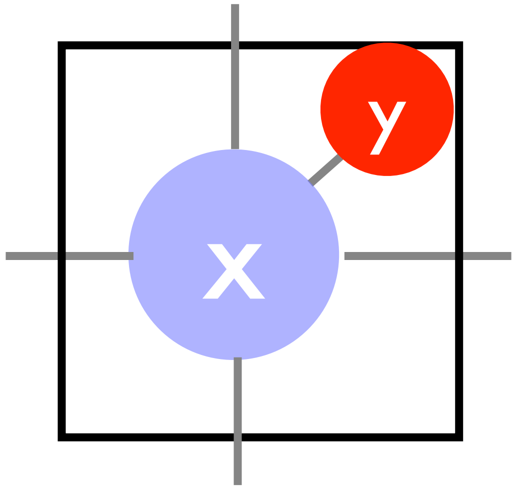


$$p(y|x) = \prod_i \psi(y_i, y_{\mathrm{parent}(i)}, x)$$

- Ontology classification (e.g. YDir, DMOZ)

Spin Glasses + Images



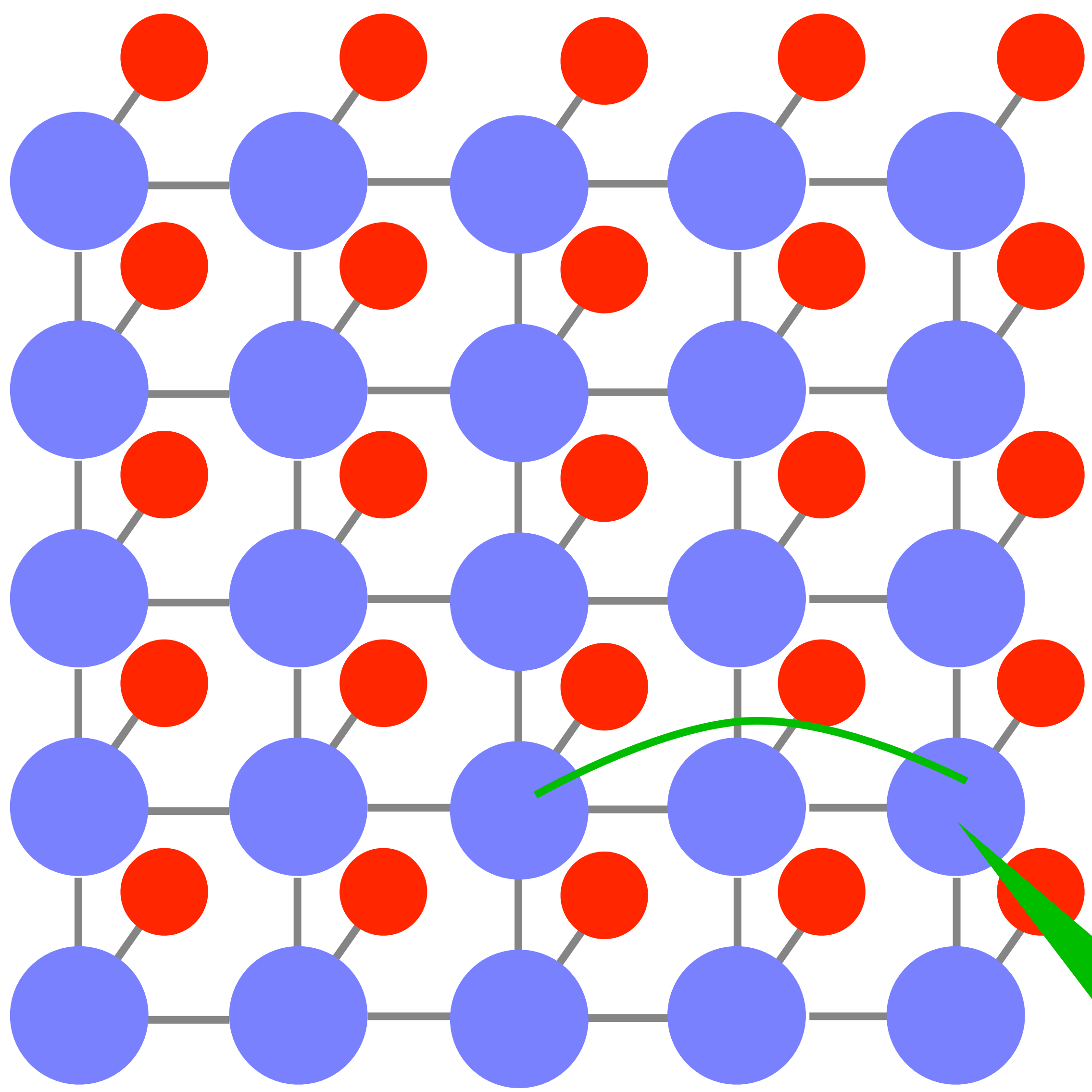


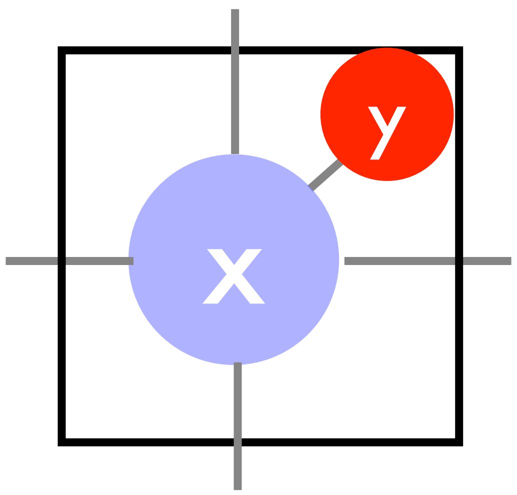
observed pixels

real image

$$p(x|y) = \prod_{ij} \psi^{\text{right}}(x_{ij}, x_{i+1,j}) \psi^{\text{up}}(x_{ij}, x_{i,j+1}) \psi^{xy}(x_{ij}, y_{ij})$$

Spin Glasses + Images

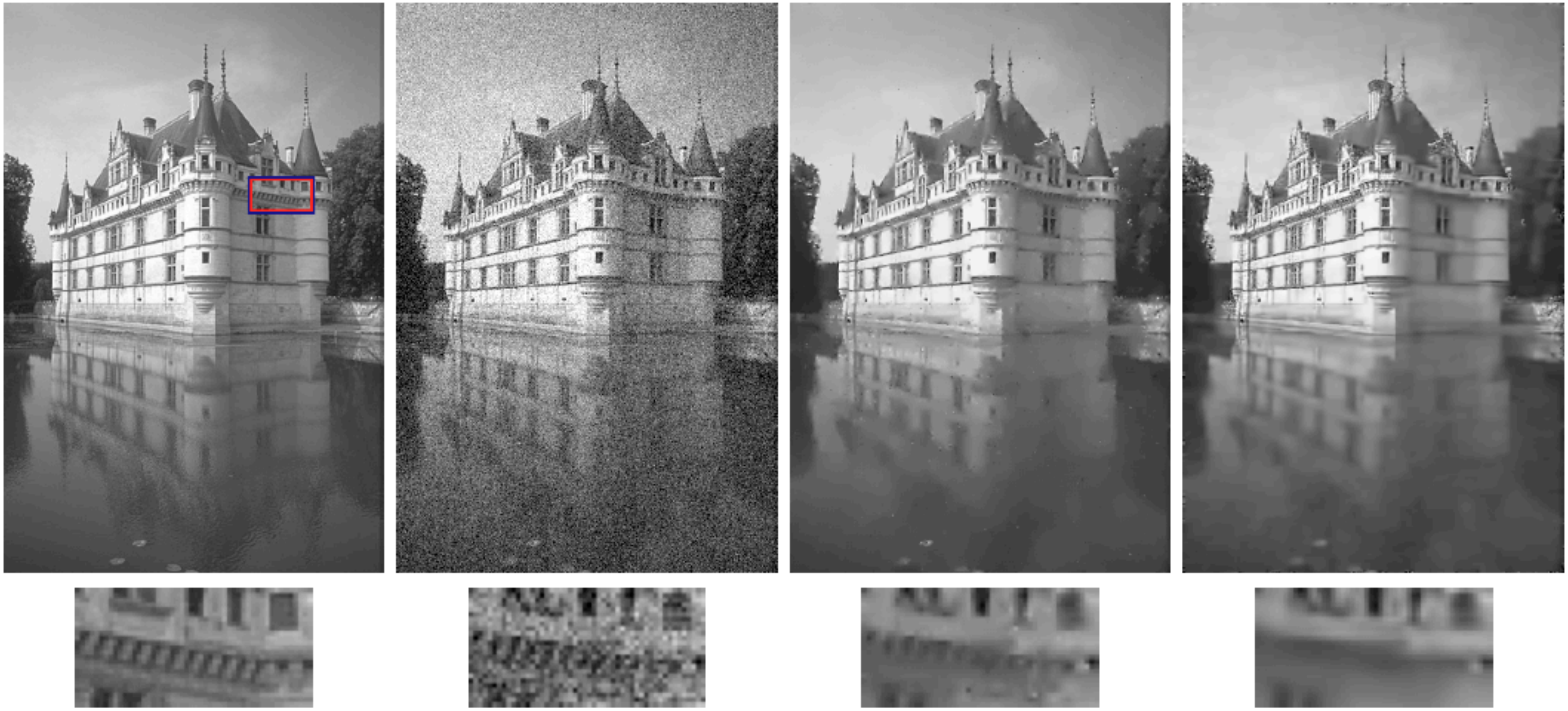




observed pixels
real image

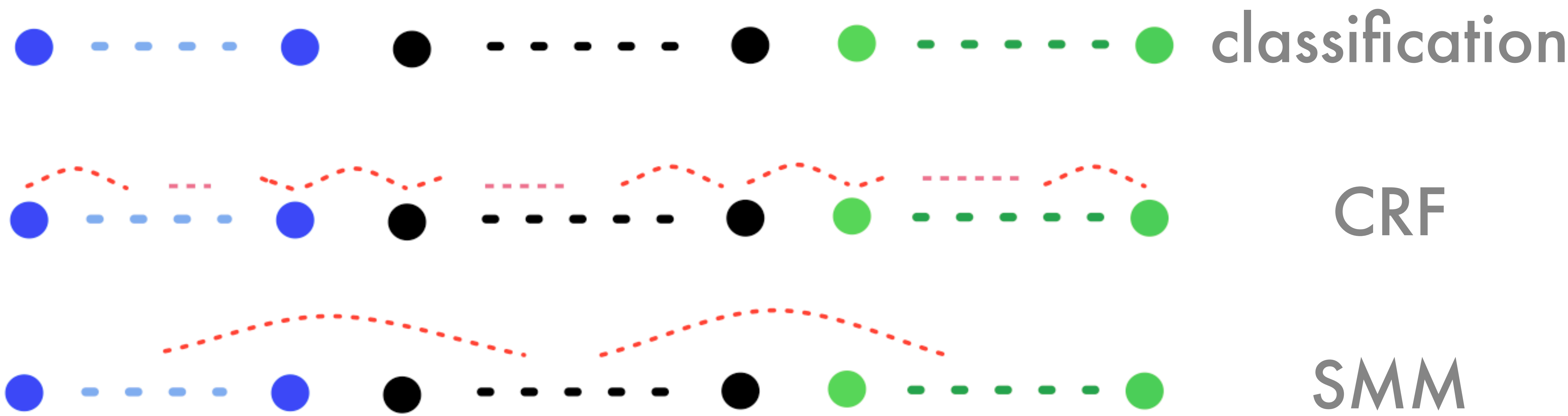
$$p(x|y) = \prod_{ij} \psi^{\text{right}}(x_{ij}, x_{i+1,j})\psi^{\text{up}}(x_{ij}, x_{i,j+1})\psi^{xy}(x_{ij}, y_{ij})$$

Image Denoising



Li&Huttenlocher, ECCV'08

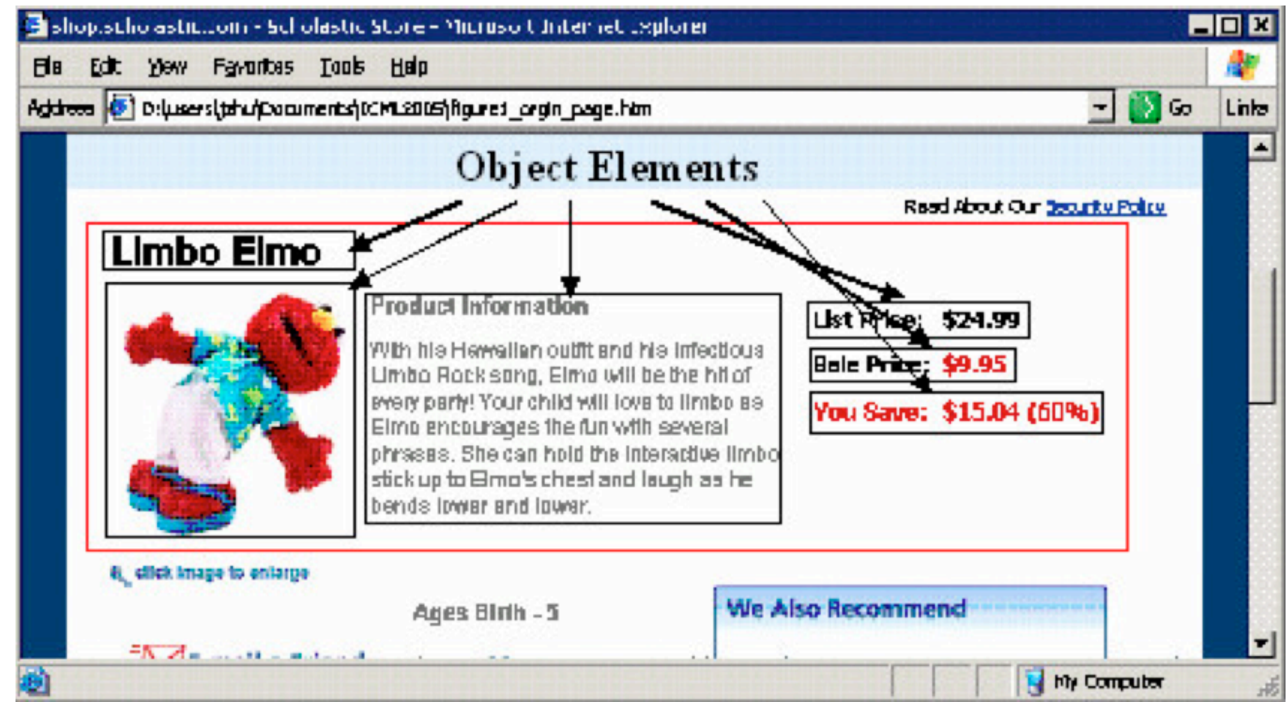
Semi-Markov Models

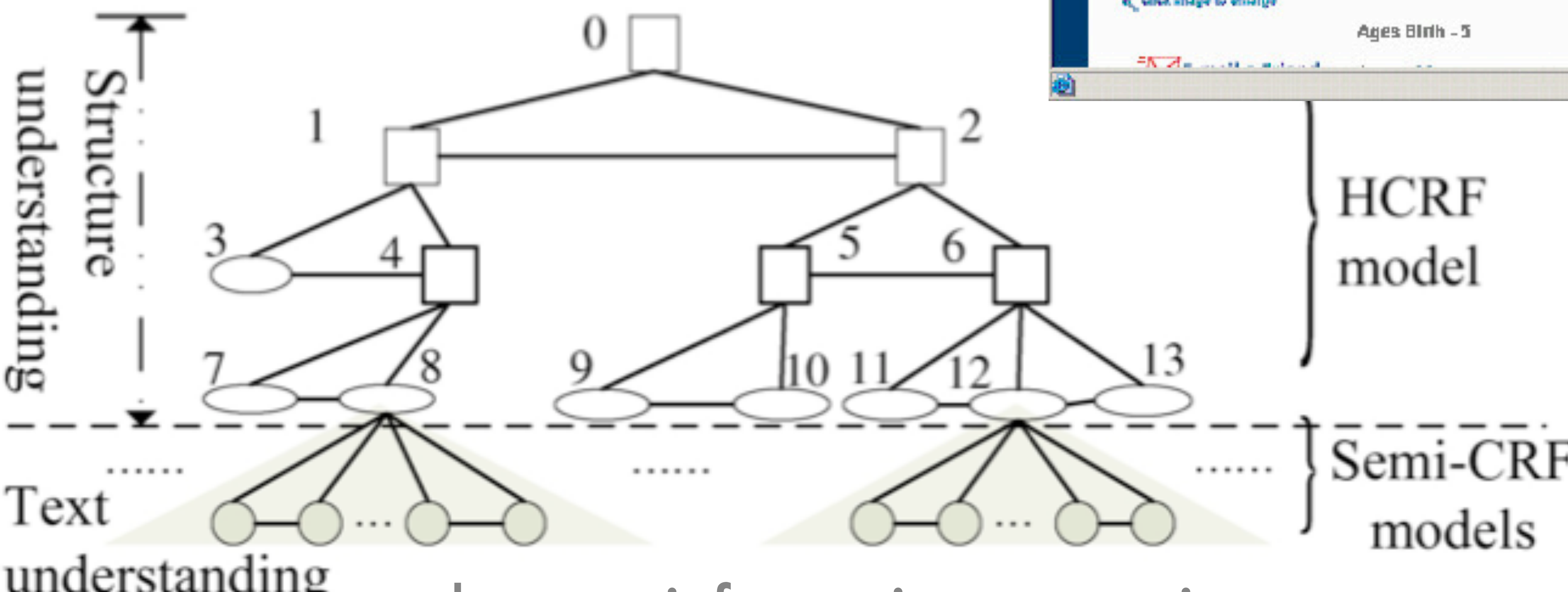


- Flexible length of an episode
- Segmentation between episodes

phrase segmentation, activity recognition, motion data analysis
Shi, Smola, Altun, Vishwanathan, Li, 2007-2009

2D CRF for Webpages





web page information extraction, segmentation, annotation
Bo, Zhu, Nie, Wen, Hon, 2005-2007

Exponential Families and Graphical Models

Exponential Family Reunion

- Density function

$$p(x;\theta) = \exp\left(\langle \phi(x), \theta \rangle - g(\theta)\right)$$
$$\text{where } g(\theta) = \log \sum_{x'} \exp\left(\langle \phi(x'), \theta \rangle\right)$$

- Log partition function generates cumulants

$$\partial_\theta g(\theta) = \mathbf{E}\left[\phi(x)\right]$$
$$\partial_\theta^2 g(\theta) = \mathrm{Var}\left[\phi(x)\right]$$

- g is convex (second derivative is p.s.d.)

Log Partition Function

$$p(x|\theta) = e^{\langle \phi(x), \theta \rangle - g(\theta)}$$

Unconditional model

$$g(\theta) = \log \sum_x e^{\langle \phi(x), \theta \rangle}$$

$$\partial_\theta g(\theta) = \frac{\sum_x \phi(x) e^{\langle \phi(x), \theta \rangle}}{\sum_x e^{\langle \phi(x), \theta \rangle}} = \sum_x \phi(x) e^{\langle \phi(x), \theta \rangle - g(\theta)}$$

$$p(y|\theta, x) = e^{\langle \phi(x,y), \theta \rangle - g(\theta|x)}$$

Conditional model

$$g(\theta|x) = \log \sum_y e^{\langle \phi(x,y), \theta \rangle}$$

$$\partial_\theta g(\theta|x) = \frac{\sum_y \phi(x,y) e^{\langle \phi(x,y), \theta \rangle}}{\sum_y e^{\langle \phi(x,y), \theta \rangle}} = \sum_y \phi(x,y) e^{\langle \phi(x,y), \theta \rangle - g(\theta|x)}$$

Estimation

- Conditional log-likelihood

$$\log p(y|x;\theta) = \langle \phi(x,y), \theta \rangle - g(\theta|x)$$

- Log-posterior (Gaussian Prior)

$$\log p(\theta|X,Y) = \sum_i \log(y_i|x_i;\theta) + \log p(\theta) + \text{const.}$$

$$= \left\langle \sum_i \phi(x_i,y_i), \theta \right\rangle - \sum_i g(\theta|x_i) - \frac{1}{2\sigma^2} \left\|\theta\right\|^2 + \text{const.}$$

- First order optimality conditions

expensive

prior

maxent model

$$\sum_i \phi(x_i,y_i) = \sum_i \mathbf{E}_{y|x_i}\left[\phi(x_i,y)\right] + \frac{1}{\sigma^2}\theta$$

Logistic Regression

- **Label space**

$$\phi(x, y) = y\phi(x) \text{ where } y \in \{\pm 1\}$$

- **Log-partition function**

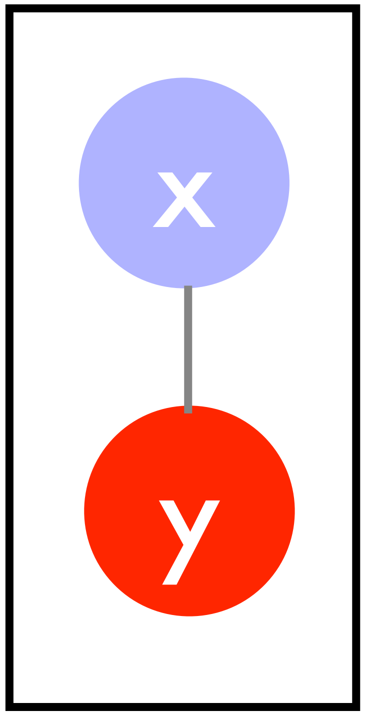
$$g(\theta|x) = \log\left[e^{1\cdot\langle\phi(x),\theta\rangle} + e^{-1\cdot\langle\phi(x),\theta\rangle}\right] = \log 2\cosh\langle\phi(x),\theta\rangle$$

- **Convex minimization problem**

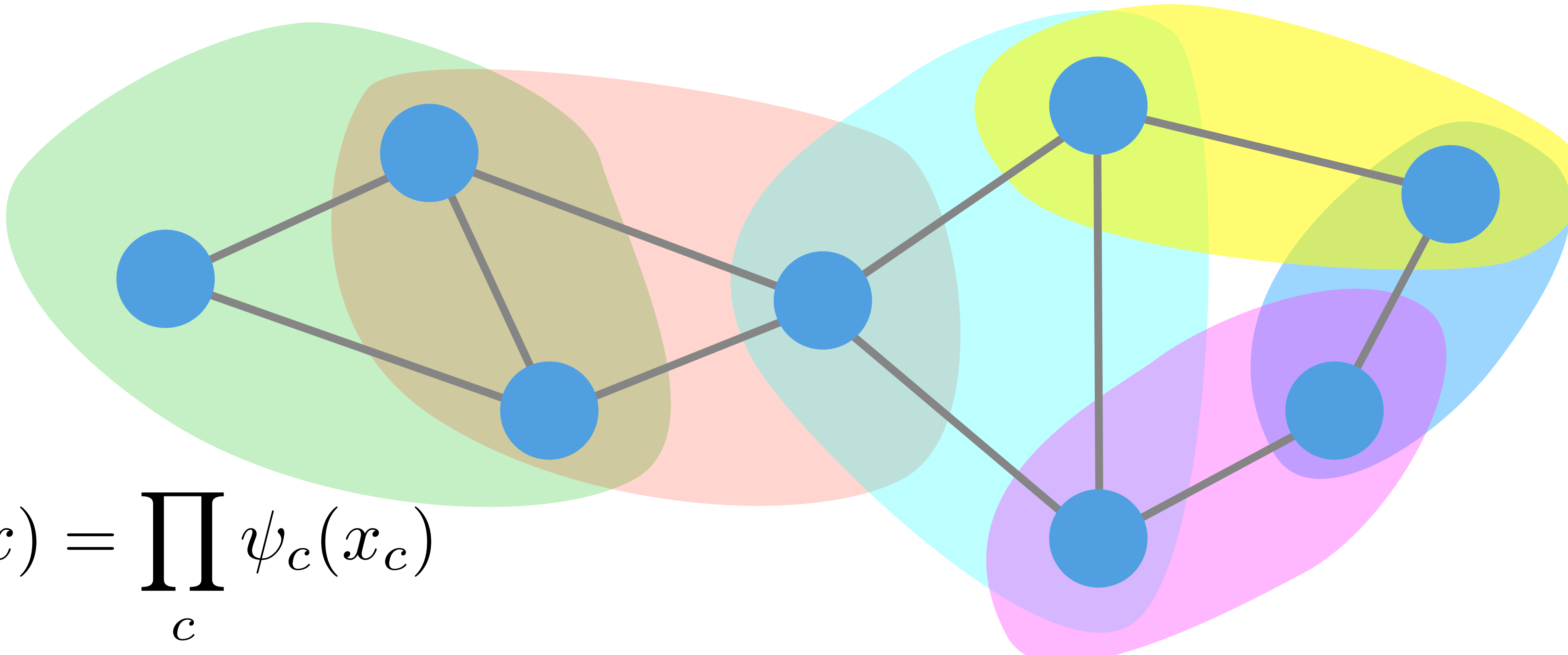
$$\underset{\theta}{\text{minimize}}\ \frac{1}{2\sigma^2}\|\theta\|^2 + \sum_i \log 2\cosh\langle\phi(x_i),\theta\rangle - y_i\langle\phi(x_i,\theta\rangle$$

- **Prediction**

$$p(y|x,\theta) = \frac{e^{y\langle\phi(x),\theta\rangle}}{e^{\langle\phi(x),\theta\rangle} + e^{-\langle\phi(x),\theta\rangle}} = \frac{1}{1+e^{-2y\langle\phi(x),\theta\rangle}}$$



Exponential Clique Decomposition



$$p(x) = \prod_c \psi_c(x_c)$$

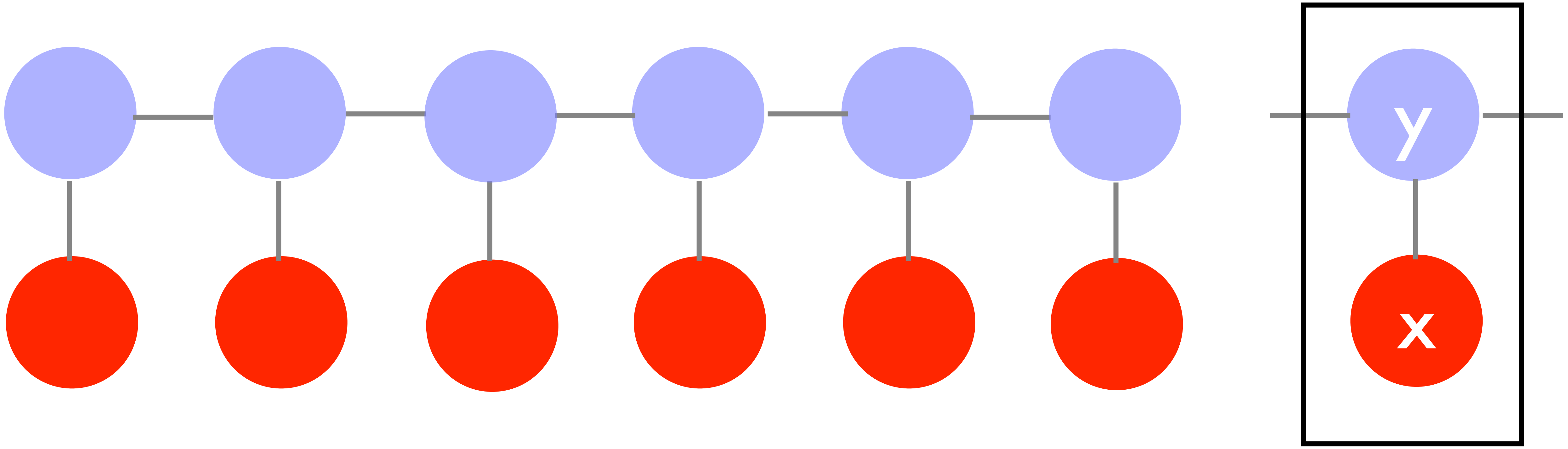
Theorem: Clique decomposition holds in sufficient statistics

$$\phi(x) = (\ldots, \phi_c(x_c), \ldots) \text{ and } \langle \phi(x), \theta \rangle = \sum_c \langle \phi_c(x_c), \theta_c \rangle$$

Corollary: we only need expectations on cliques

$$\mathbf{E}_x[\phi(x)] = (\ldots, \mathbf{E}_{x_c}[\phi_c(x_c)], \ldots)$$

Conditional Random Fields



$$\phi(x) = (y_1\phi_x(x_1), \ldots, y_n\phi_x(x_n), \phi_y(y_1, y_2), \ldots, \phi_y(y_{n-1}, y_n))$$

$$\langle \phi(x), \theta \rangle = \sum_i \langle \phi_x(x_i, y_i), \theta_x \rangle + \sum_i \langle \phi_y(y_i, y_{i+1}), \theta_y \rangle$$

$$g(\theta|x) = \sum_y \prod_i f_i(y_i, y_{i+1}) \text{ where}$$

$$f_i(y_i, y_{i+1}) = e^{\langle \phi_x(x_i, y_i), \theta_x \rangle + \langle \phi_y(y_i, y_{i+1}), \theta_y \rangle}$$

dynamic programming

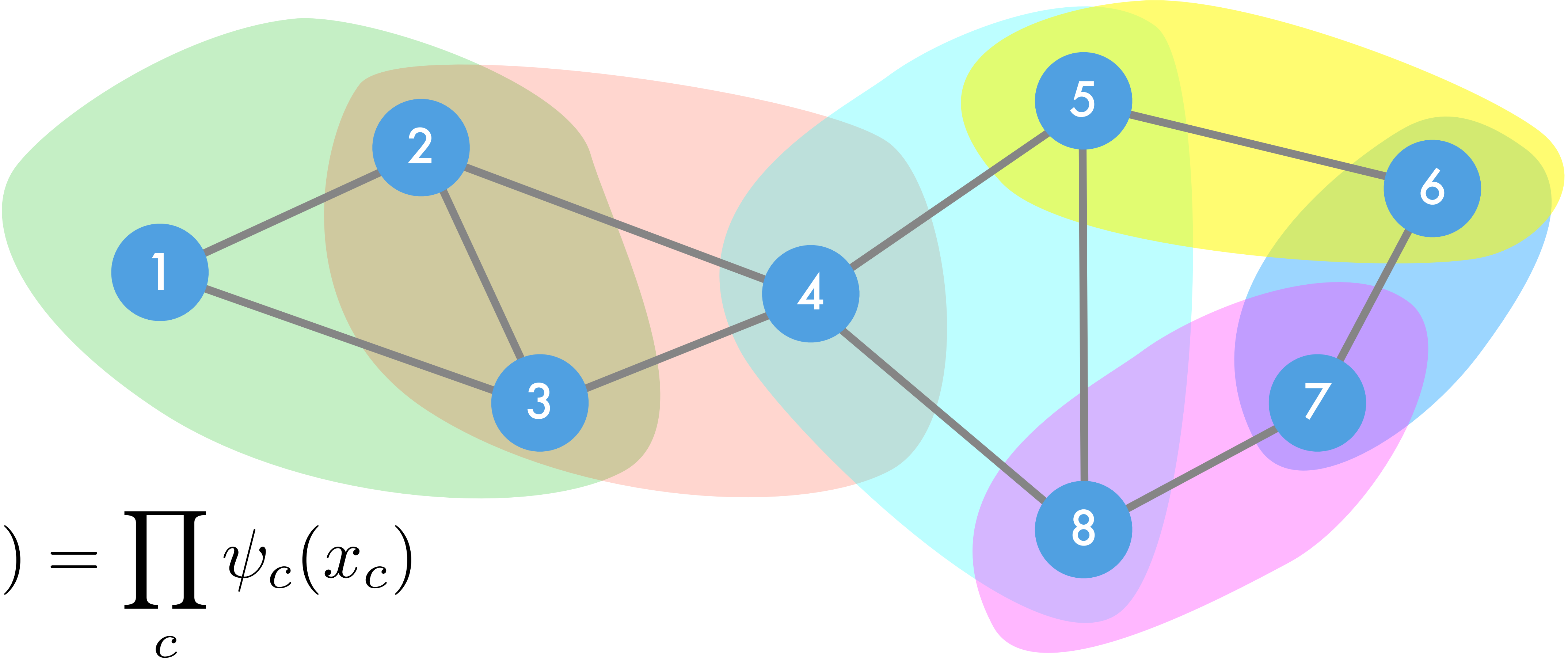
Conditional Random Fields

- Compute distribution over marginal and adjacent labels
- Take conditional expectations
- Take update step (batch or online)

- More general techniques for computing normalization via message passing ...

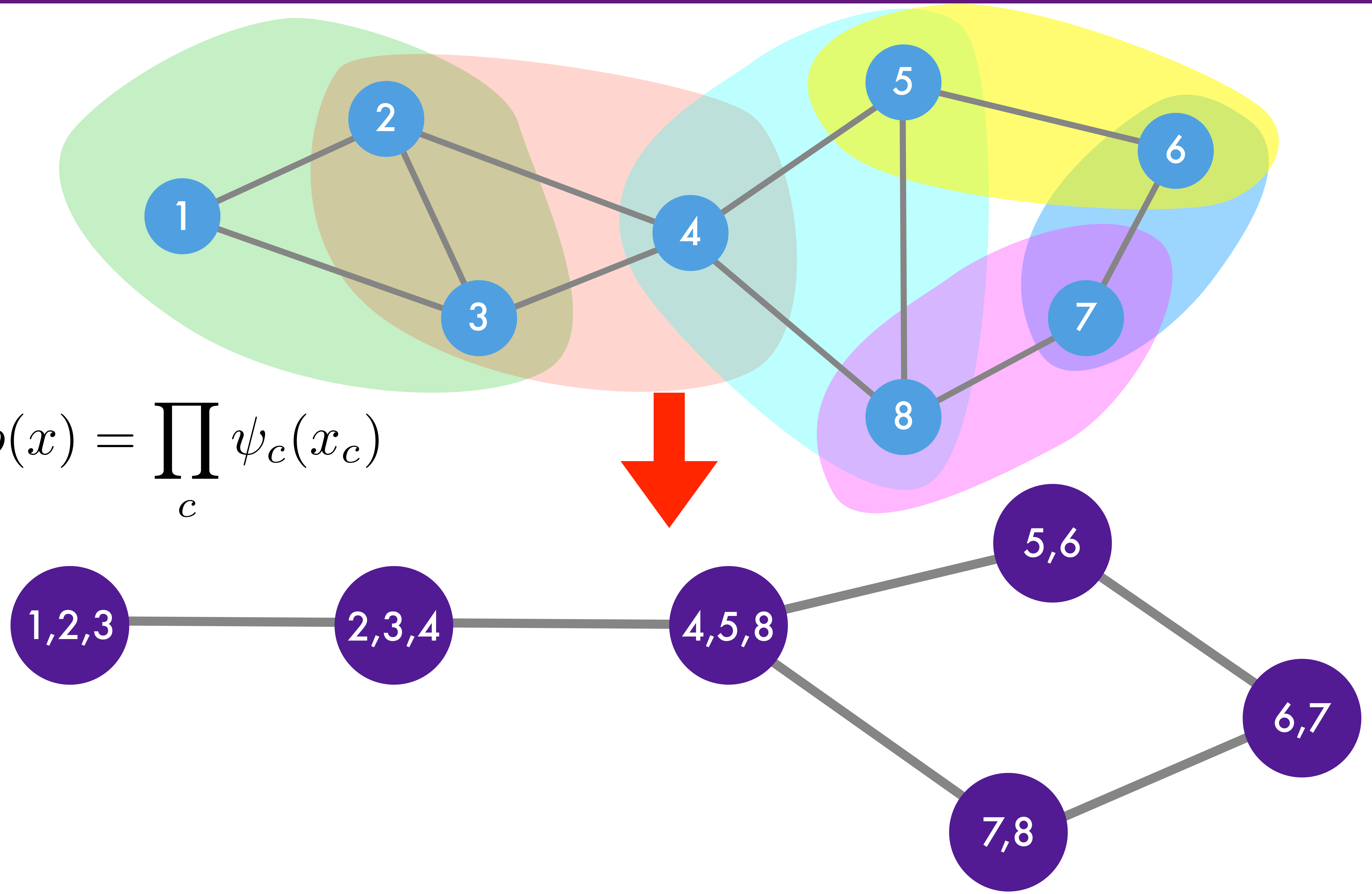
Dynamic Programming + Message Passing

Clique Graph



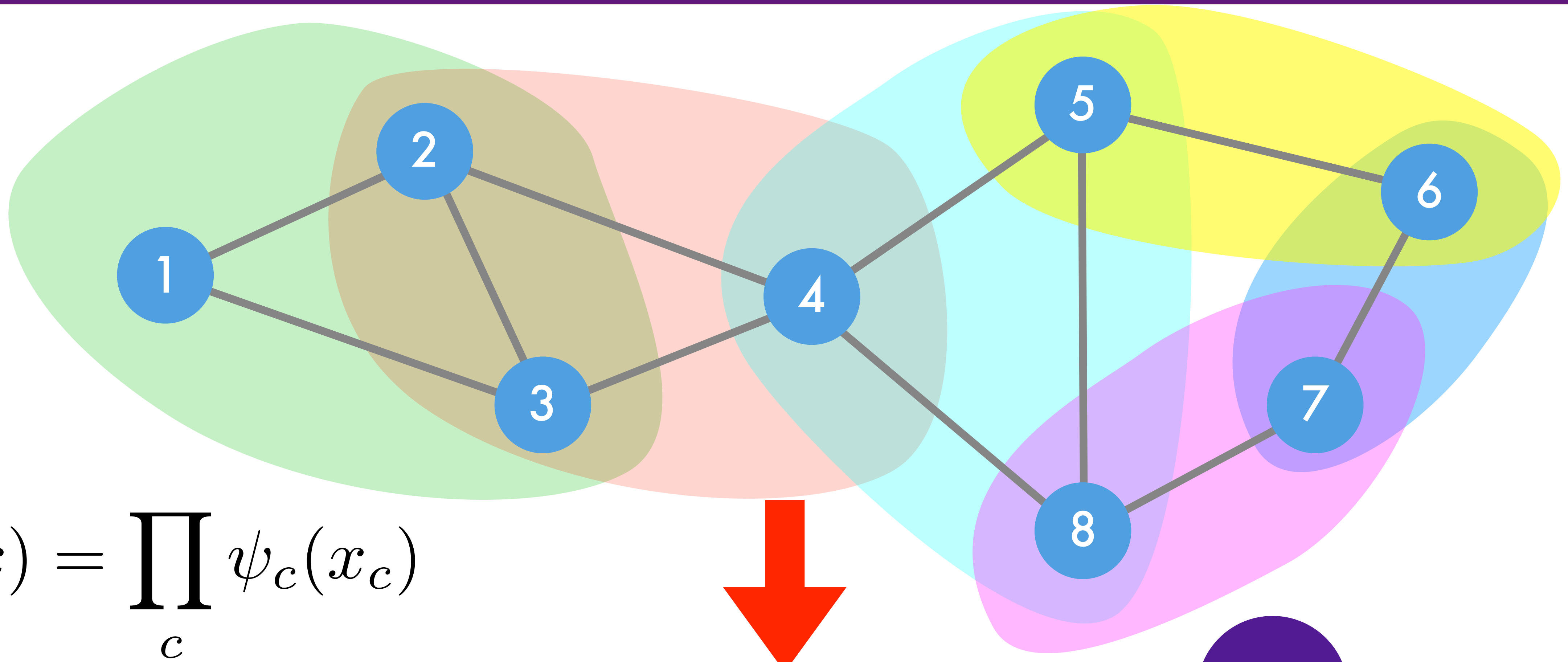
$$p(x) = \prod_c \psi_c(x_c)$$

Clique Graph

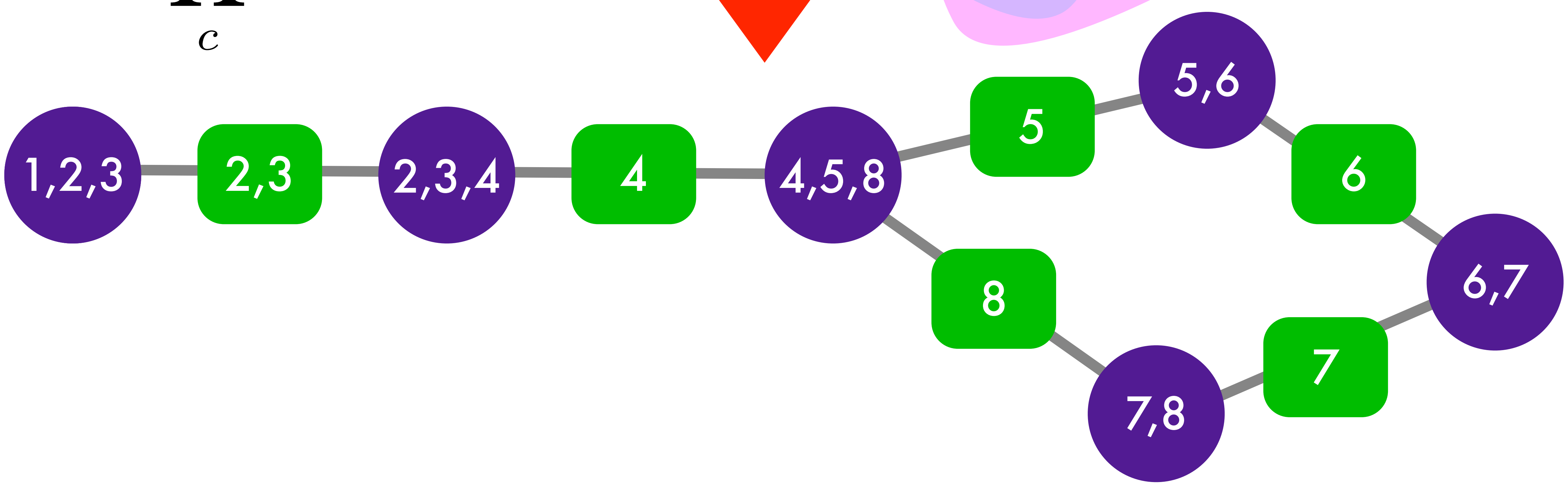


$$p(x) = \prod_c \psi_c(x_c)$$

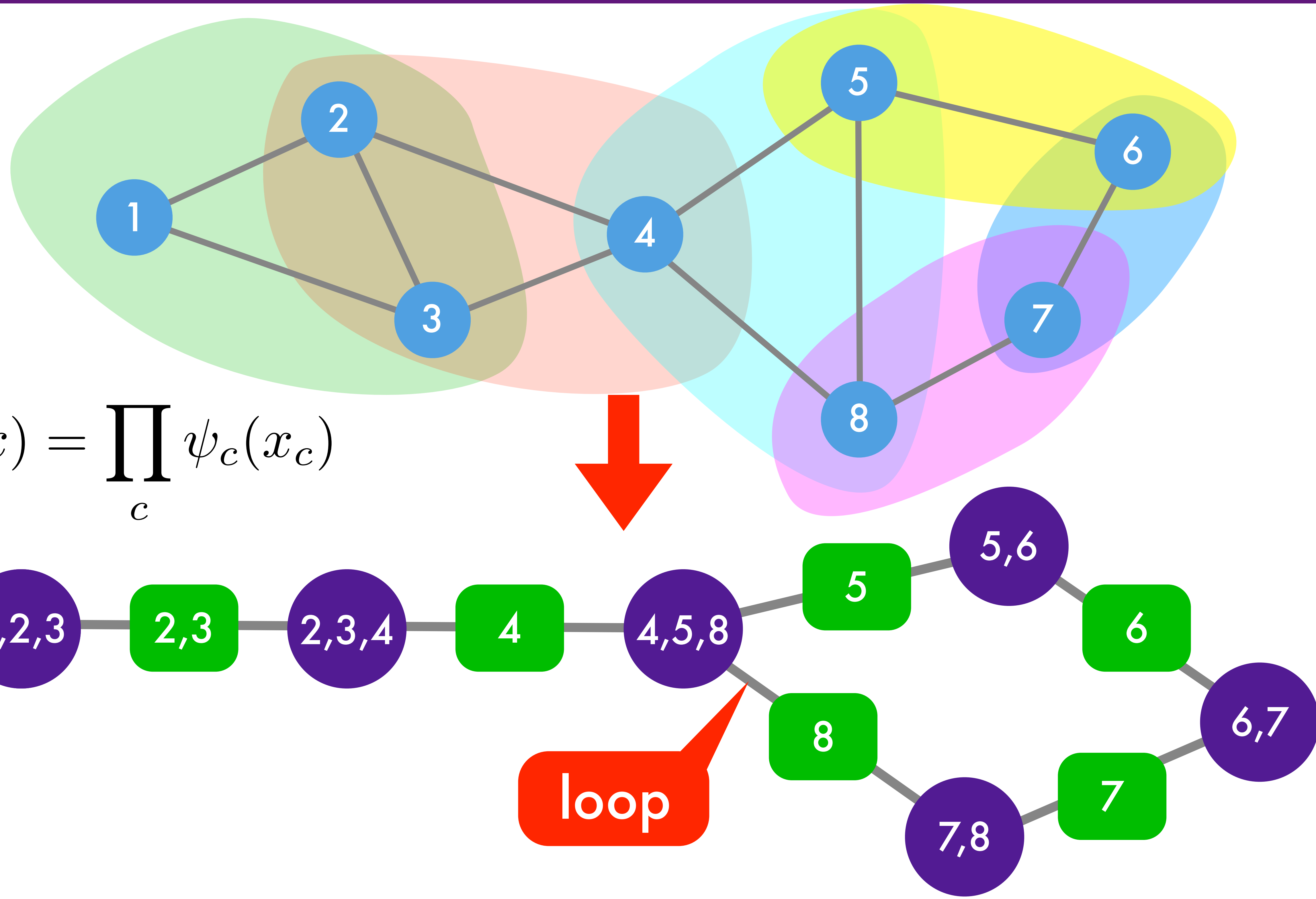
Clique Graph



$$p(x) = \prod_c \psi_c(x_c)$$

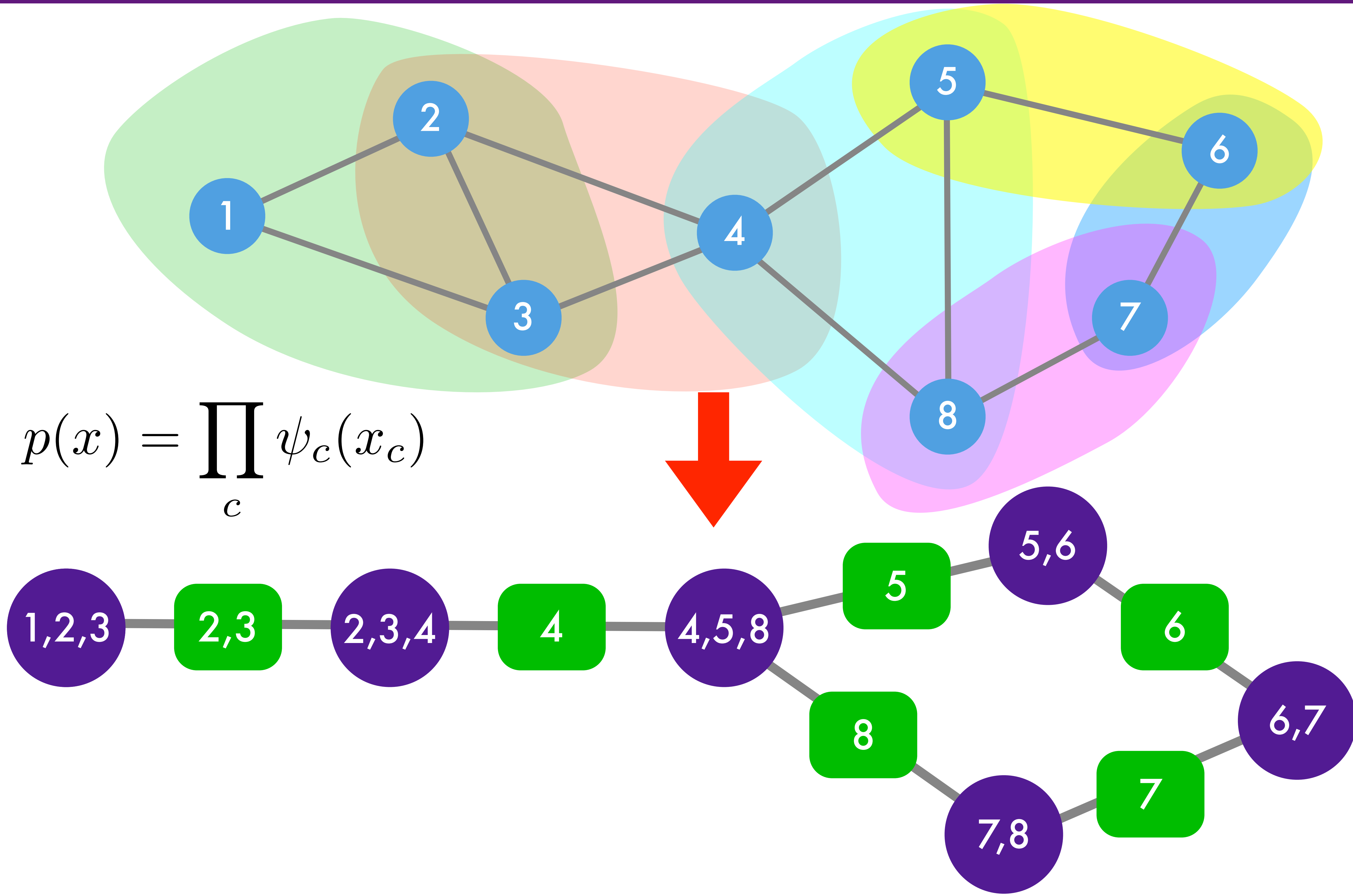


Clique Graph



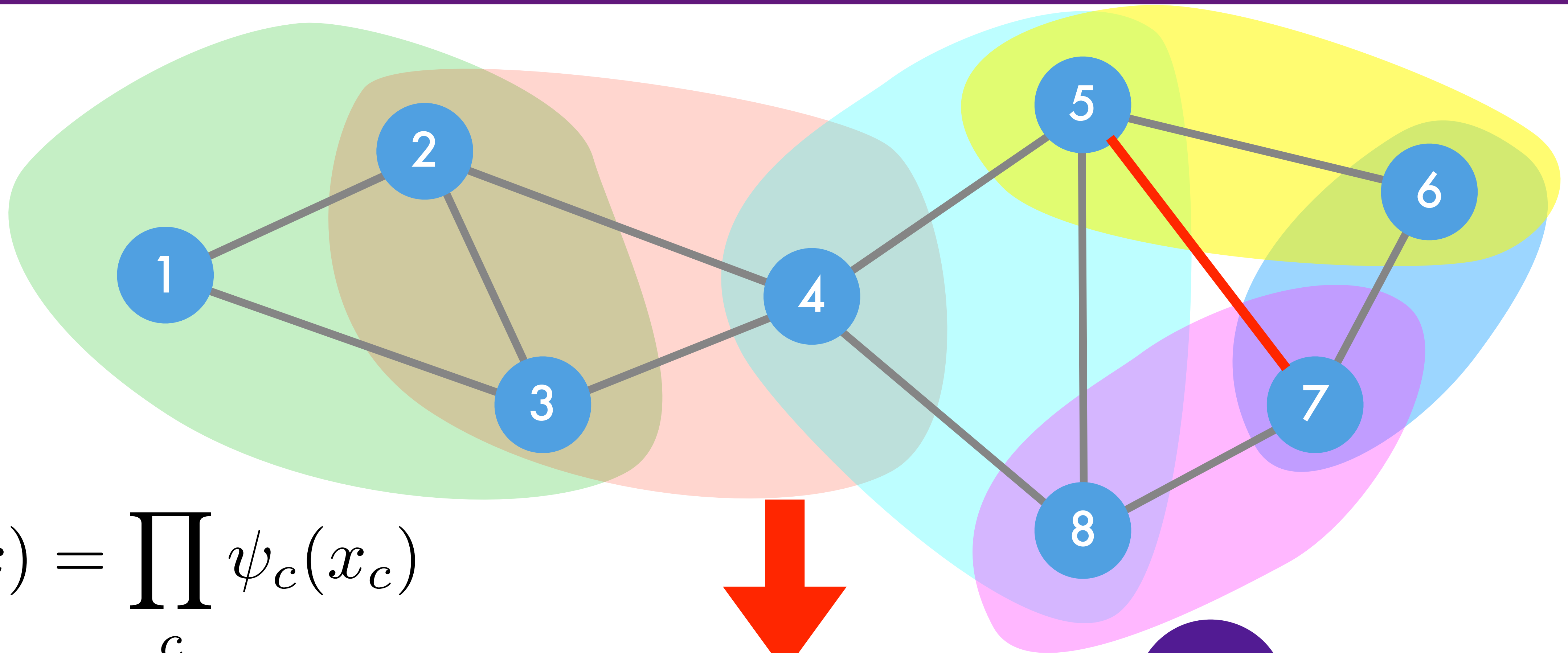
$$p(x) = \prod_c \psi_c(x_c)$$

Junction Tree / Triangulation

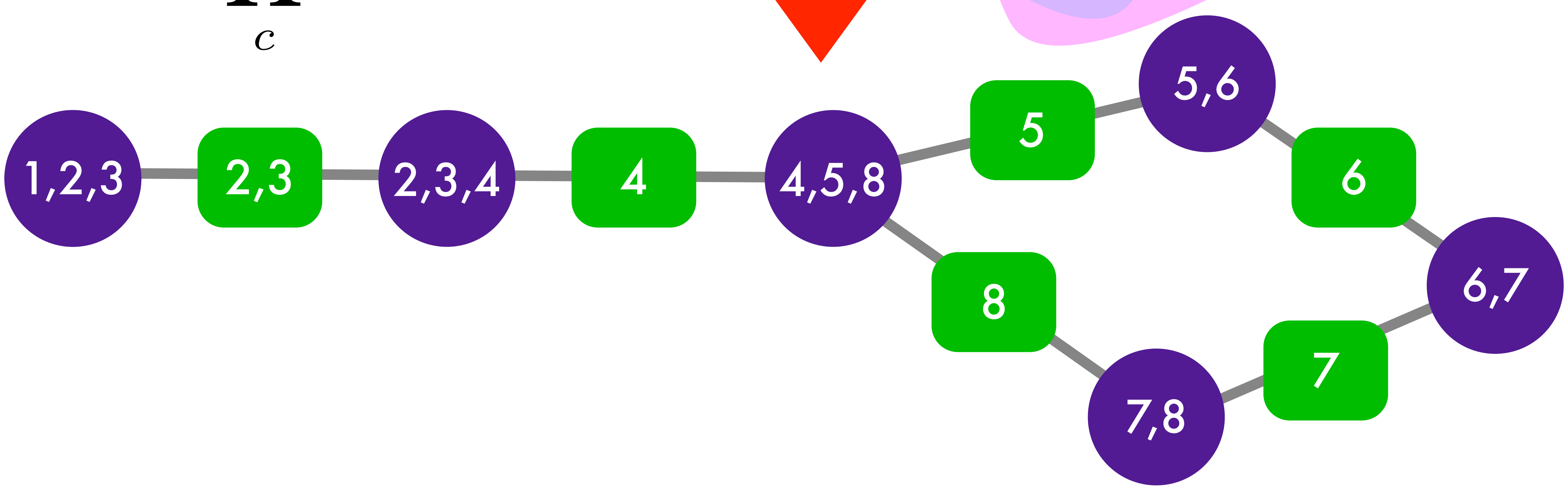


$$p(x) = \prod_c \psi_c(x_c)$$

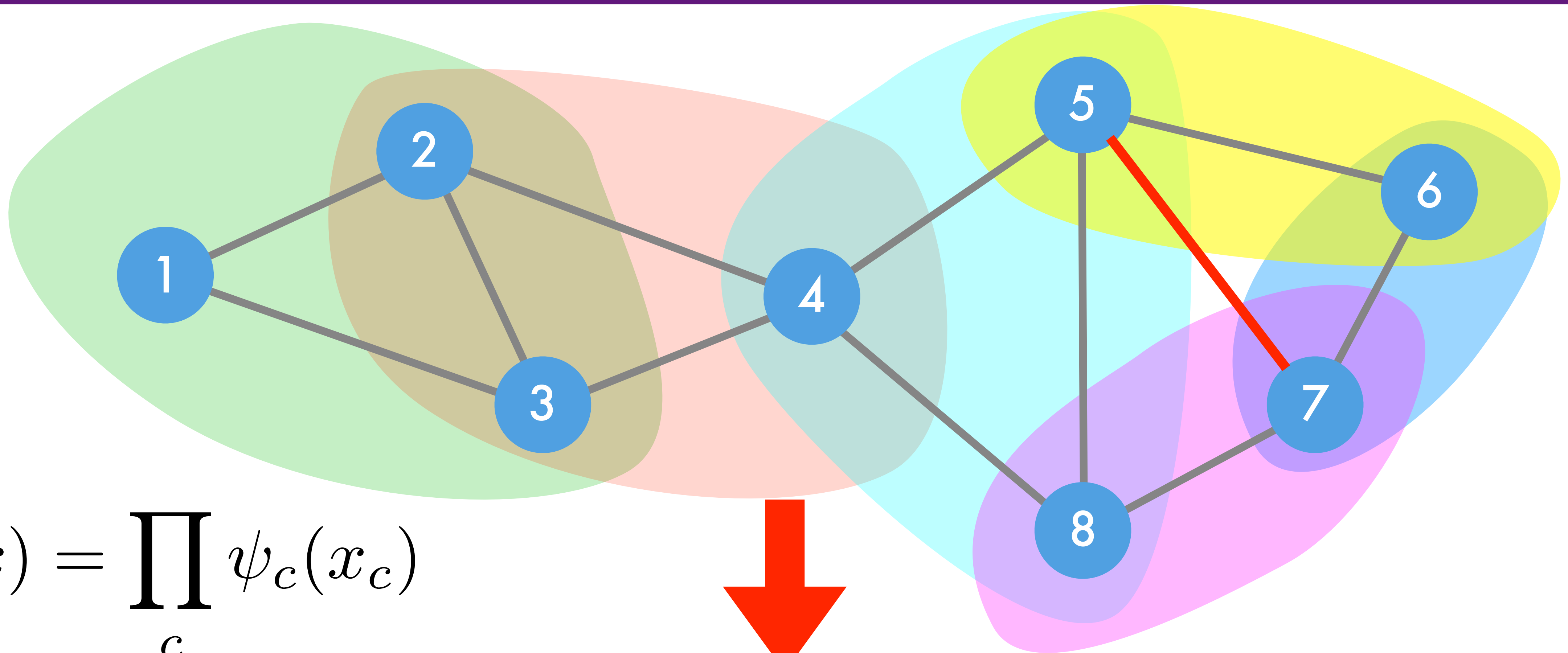
Junction Tree / Triangulation



$$p(x) = \prod_c \psi_c(x_c)$$



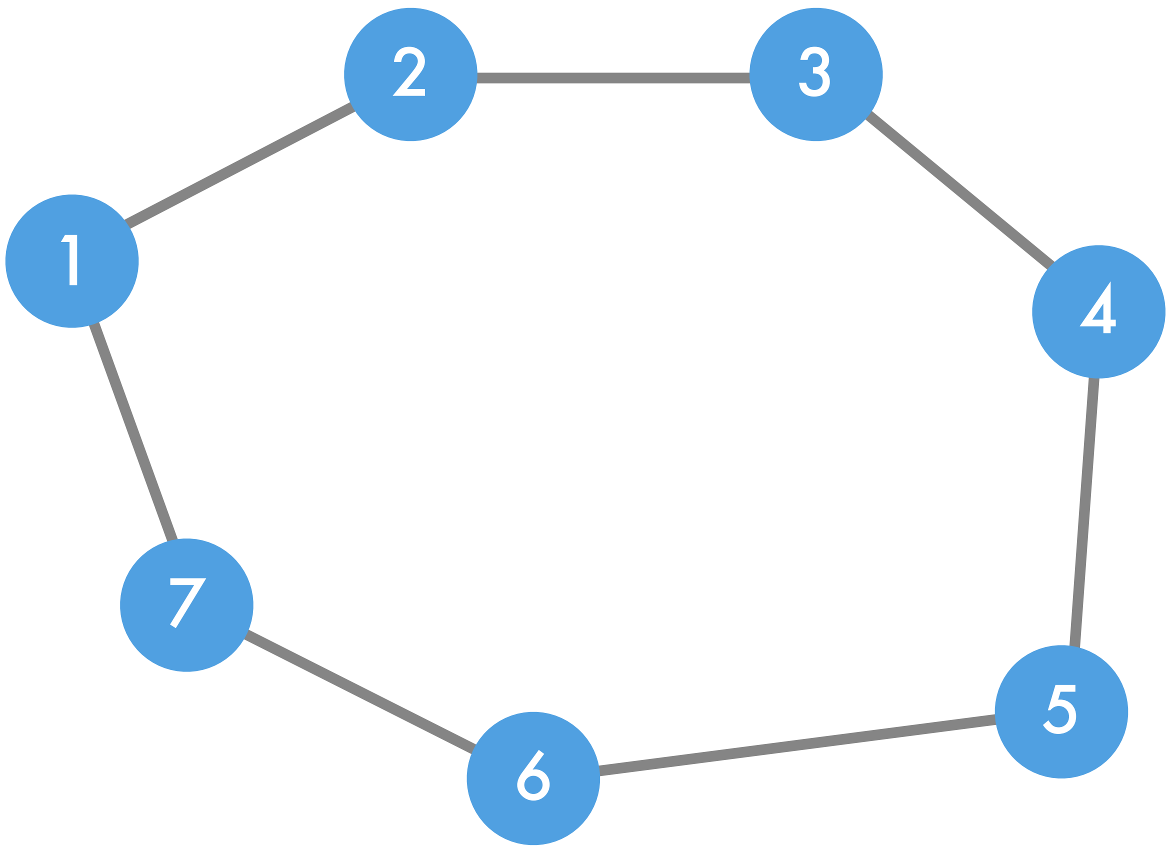
Junction Tree / Triangulation



$$p(x) = \prod_c \psi_c(x_c)$$

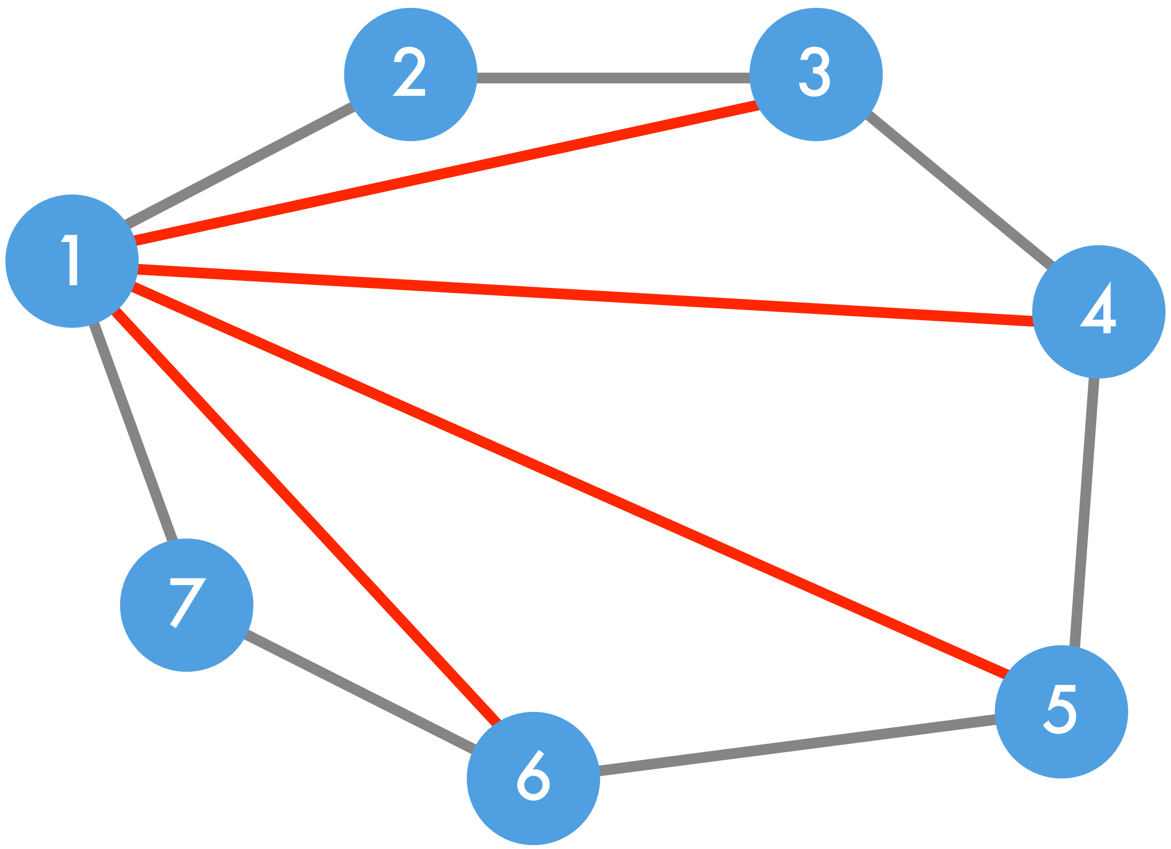
message passing possible

Triangulation Examples



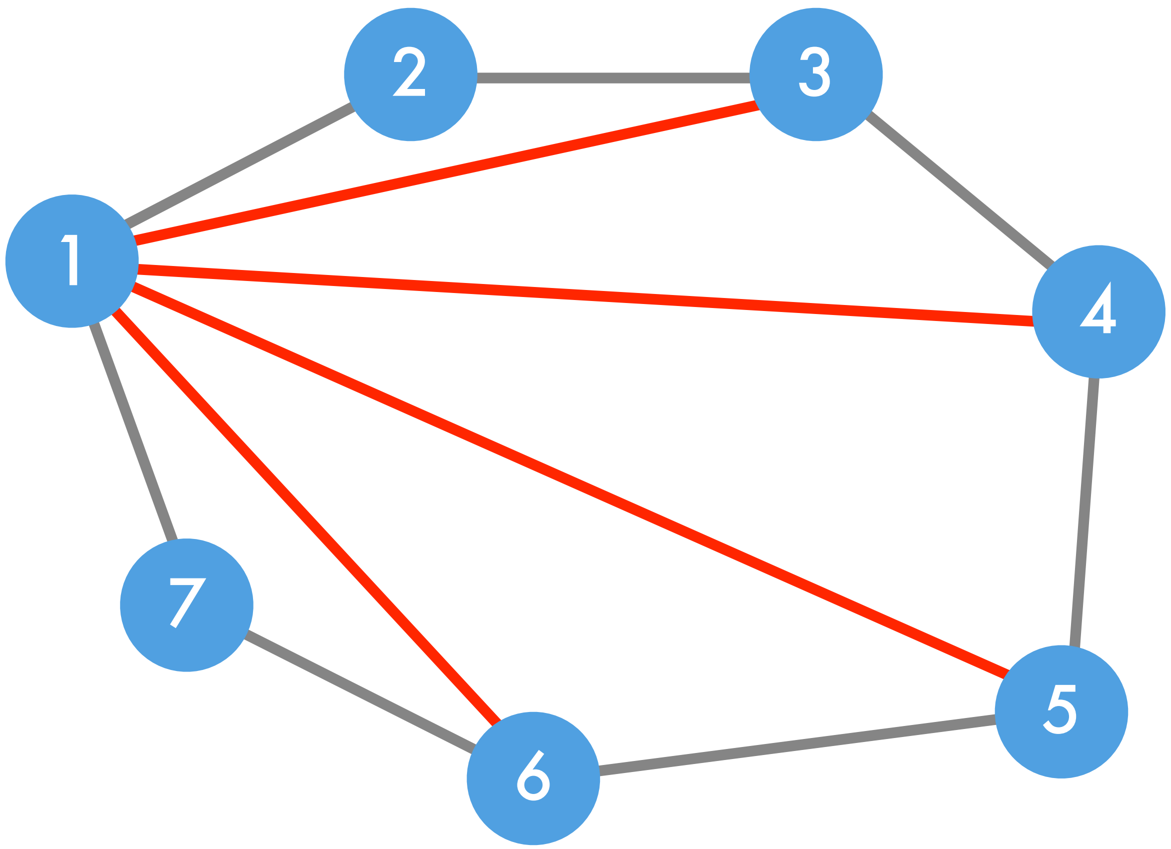
- Clique size increases
- Separator set size increases

Triangulation Examples



- Clique size increases
- Separator set size increases

Triangulation Examples



- Clique size increases
- Separator set size increases

Message Passing

- Joint Probability

$$p(x) \propto \psi(x_1, x_2, x_3)\psi(x_1, x_3, x_4)\psi(x_1, x_4, x_5)\psi(x_1, x_5, x_6)\psi(x_1, x_6, x_7)$$

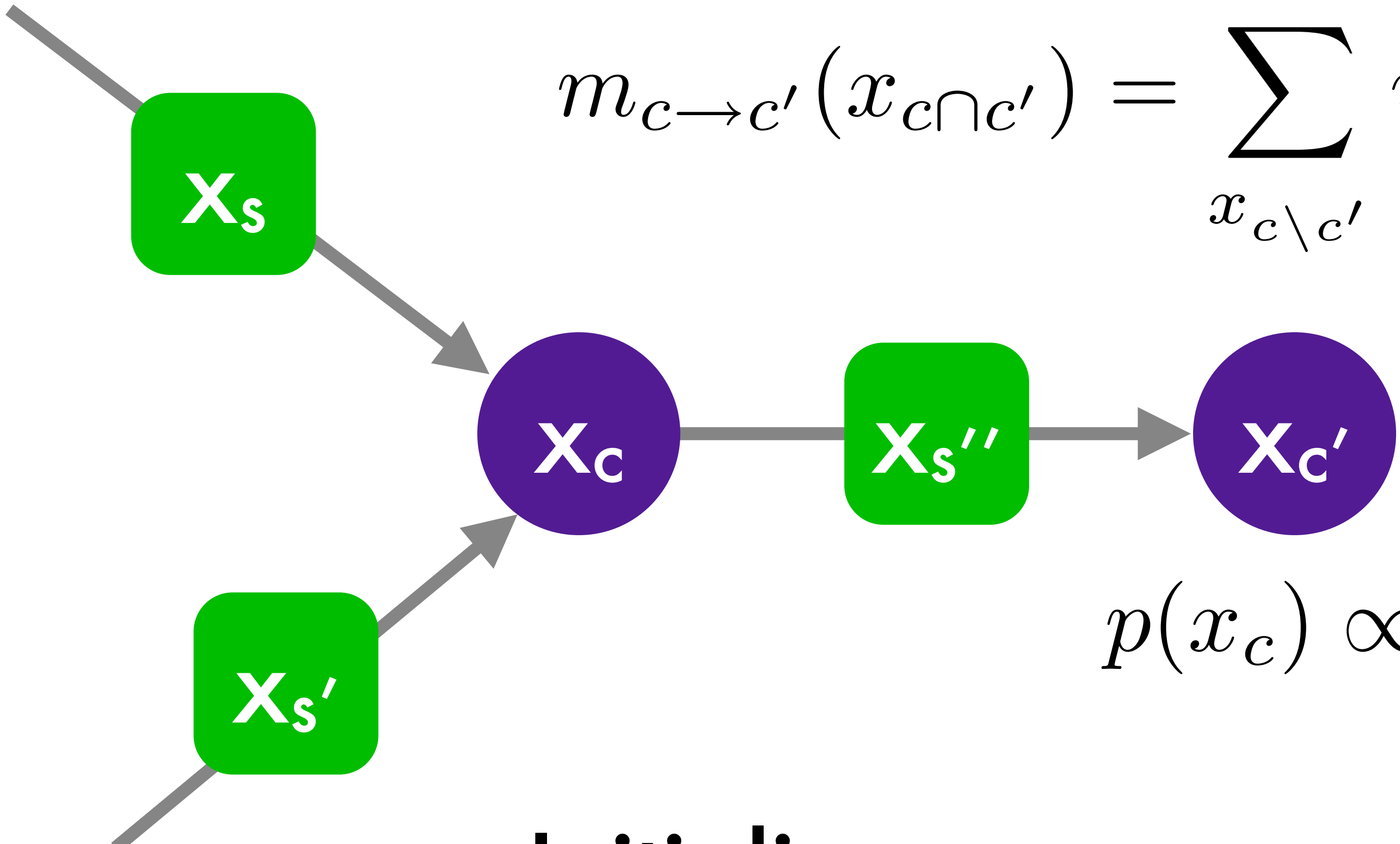
- Computing the normalization

$$m_{\rightarrow}(x_1, x_3) = \sum_{x_2} \psi(x_1, x_2, x_3)$$

$$m_{\rightarrow}(x_1, x_4) = \sum_{x_3} m_{\rightarrow}(x_1, x_3)\psi(x_1, x_3, x_4)$$

$$m_{\rightarrow}(x_1, x_5) = \sum_{x_4} m_{\rightarrow}(x_1, x_4)\psi(x_1, x_4, x_5)$$

Message Passing



$$m_{c\to c'}(x_{c\cap c'}) = \sum_{x_{c\setminus c'}} \psi_c(x_c) \prod_{c''\in N(c)\setminus c'} m_{c''\to c}(x_{c\cap c''})$$

$$p(x_c) \propto \psi_c(x_c) \prod_{c''\in N(c)} m_{c''\to c}(x_{c\cap c''})$$

- Initialize messages with 1
- Guaranteed to converge for (junction) trees
- Works well in practice even for loopy graphs
- Only local computations are required

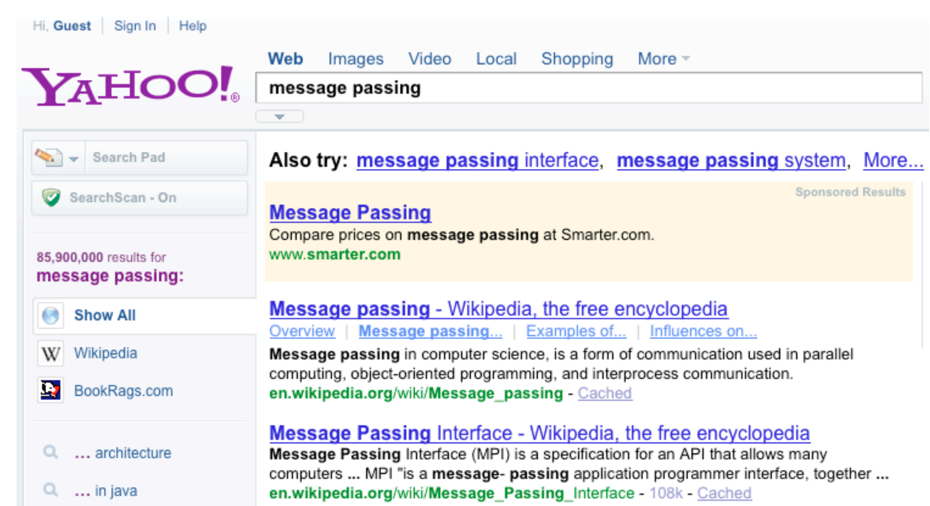
Message Passing in Practice

- Incoming messages contain aggregate uncertainty from neighboring random variables
- Message passing combines and transmits this information in both directions

crawler

phase 1 ranker

tiering



Message Passing in Practice

- Incoming messages contain aggregate uncertainty from neighboring random variables
- Message passing combines and transmits this information in both directions

Part 5 - Scalable Topic Models

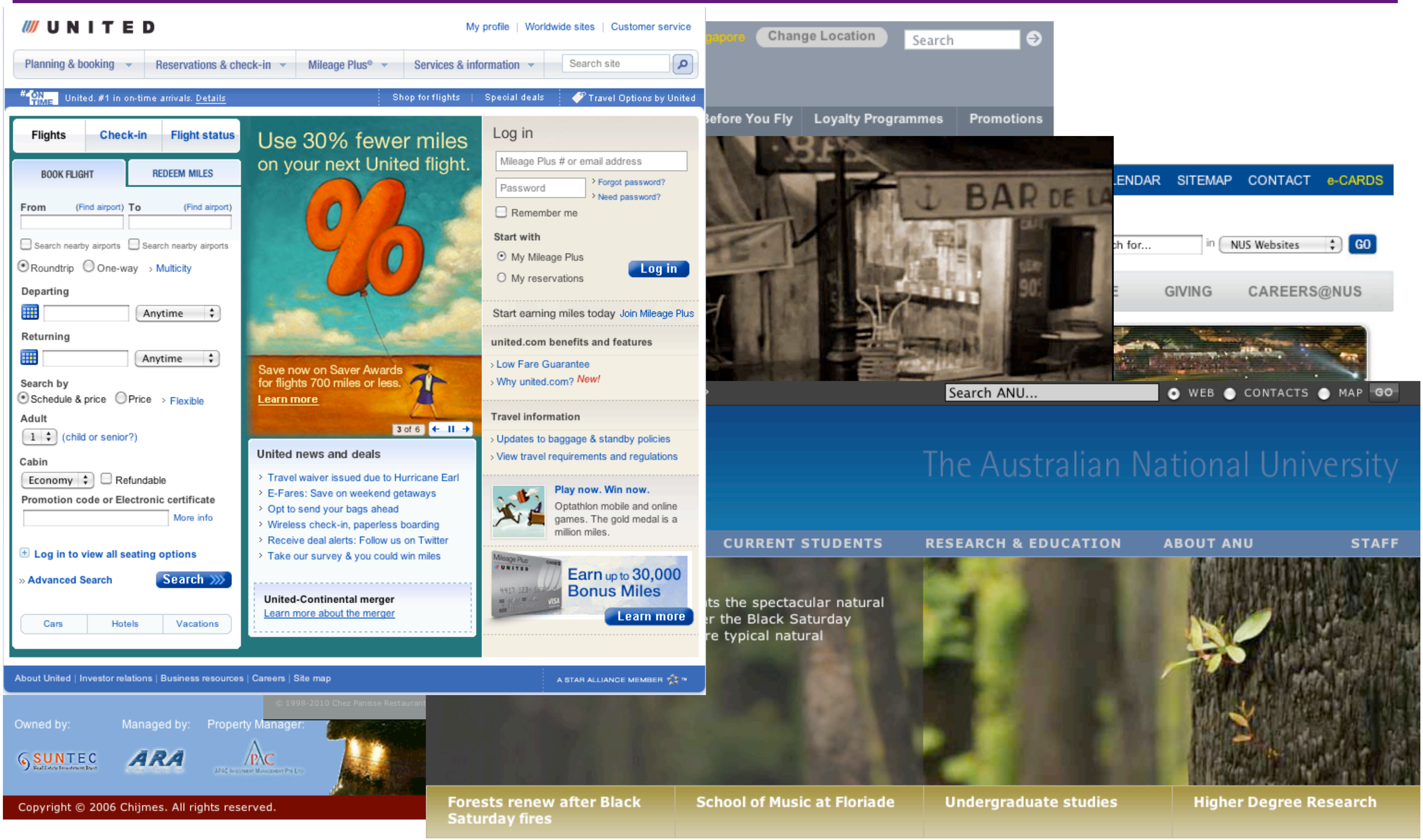
Topic models

Grouping objects

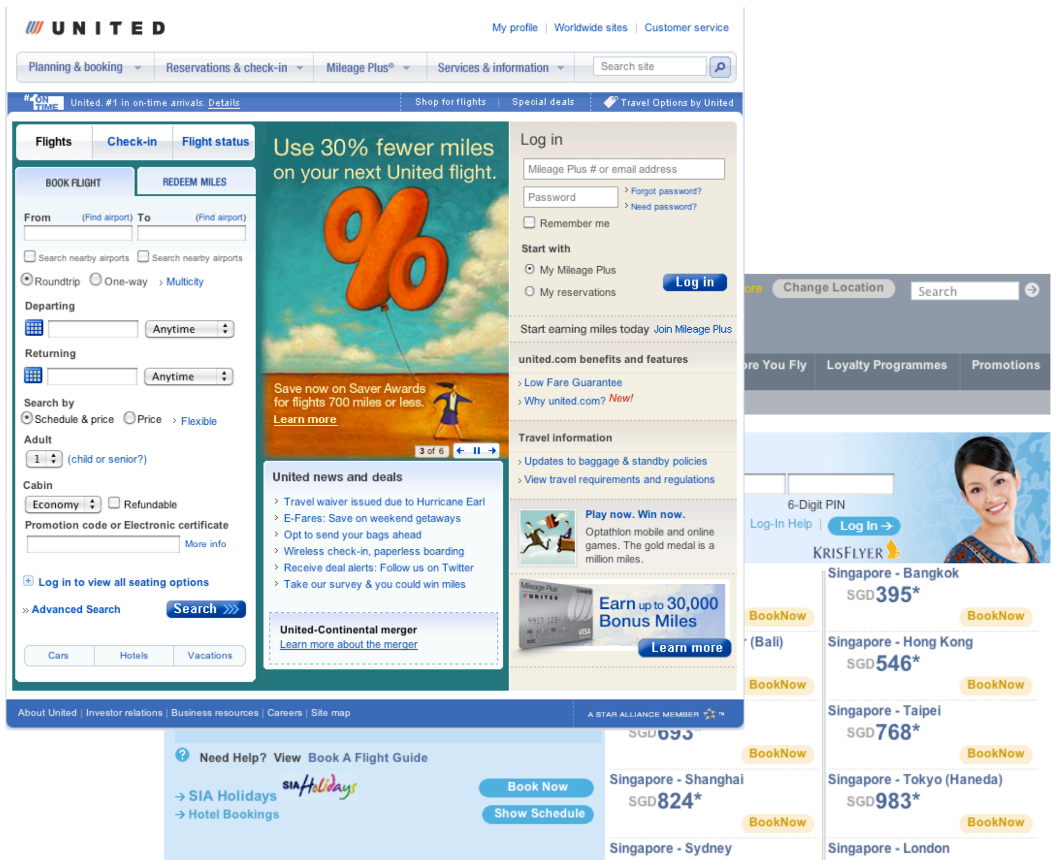
Grouping objects

YAHOO!

Grouping objects

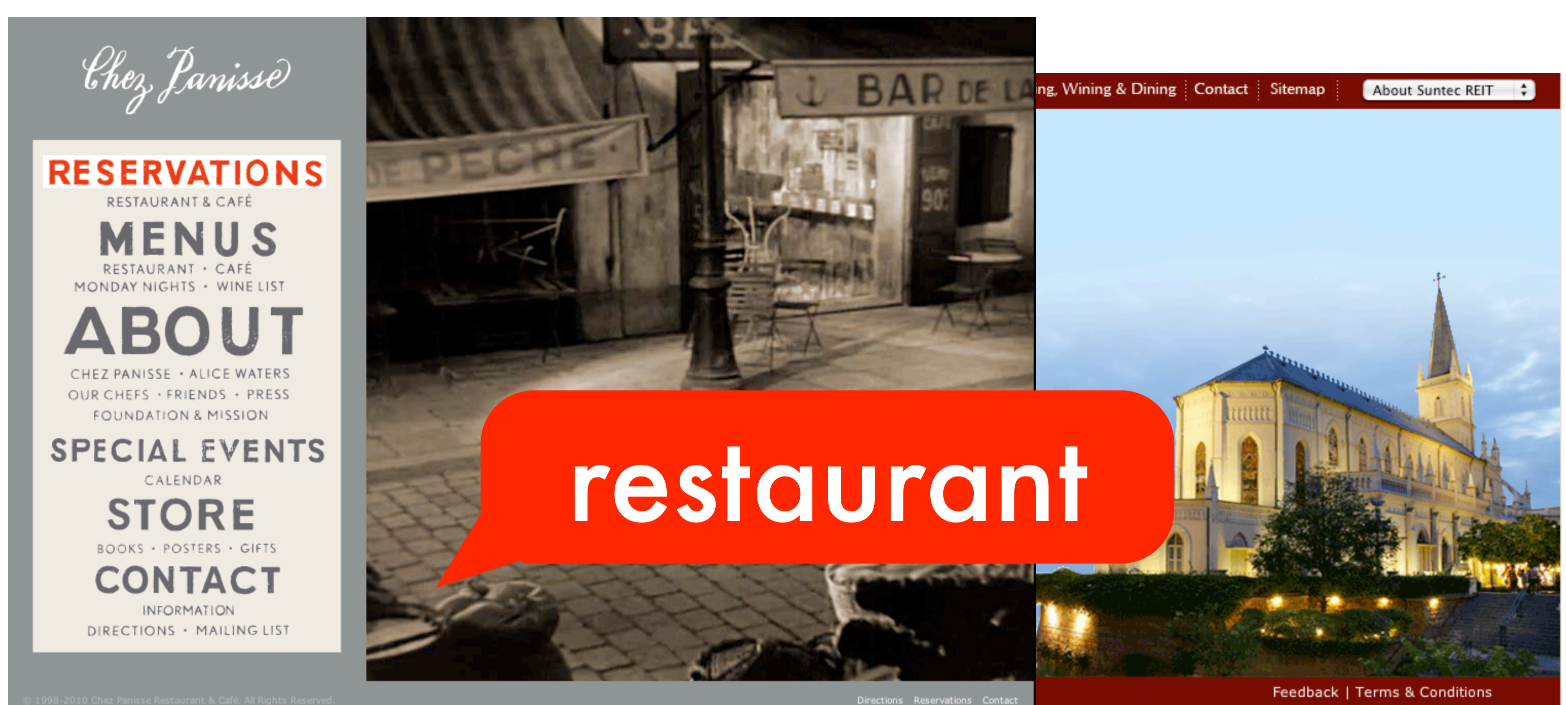


Grouping objects



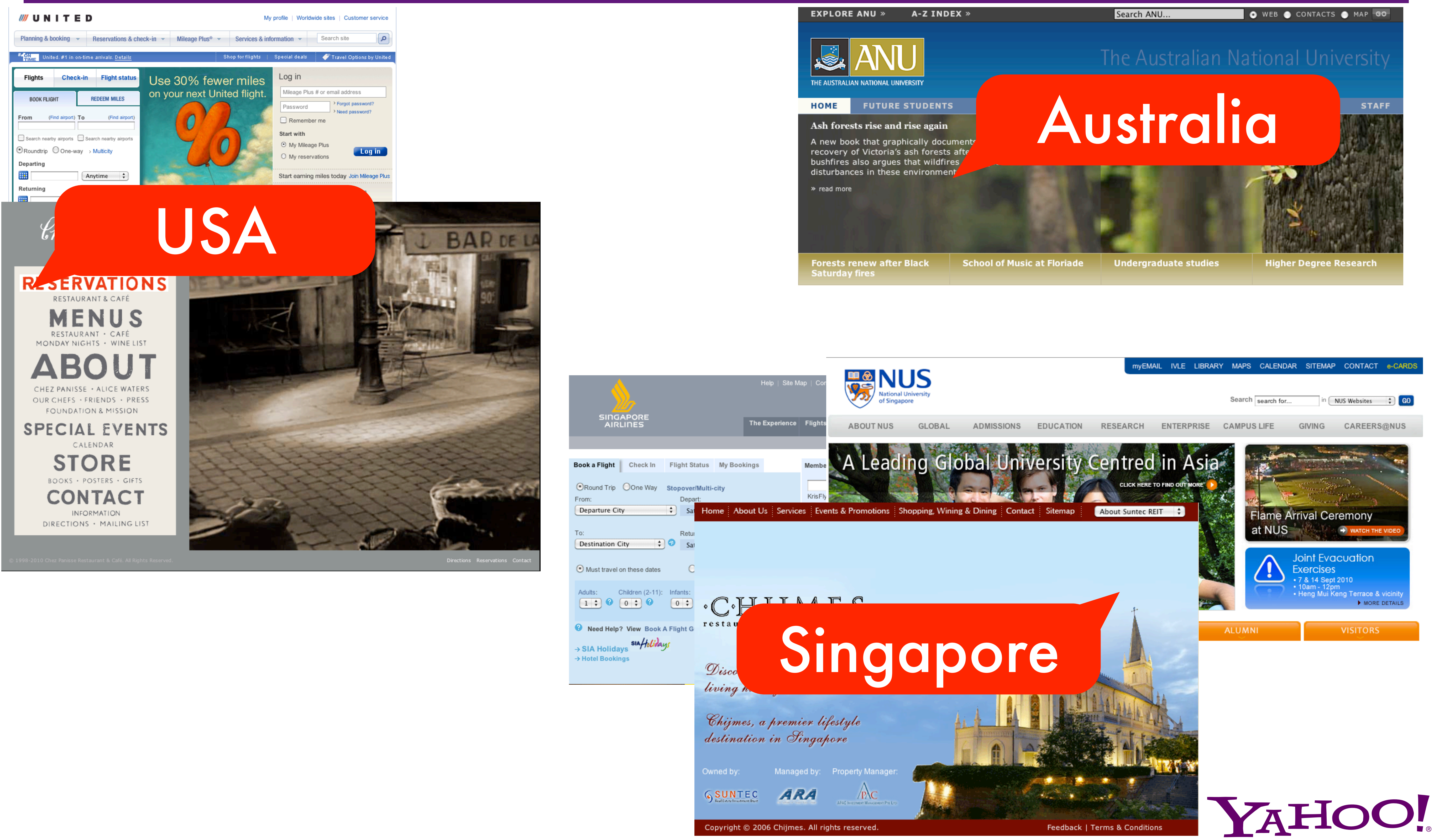
YAHOO!

Grouping objects

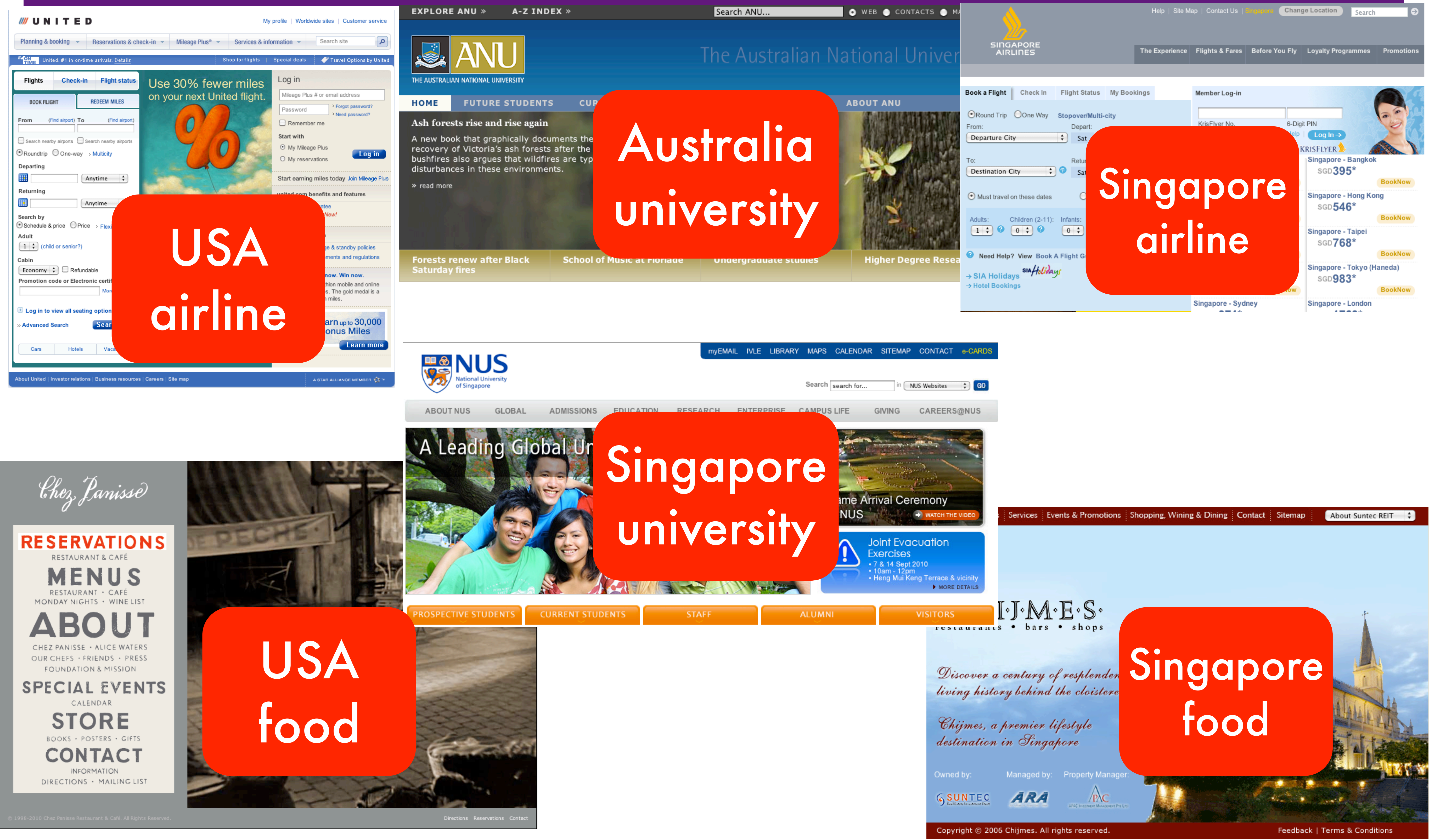


YAHOO!

Grouping objects

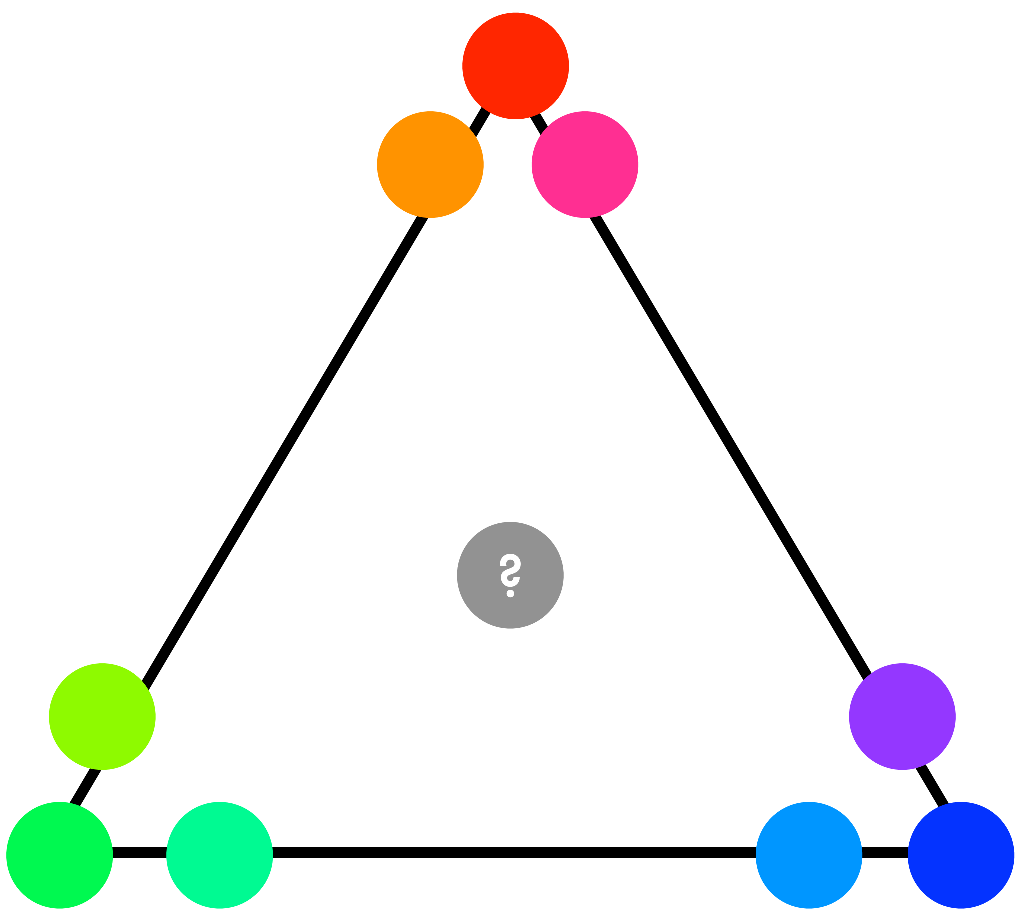


Topic Models



Clustering & Topic Models

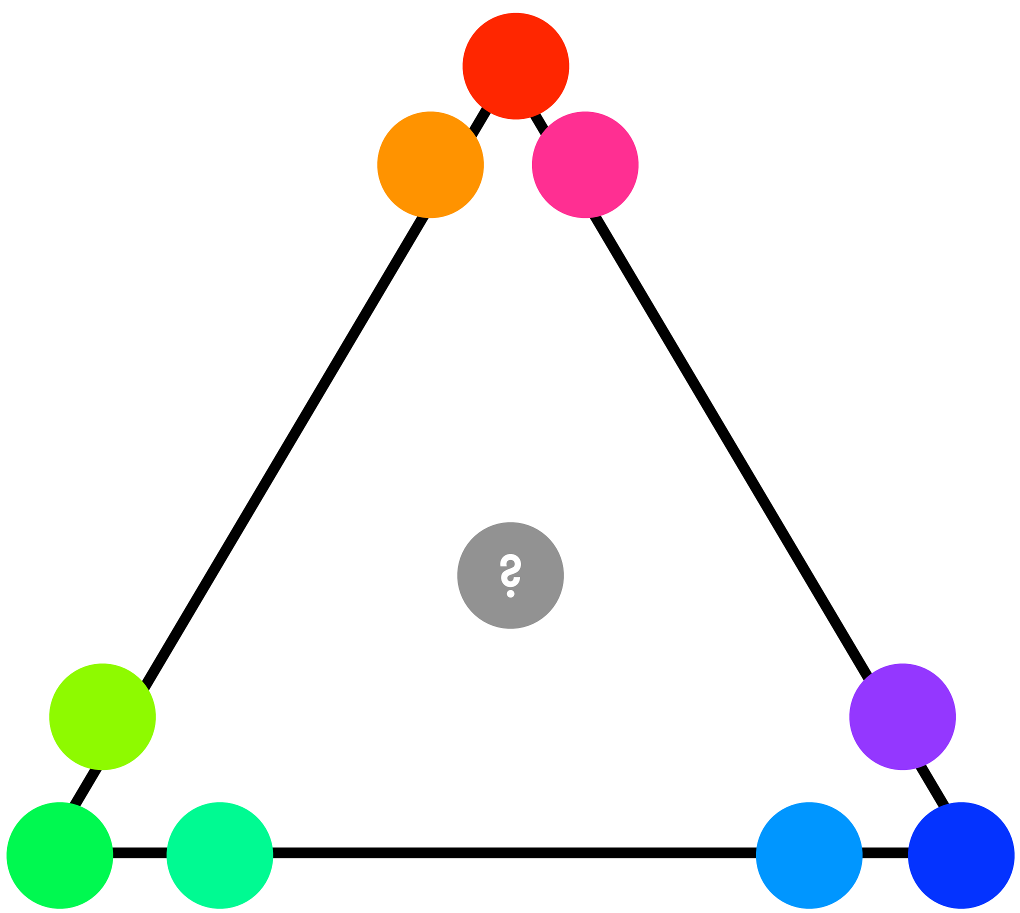
Clustering



group objects
by prototypes

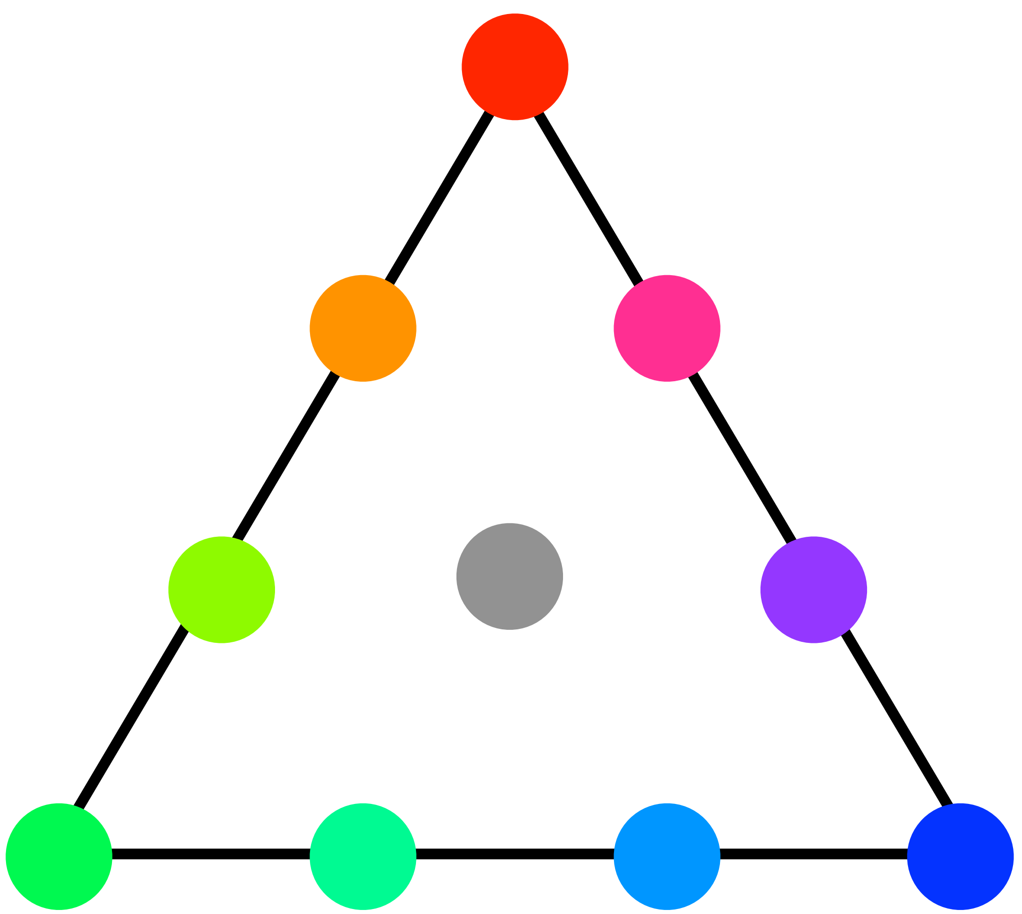
Clustering & Topic Models

Clustering



group objects
by prototypes

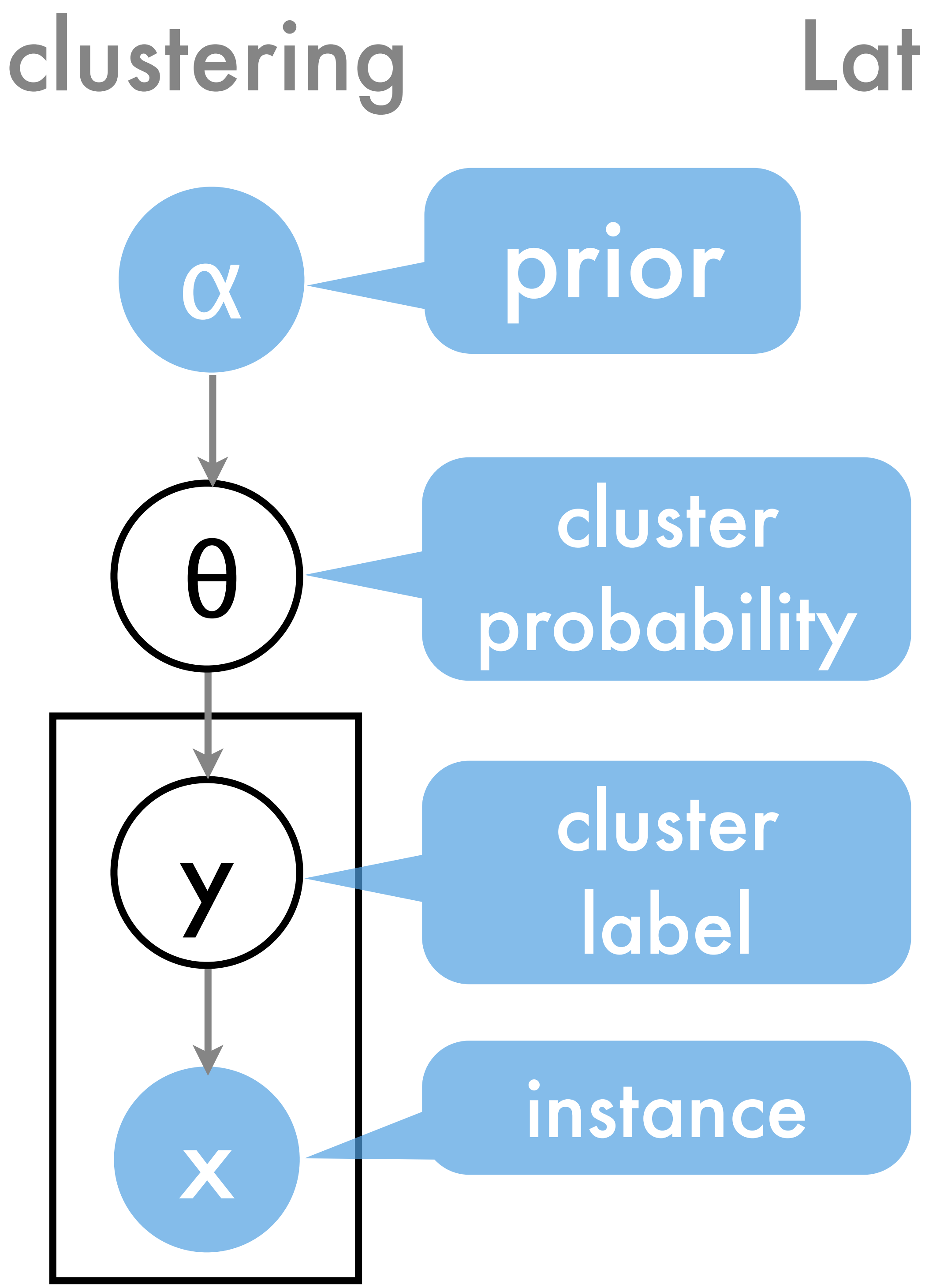
Topics

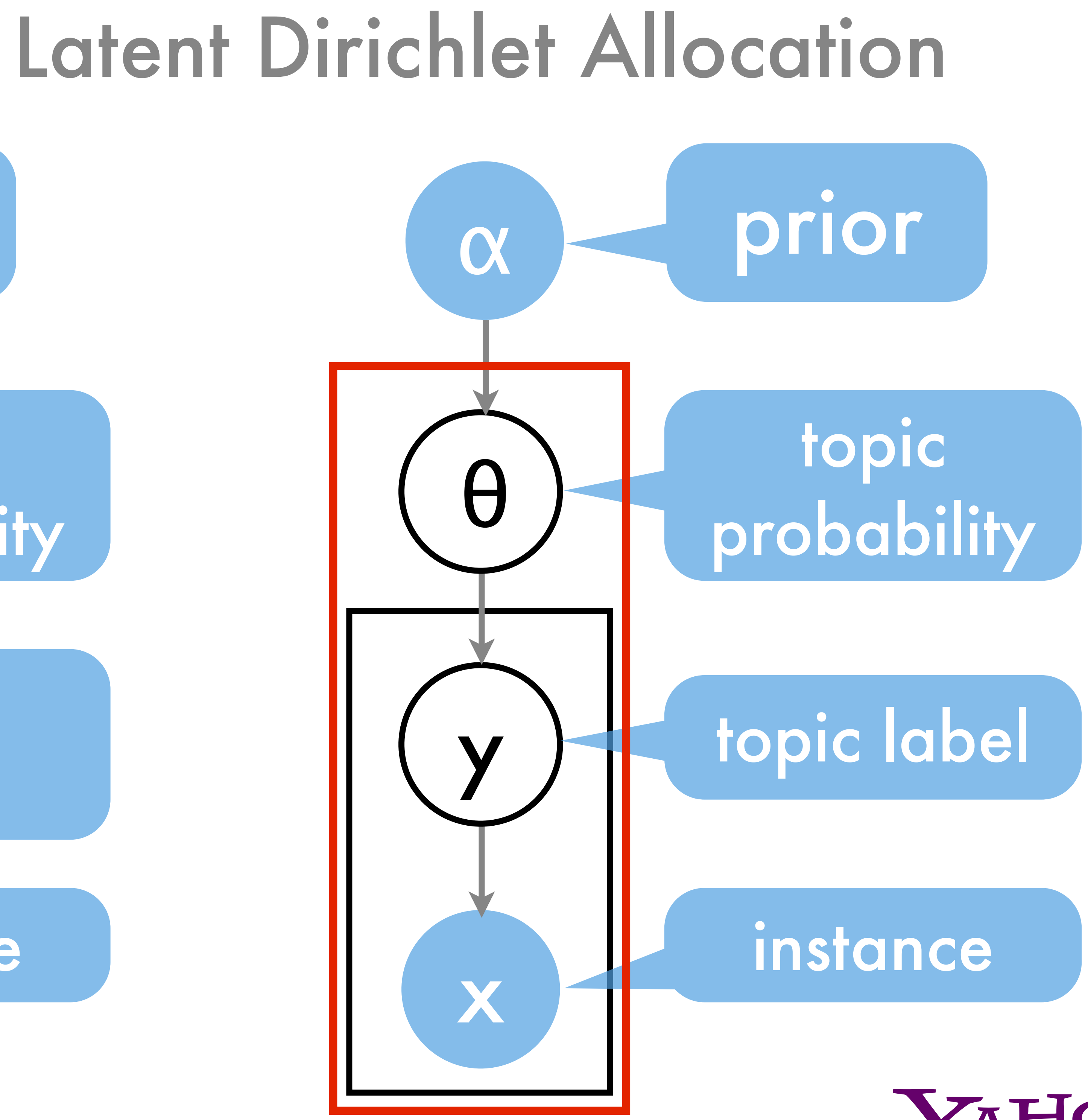


decompose objects
into prototypes

YAHOO!

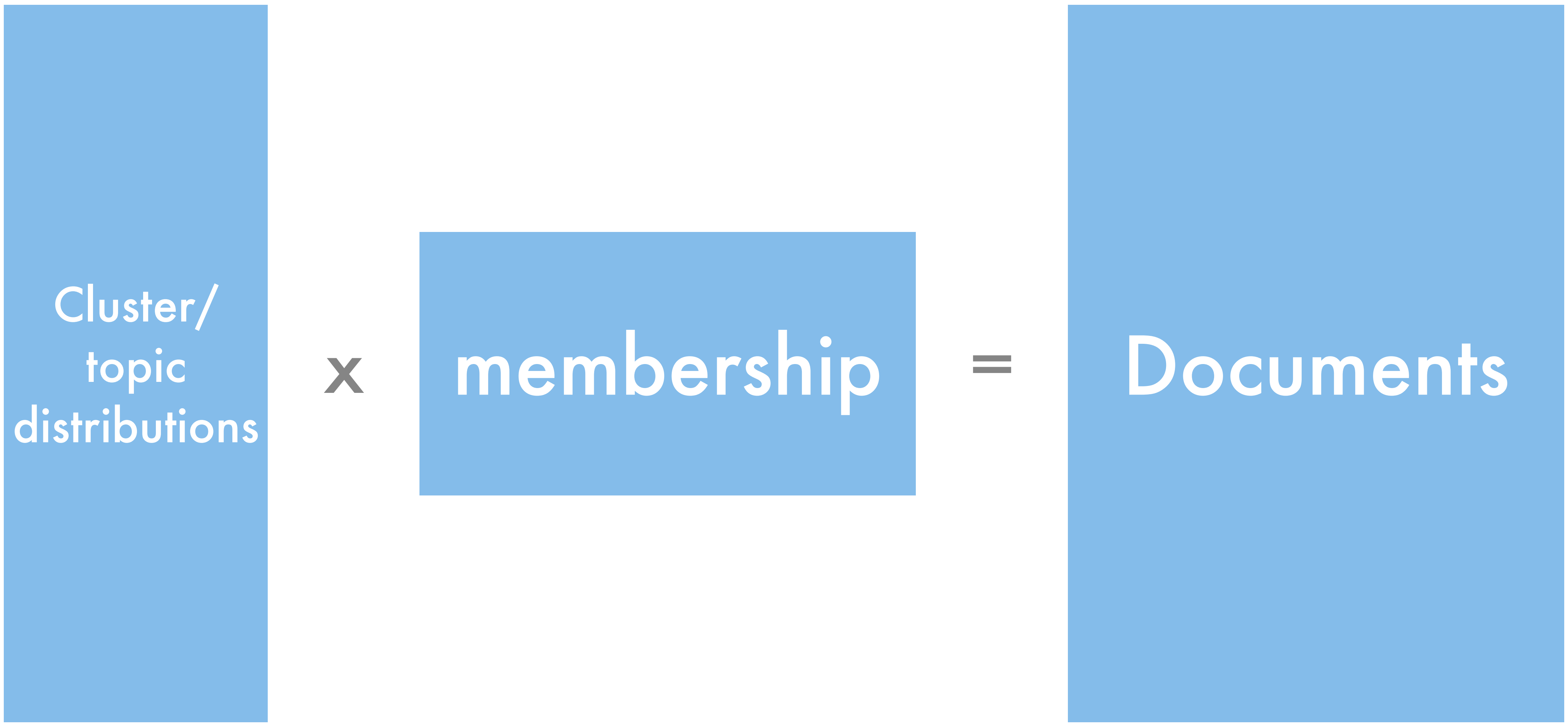
Clustering & Topic Models





YAHOO!

Clustering & Topic Models



clustering: (0, 1) matrix
topic model: stochastic matrix
LSI: arbitrary matrices

Topics in text

The William Randolph Hearst Foundation will give $1.25 million to Lincoln Center, Metropolitan Opera Co., New York Philharmonic and Juilliard School. "Our board felt that we had a real opportunity to make a mark on the future of the performing arts with these grants an act every bit as important as our traditional areas of support in health, medical research, education and the social services," Hearst Foundation President Randolph A. Hearst said Monday in announcing the grants. Lincoln Center's share will be $200,000 for its new building, which will house young artists and provide new public facilities. The Metropolitan Opera Co. and New York Philharmonic will receive $400,000 each. The Juilliard School, where music and the performing arts are taught, will get $250,000. The Hearst Foundation, a leading supporter of the Lincoln Center Consolidated Corporate Fund, will make its usual annual $100,000 donation, too.

Latent Dirichlet Allocation; Blei, Ng, Jordan, JMLR 2003

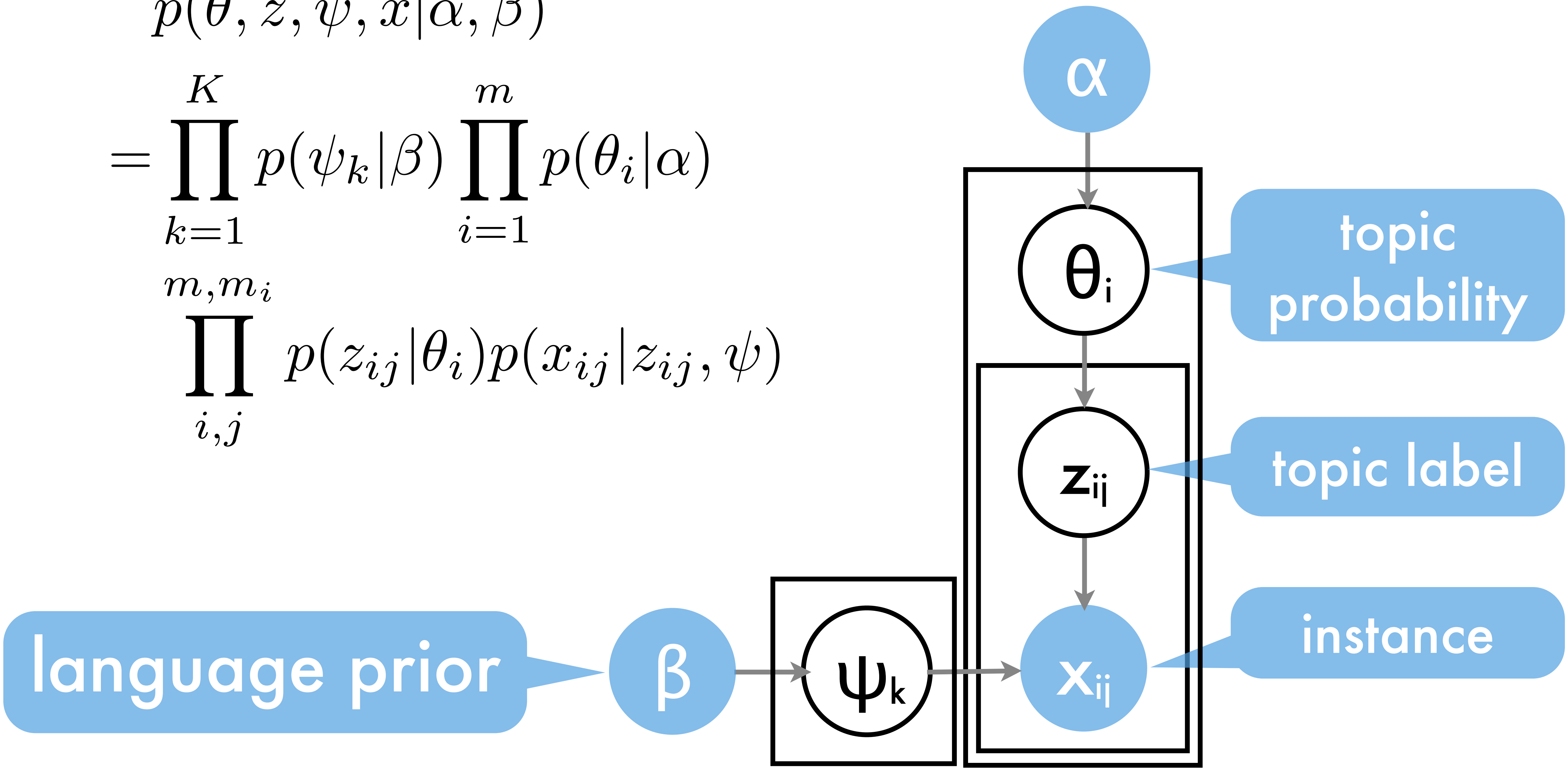
Collapsed Gibbs Sampler

Joint Probability Distribution

$$p(\theta, z, \psi, x | \alpha, \beta)$$

$$= \prod_{k=1}^{K} p(\psi_k | \beta) \prod_{i=1}^{m} p(\theta_i | \alpha)$$

$$\prod_{i,j}^{m, m_i} p(z_{ij} | \theta_i) p(x_{ij} | z_{ij}, \psi)$$



Joint Probability Distribution

sample Ψ independently

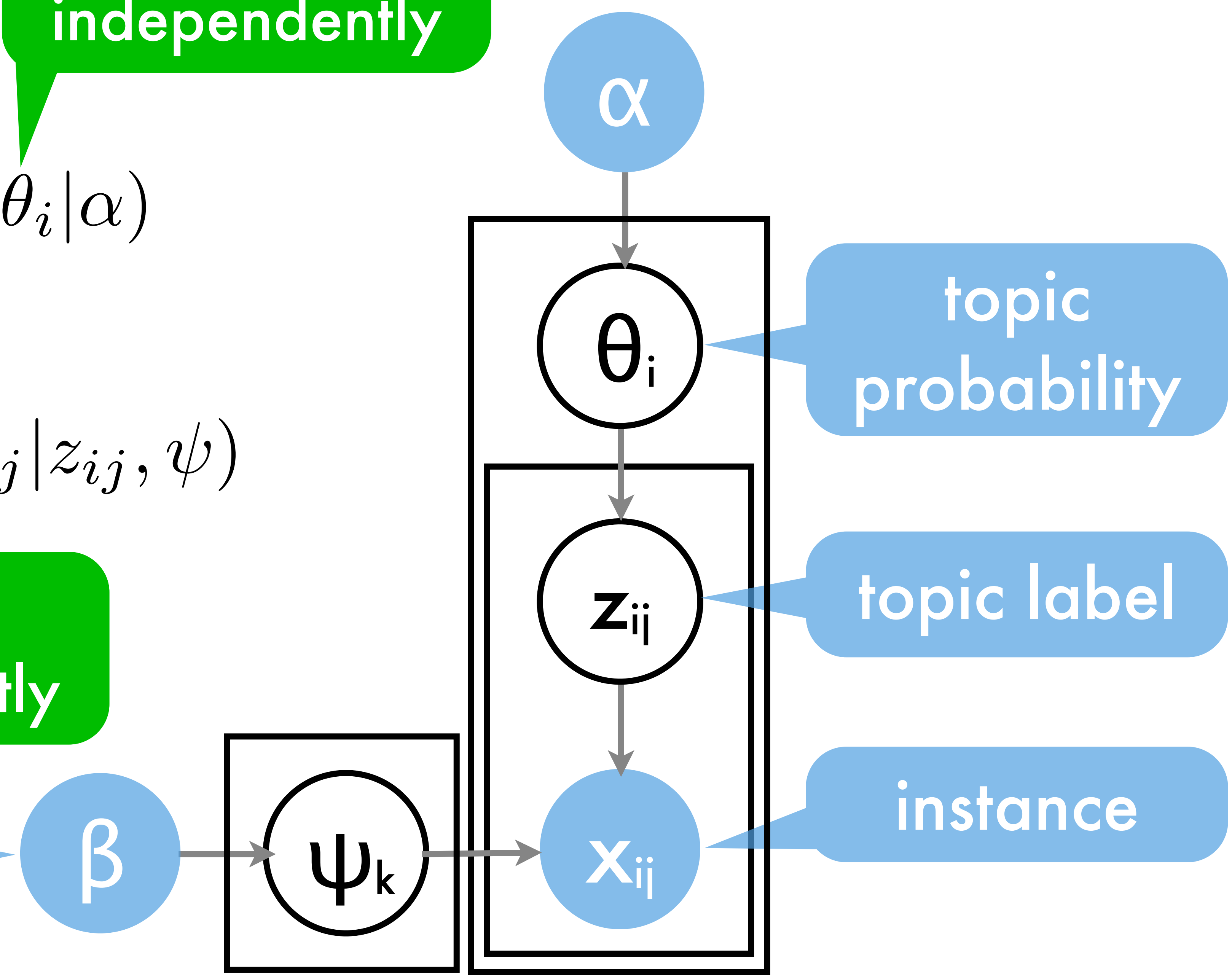
sample θ independently

$$p(\theta, z, \psi, x|\alpha, \beta)$$

$$= \prod_{k=1}^{K} p(\psi_k|\beta) \prod_{i=1}^{m} p(\theta_i|\alpha)$$

$$\prod_{i,j}^{m,m_i} p(z_{ij}|\theta_i)p(x_{ij}|z_{ij}, \psi)$$

sample z independently



Joint Probability Distribution

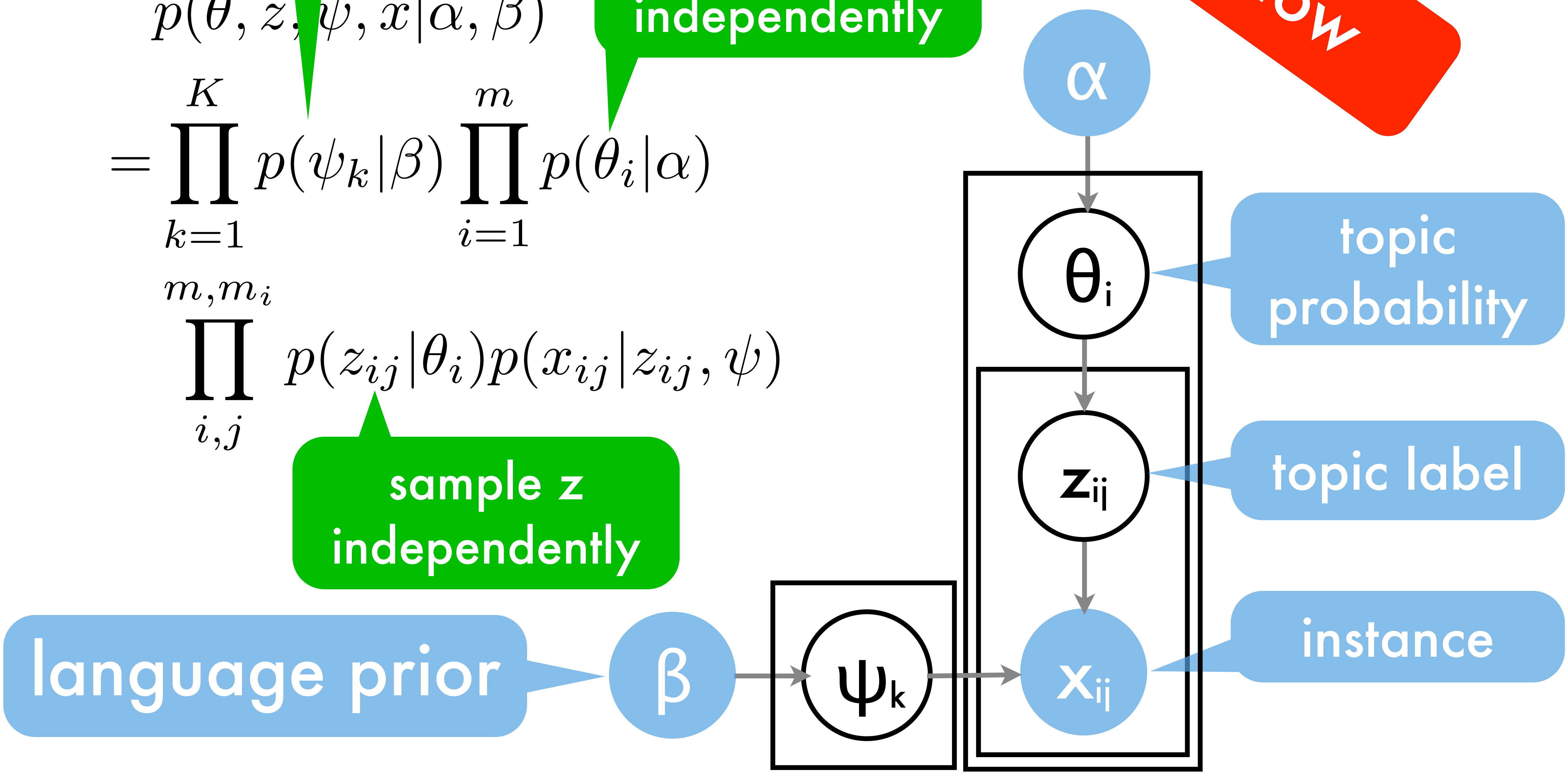
sample Ψ independently

sample θ independently

$$p(\theta, z, \psi, x|\alpha, \beta)$$
$$= \prod_{k=1}^{K} p(\psi_k|\beta) \prod_{i=1}^{m} p(\theta_i|\alpha)$$
$$\prod_{i,j}^{m,m_i} p(z_{ij}|\theta_i) p(x_{ij}|z_{ij}, \psi)$$

sample z independently

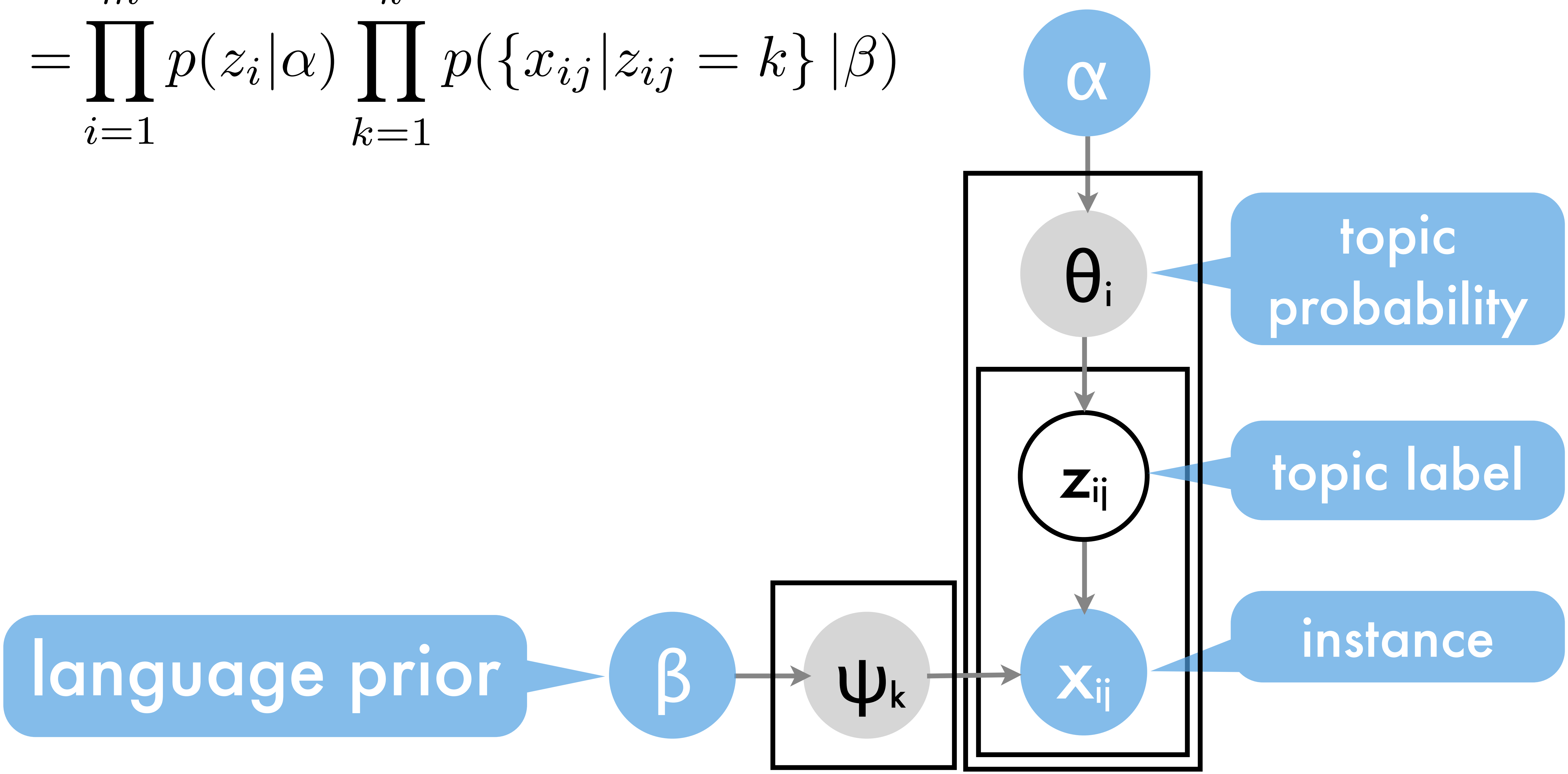
slow



Collapsed Sampler

$$p(z, x|\alpha, \beta)$$

$$= \prod_{i=1}^{m} p(z_i|\alpha) \prod_{k=1}^{k} p(\{x_{ij}|z_{ij} = k\} \,|\beta)$$

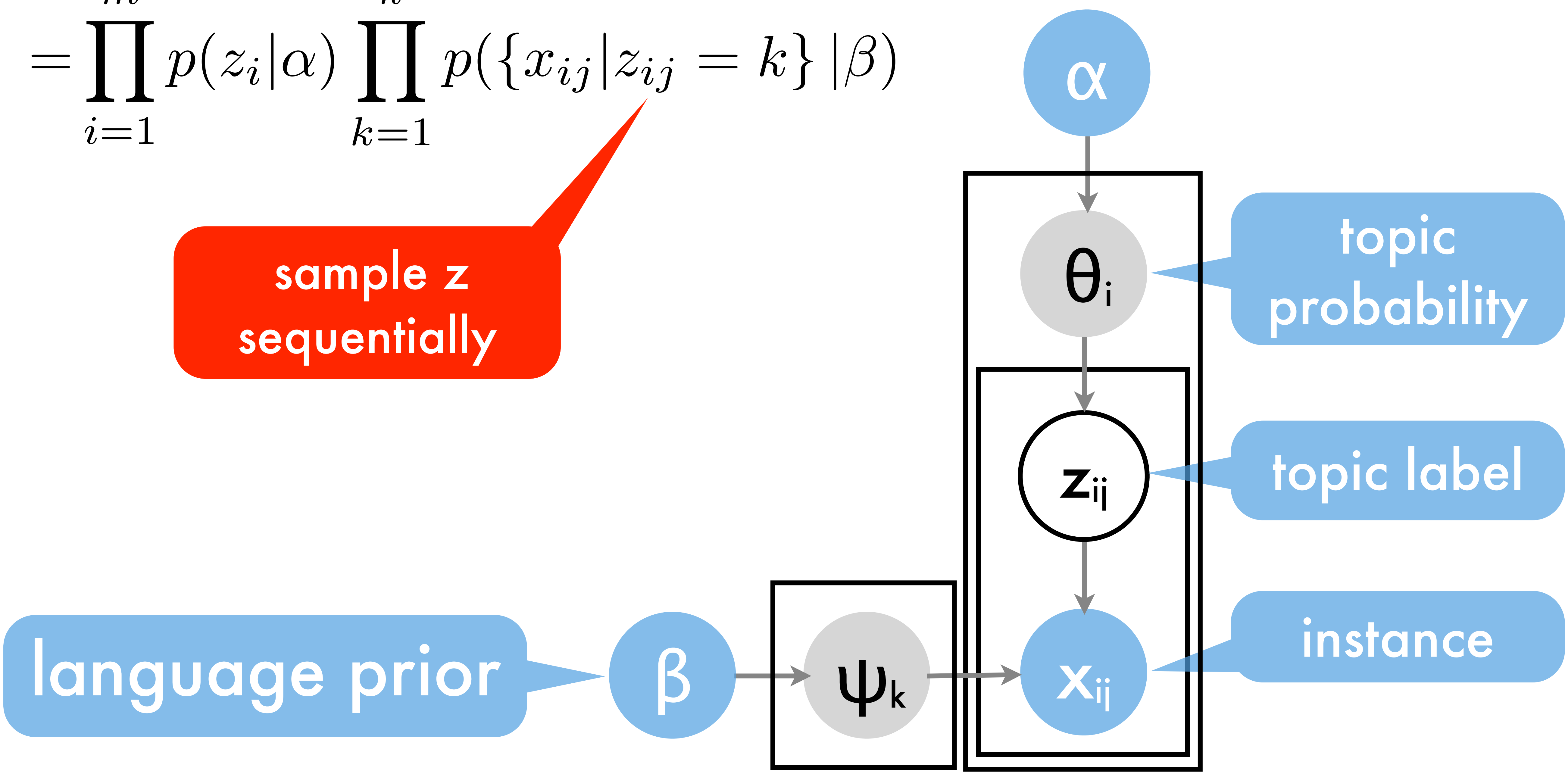


Collapsed Sampler

$$p(z, x|\alpha, \beta)$$

$$= \prod_{i=1}^{m} p(z_i|\alpha) \prod_{k=1}^{k} p(\{x_{ij}|z_{ij} = k\}\,|\beta)$$

sample z
sequentially



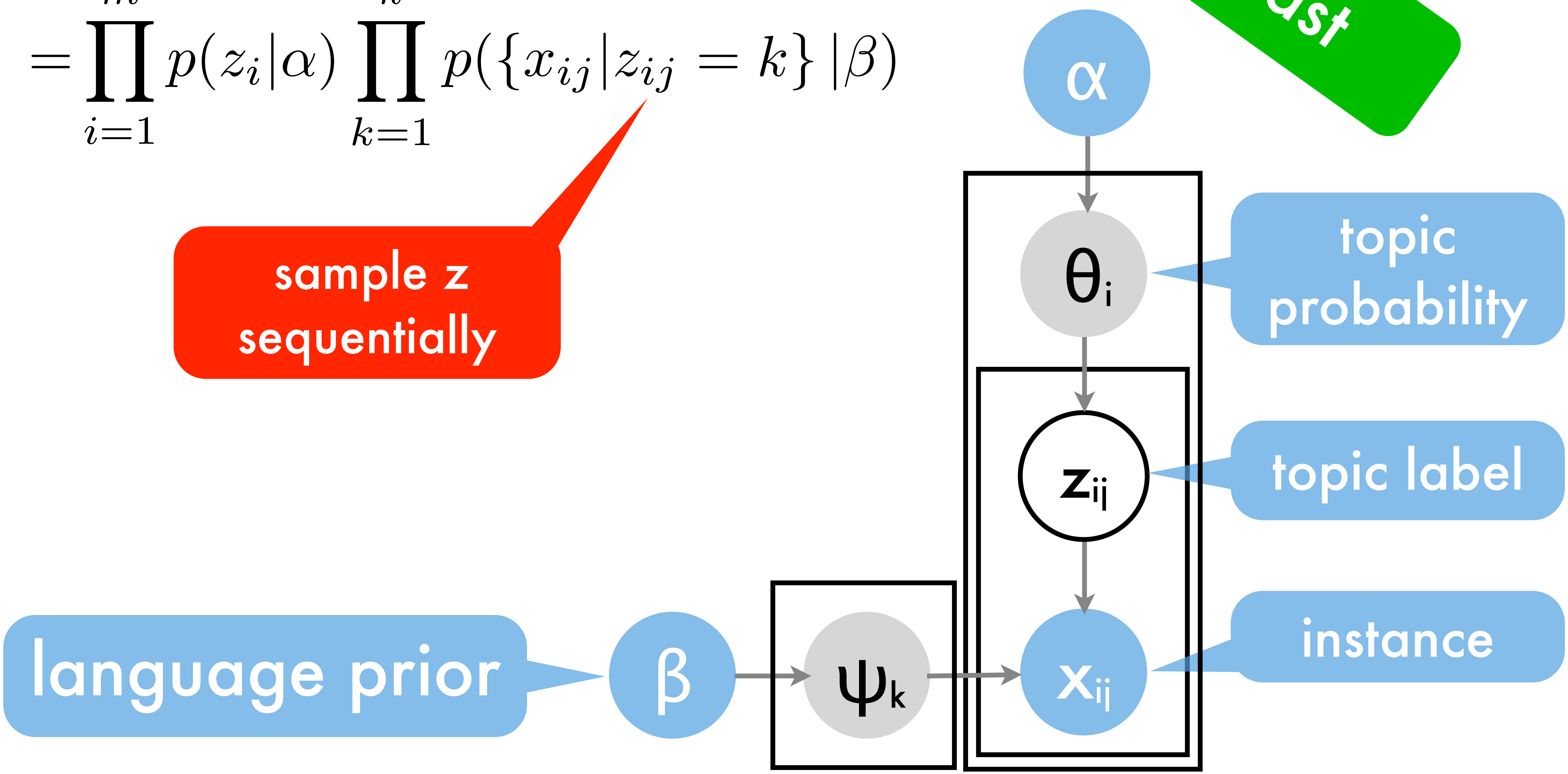
Collapsed Sampler

$$p(z, x|\alpha, \beta)$$

$$= \prod_{i=1}^{m} p(z_i|\alpha) \prod_{k=1}^{k} p(\{x_{ij}|z_{ij} = k\} |\beta)$$

sample z sequentially

fast



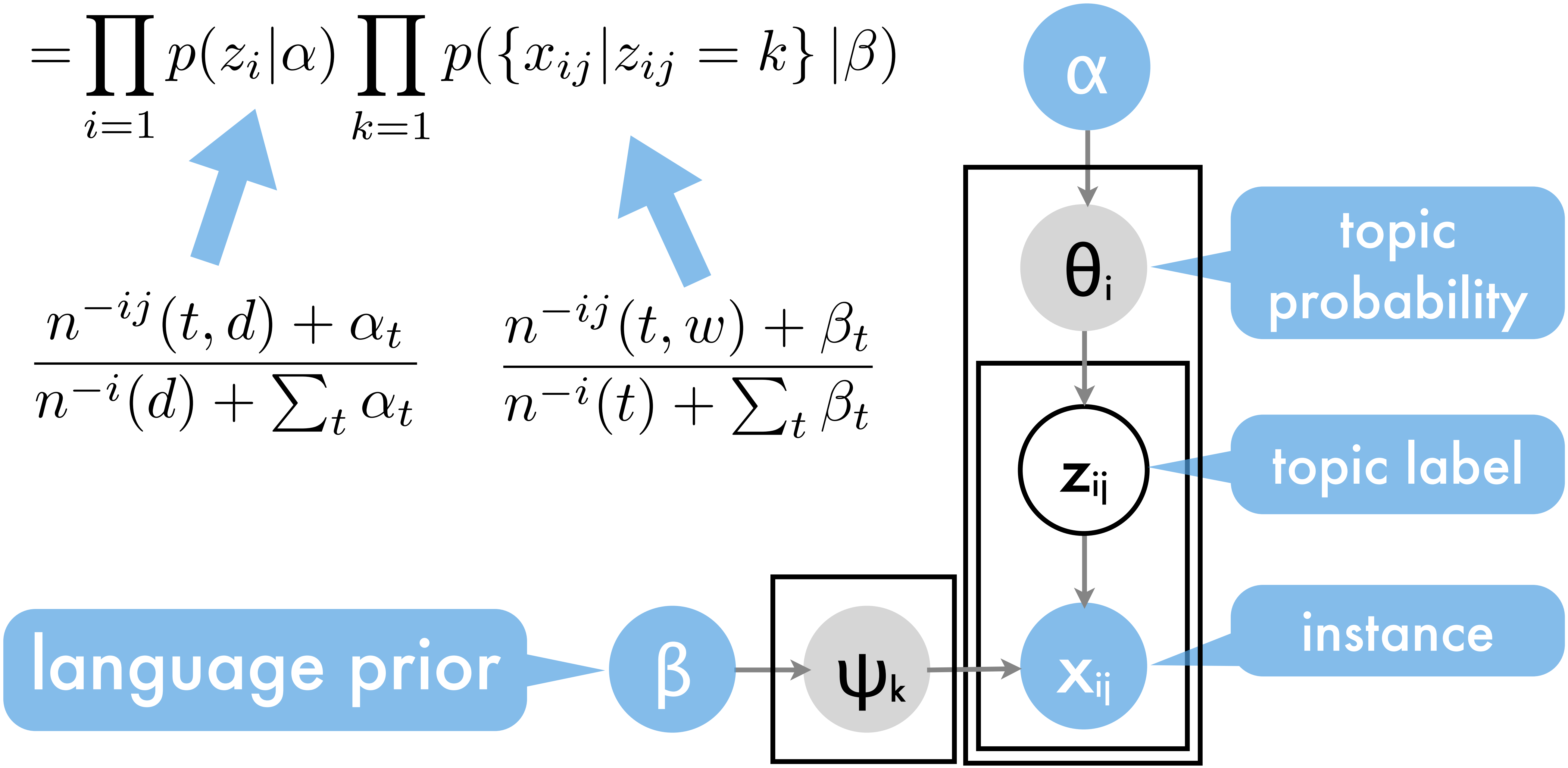
Collapsed Sampler

Griffiths & Steyvers, 2005

$$p(z, x|\alpha, \beta)$$

$$= \prod_{i=1}^{m} p(z_i|\alpha) \prod_{k=1}^{k} p(\{x_{ij}|z_{ij} = k\} |\beta)$$

$$\frac{n^{-ij}(t,d) + \alpha_t}{n^{-i}(d) + \sum_t \alpha_t} \qquad \frac{n^{-ij}(t,w) + \beta_t}{n^{-i}(t) + \sum_t \beta_t}$$



Collapsed Sampler

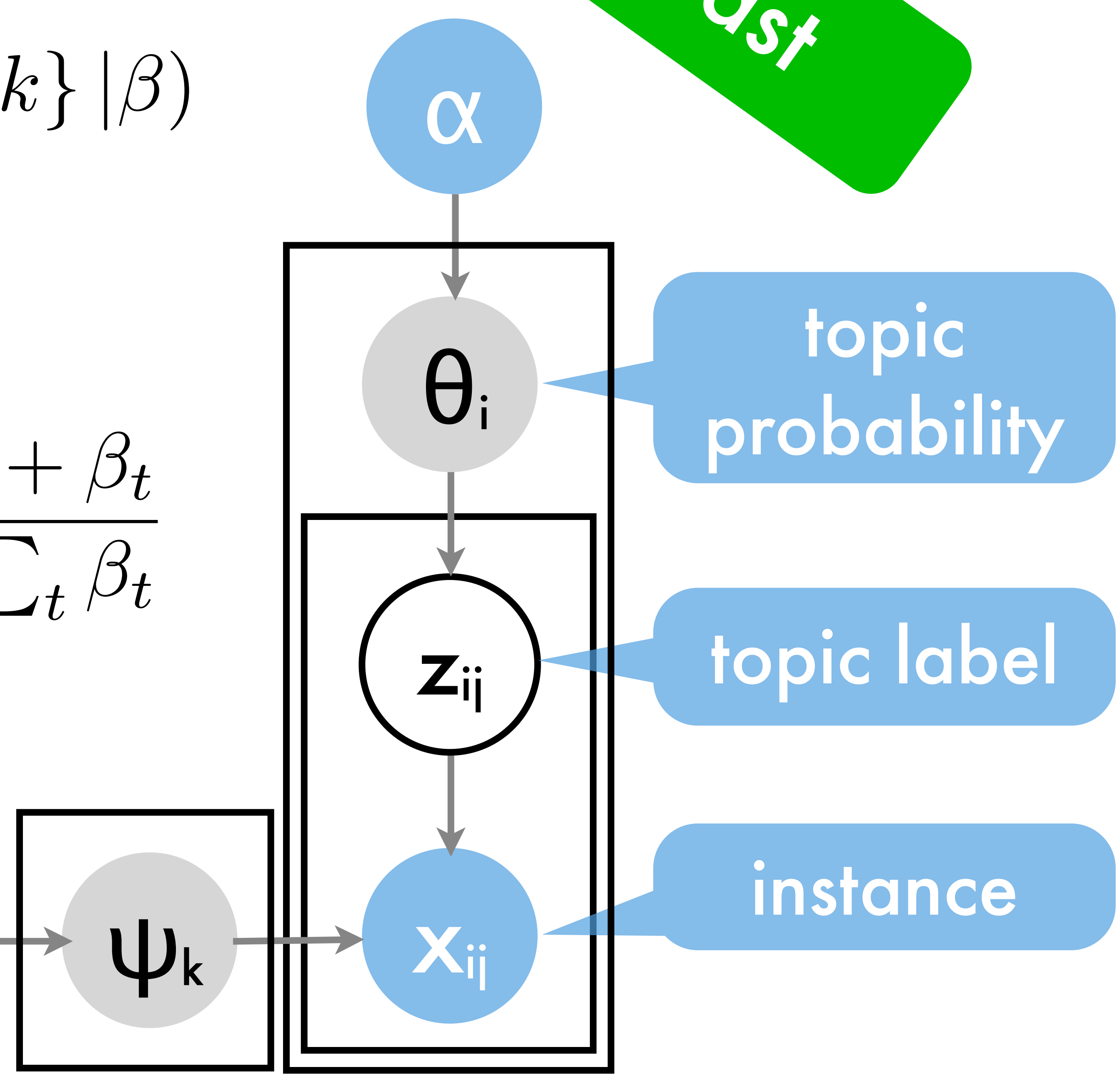
Griffiths & Steyvers, 2005

fast

$$p(z, x|\alpha, \beta)$$

$$= \prod_{i=1}^{m} p(z_i|\alpha) \prod_{k=1}^{k} p(\{x_{ij}|z_{ij} = k\}\,|\beta)$$

$$\frac{n^{-ij}(t,d) + \alpha_t}{n^{-i}(d) + \sum_t \alpha_t} \qquad \frac{n^{-ij}(t,w) + \beta_t}{n^{-i}(t) + \sum_t \beta_t}$$



Sequential Algorithm (Gibbs sampler)

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
      - Update global (word, topic) table

Sequential Algorithm (Gibbs sampler)

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
      - Update global (word, topic) table

this kills parallelism

State of the art
UMass Mallet, UC Irvine, Google

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
  - Update global (word, topic) table

$$p(t|w_{ij}) \propto \beta_w \frac{\alpha_t}{n(t) + \bar{\beta}} + \beta_w \frac{n(t, d = i)}{n(t) + \bar{\beta}} + \frac{n(t, w = w_{ij}) \left[n(t, d = i) + \alpha_t \right]}{n(t) + \bar{\beta}}$$

YAHOO!

State of the art
UMass Mallet, UC Irvine, Google

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
  - Update global (word, topic) table

$$p(t|w_{ij}) \propto \beta_w \frac{\alpha_t}{n(t) + \bar{\beta}} + \beta_w \frac{n(t, d=i)}{n(t) + \bar{\beta}} + \frac{n(t, w=w_{ij}) \left[n(t, d=i) + \alpha_t\right]}{n(t) + \bar{\beta}}$$

slow

YAHOO!

State of the art
UMass Mallet, UC Irvine, Google

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
  - Update global (word, topic) table

changes rapidly

$$p(t|w_{ij}) \propto \beta_w \frac{\alpha_t}{n(t) + \bar{\beta}} + \beta_w \frac{n(t, d = i)}{n(t) + \bar{\beta}} + \frac{n(t, w = w_{ij}) \left[n(t, d = i) + \alpha_t\right]}{n(t) + \bar{\beta}}$$

slow

State of the art
UMass Mallet, UC Irvine, Google

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
  - Update global (word, topic) table

changes rapidly

$$p(t|w_{ij}) \propto \beta_w \frac{\alpha_t}{n(t) + \bar{\beta}} + \beta_w \frac{n(t, d=i)}{n(t) + \bar{\beta}} + \frac{n(t, w=w_{ij}) \left[n(t, d=i) + \alpha_t \right]}{n(t) + \bar{\beta}}$$

slow

moderately fast

State of the art
UMass Mallet, UC Irvine, Google

- For 1000 iterations do
  - For each document do
    - For each word in the document do
      - Resample topic for the word
      - Update local (document, topic) table
      - Update CPU local (word, topic) table
  - Update global (word, topic) table

table out of sync

memory inefficient

blocking

network bound

changes rapidly

$$p(t|w_{ij}) \propto \beta_w \frac{\alpha_t}{n(t) + \bar{\beta}} + \beta_w \frac{n(t, d = i)}{n(t) + \bar{\beta}} + \frac{n(t, w = w_{ij}) \left[n(t, d = i) + \alpha_t\right]}{n(t) + \bar{\beta}}$$

slow

moderately fast

YAHOO!

Our Approach

- For 1000 iterations do (independently per computer)
  - For each thread/core do
    - For each document do
      - For each word in the document do
        - Resample topic for the word
        - Update local (document, topic) table
        - Generate computer local (word, topic) message
    - In parallel update local (word, topic) table
  - In parallel update global (word, topic) table

Our Approach

- For 1000 iterations do (independently per computer)
  - For each thread/core do
    - For each document do
      - For each word in the document do
        - Resample topic for the word
        - Update local (document, topic) table
        - Generate computer local (word, topic) message
    - In parallel update local (word, topic) table
  - In parallel update global (word, topic) table

network bound

concurrent cpu hdd net

YAHOO!

Our Approach

- For 1000 iterations do (independently per computer)
  - For each thread/core do
    - For each document do
      - For each word in the document do
        - Resample topic for the word
        - Update local (document, topic) table
        - Generate computer local (word, topic) message
    - In parallel update local (word, topic) table
  - In parallel update global (word, topic) table

network bound

memory inefficient

concurrent cpu hdd net

minimal view

YAHOO!

Our Approach

- For 1000 iterations do (independently per computer)
  - For each thread/core do
    - For each document do
      - For each word in the document do
        - Resample topic for the word
        - Update local (document, topic) table
        - Generate computer local (word, topic) message
    - In parallel update local (word, topic) table
  - In parallel update global (word, topic) table

network bound

memory inefficient

table out of sync

concurrent cpu hdd net

minimal view

continuous sync

YAHOO!

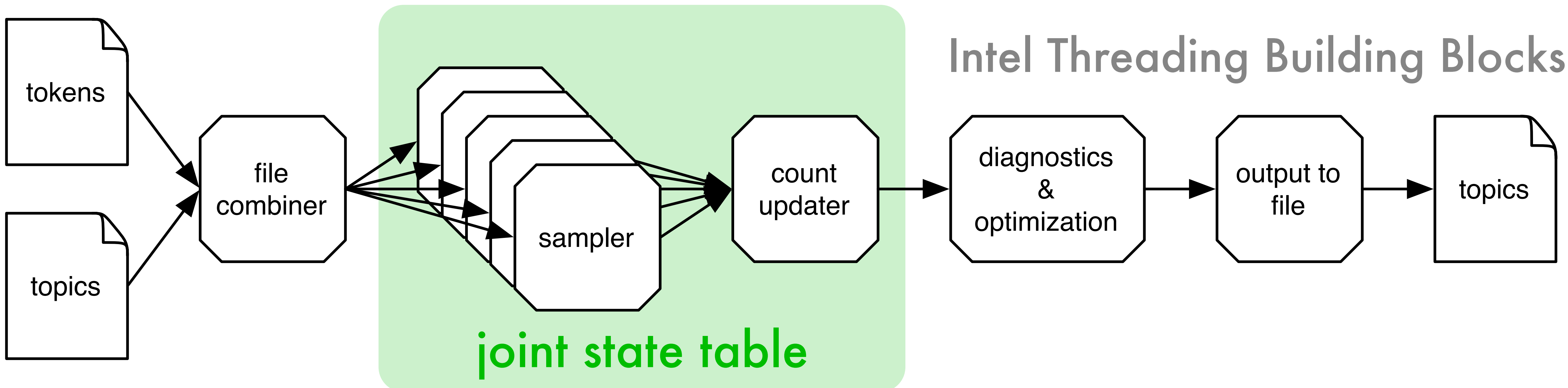
Our Approach

- For 1000 iterations do (independently per computer)
  - For each thread/core do
    - For each document do
      - For each word in the document do
        - Resample topic for the word
        - Update local (document, topic) table
        - Generate computer local (word, topic) message
    - In parallel update local (word, topic) table
  - In parallel update global (word, topic) table

network bound	memory inefficient	table out of sync	blocking
concurrent cpu hdd net	minimal view	continuous sync	barrier free

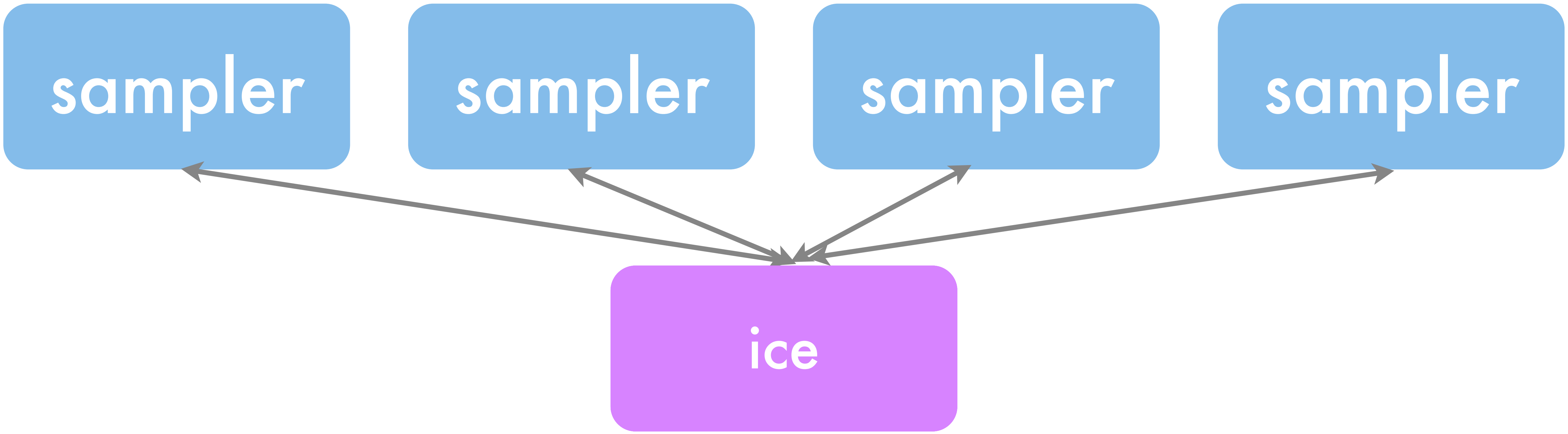
Architecture details

Multicore Architecture



- Decouple multithreaded sampling and updating (almost) avoids stalling for locks in the sampler
- Joint state table
  - much less memory required
  - samplers syncronized (10 docs vs. millions delay)
- Hyperparameter update via stochastic gradient descent
- No need to keep documents in memory (streaming)

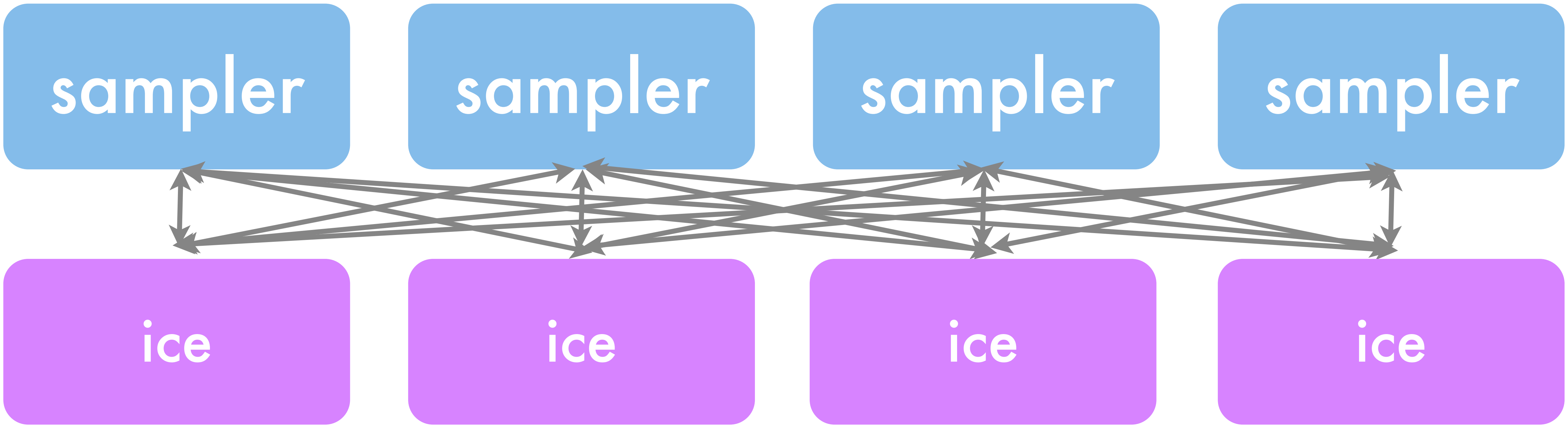
Cluster Architecture



- Distributed (key,value) storage via memcached
- Background asynchronous synchronization
  - single word at a time to avoid deadlocks
  - no need to have joint dictionary
  - uses disk, network, cpu simultaneously

YAHOO!

Cluster Architecture



- Distributed (key,value) storage via ICE
- Background asynchronous synchronization
  - single word at a time to avoid deadlocks
  - no need to have joint dictionary
  - uses disk, network, cpu simultaneously

Making it work

- Startup
  - Randomly initialize topics on each node
    (read from disk if already assigned - hotstart)
  - Sequential Monte Carlo for startup much faster
  - Aggregate changes on the fly
- Failover
  - State constantly being written to disk
    (worst case we lose 1 iteration out of 1000)
  - Restart via standard startup routine
- Achilles heel: need to restart from checkpoint if even a single machine dies.

YAHOO!

Easily extensible

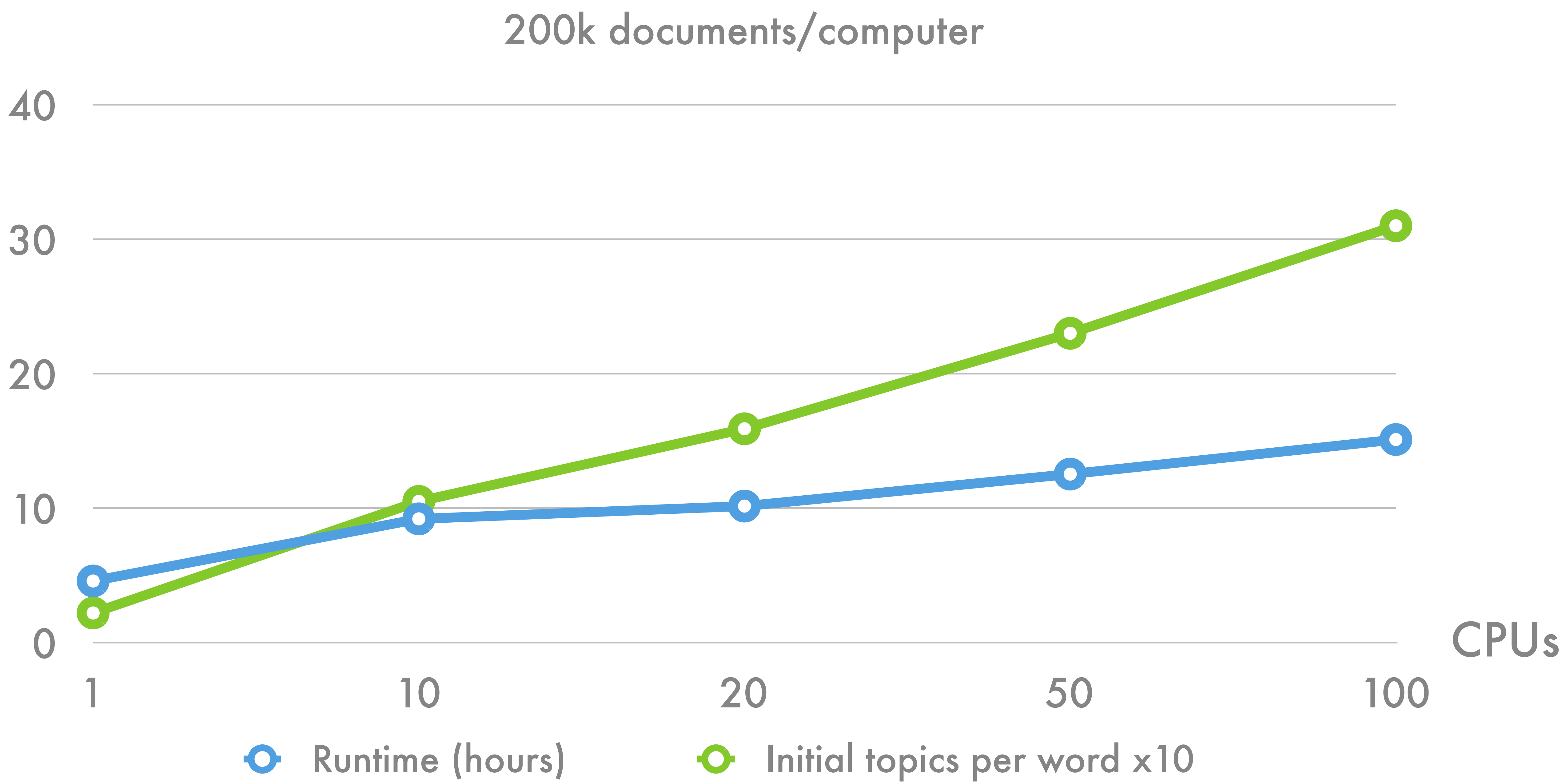
- Better language model (topical n-grams)
  can process millions of users (vs 1000s)
- Conditioning on side information (upstream)
  estimate topic based on authorship, source, joint user model ...
- Conditioning on dictionaries (downstream)
  integrate topics between different languages
- Time dependent sampler for user model
  approximate inference per episode

	Google LDA	Mallet	Irvine'08	Irvine'09	Yahoo LDA
Multicore	no	yes	yes	yes	yes
Cluster	MPI	no	MPI	point 2 point	memcached
State table	dictionary split	separate sparse	separate	separate	joint sparse
Schedule	synchronous exact	synchronous exact	synchronous exact	asynchronous approximate messages	asynchronous exact

Speed

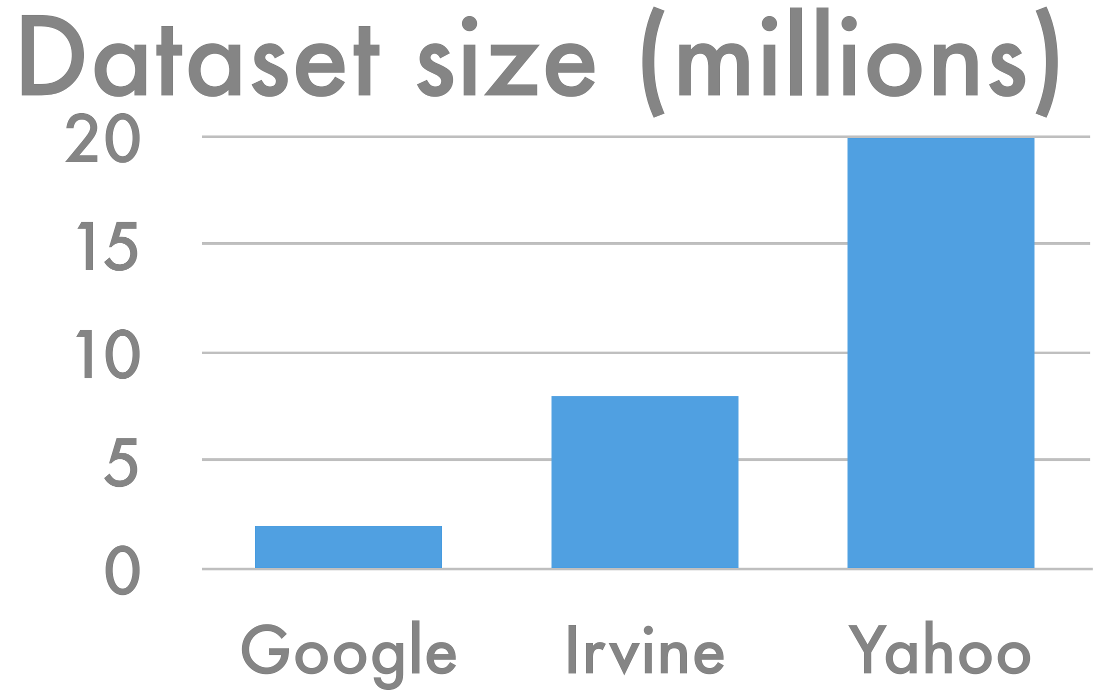
- **1M documents per day** on 1 computer
  (1000 topics per doc, 1000 words per doc)
- **350k documents per day** per node
  (context switches & memcached & stray reducers)
- 8 Million docs (Pubmed)
  (sampler does not burn in well - too short doc)
  - Irvine: 128 machines, 10 hours
  - Yahoo: 1 machine, 11 days
  - Yahoo: 20 machines, 9 hours
- 20 Million docs (Yahoo! News Articles)
  - Yahoo: 100 machines, 12 hours

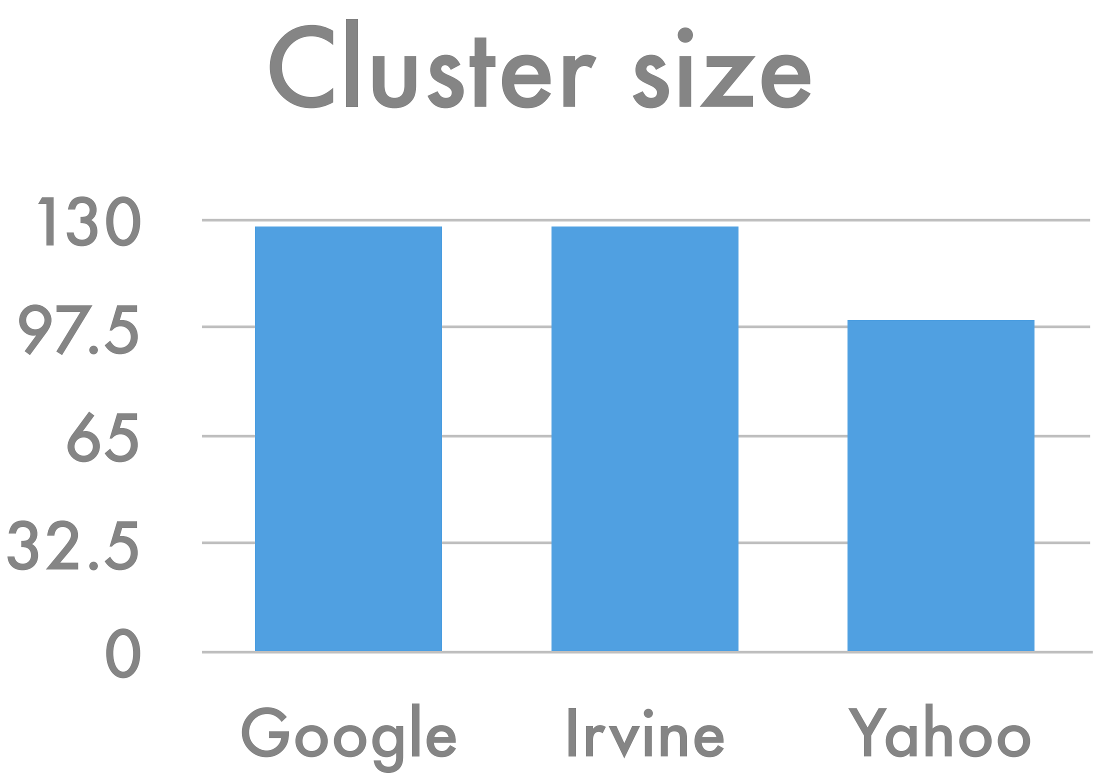
Scalability

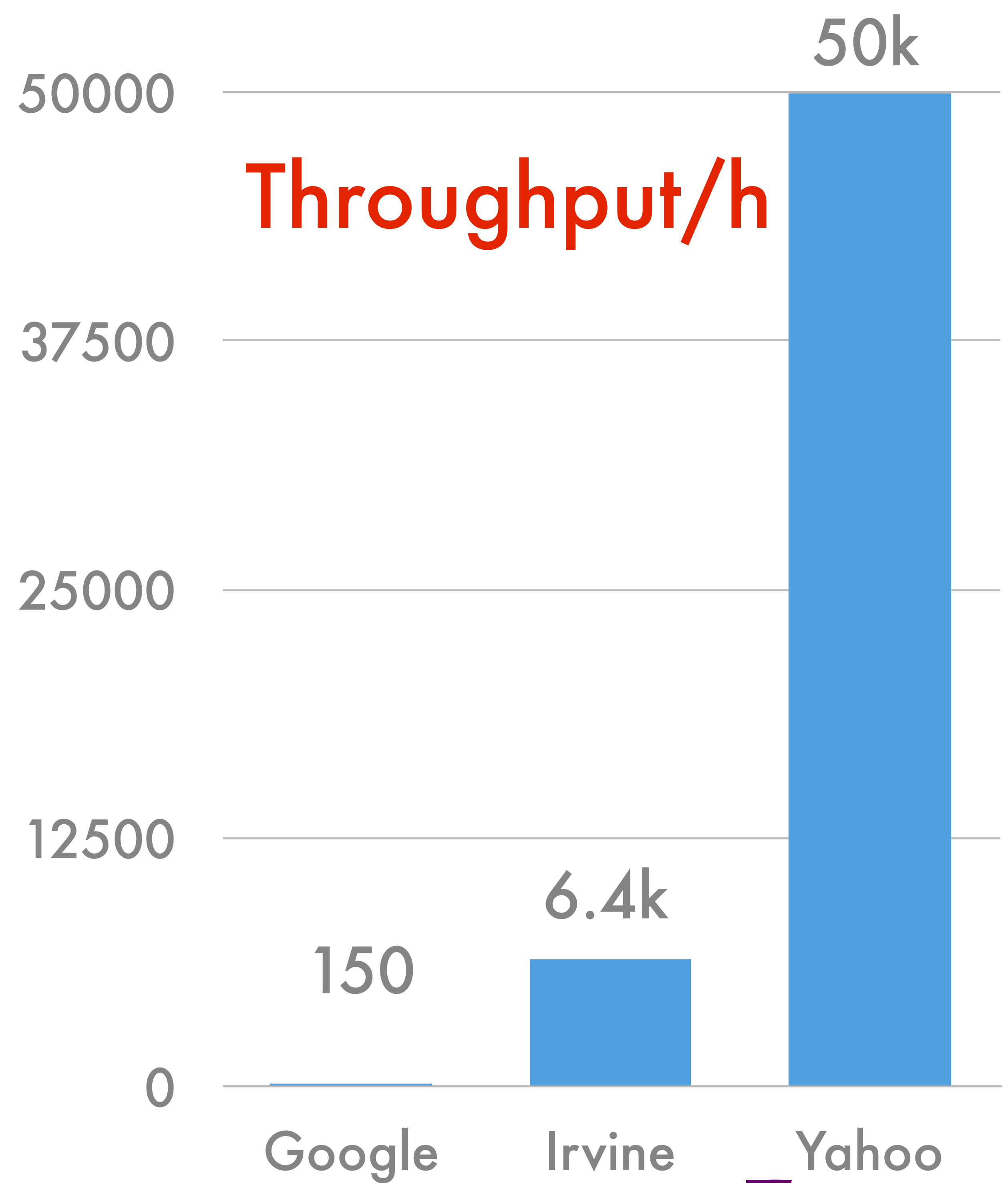


Likelihood even improves with parallelism!
-3.295 (1 node) -3.288 (10 nodes) -3.287 (20 nodes)

The Competition



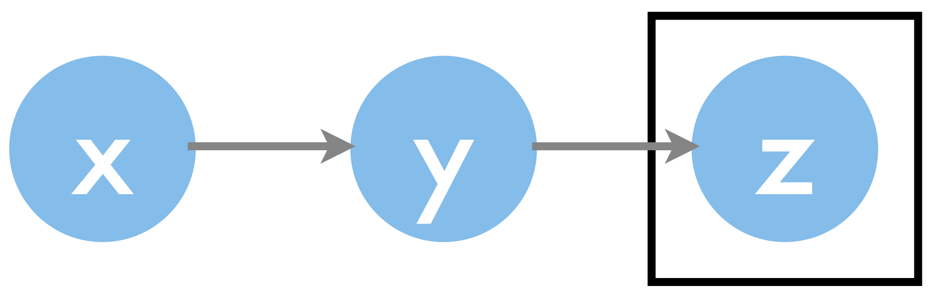


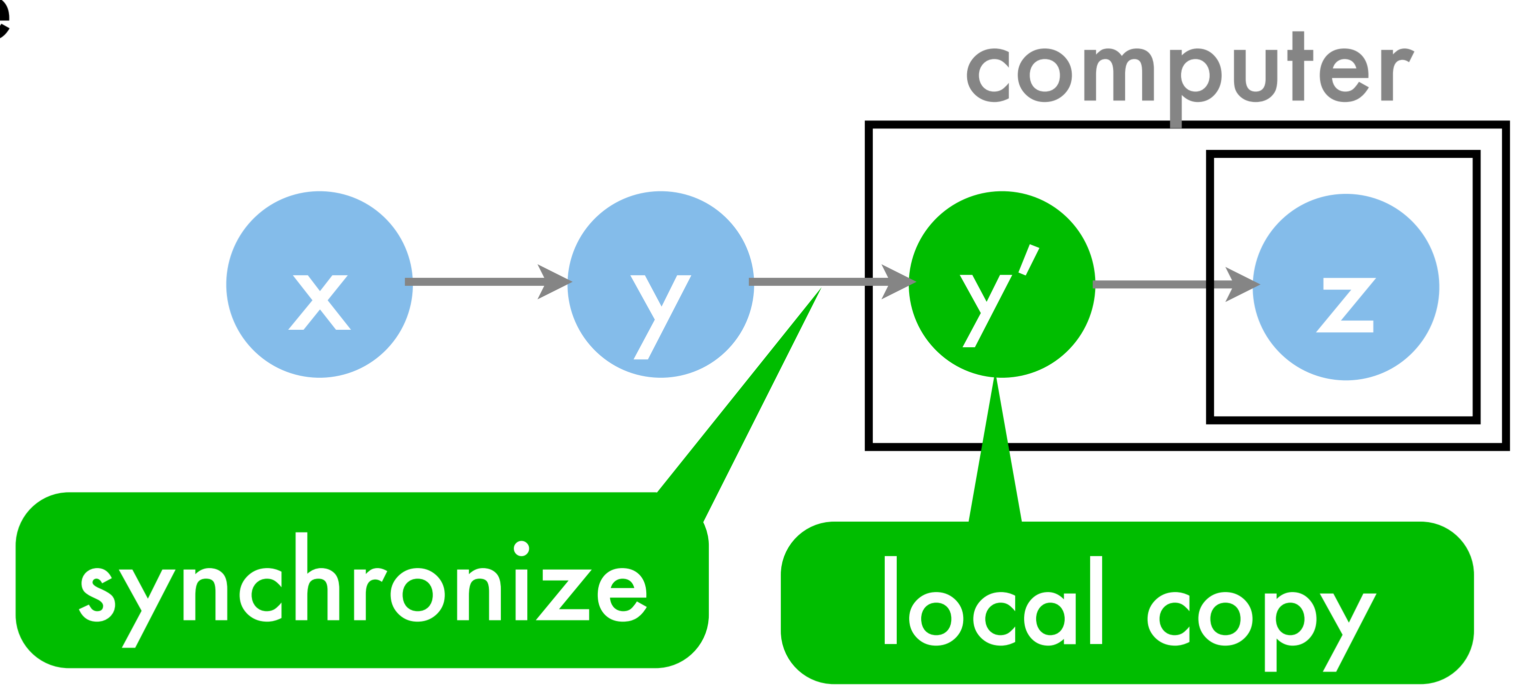


Design Principles

Variable Replication

- Global shared variable

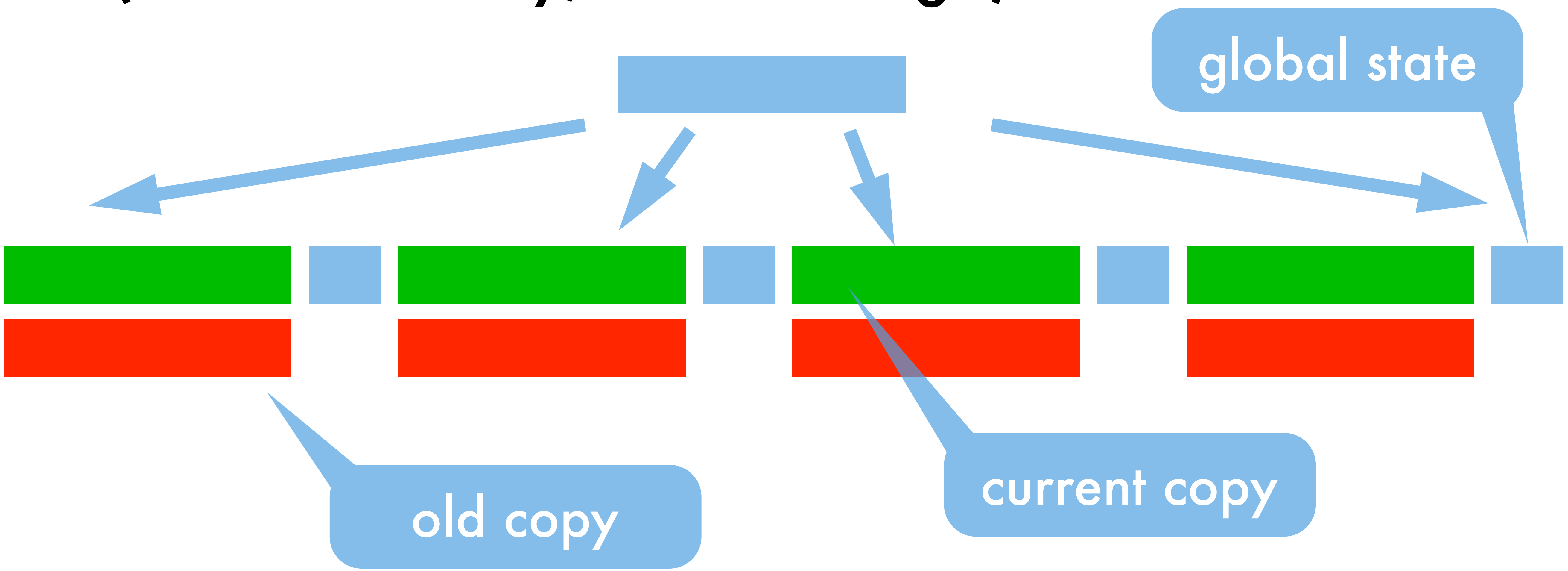




- Make local copy
  - Distributed (key,value) storage table for global copy
  - Do all bookkeeping locally (store old versions)
  - Sync local copies asynchronously using message passing (no global locks are needed)
- This is an approximation!

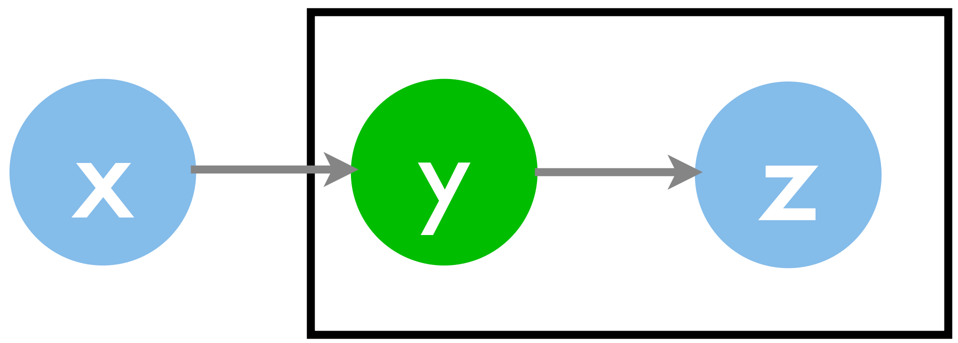
Asymmetric Message Passing

- Large global shared state space (essentially as large as the memory in computer)
- Distribute global copy over several machines (distributed key,value storage)

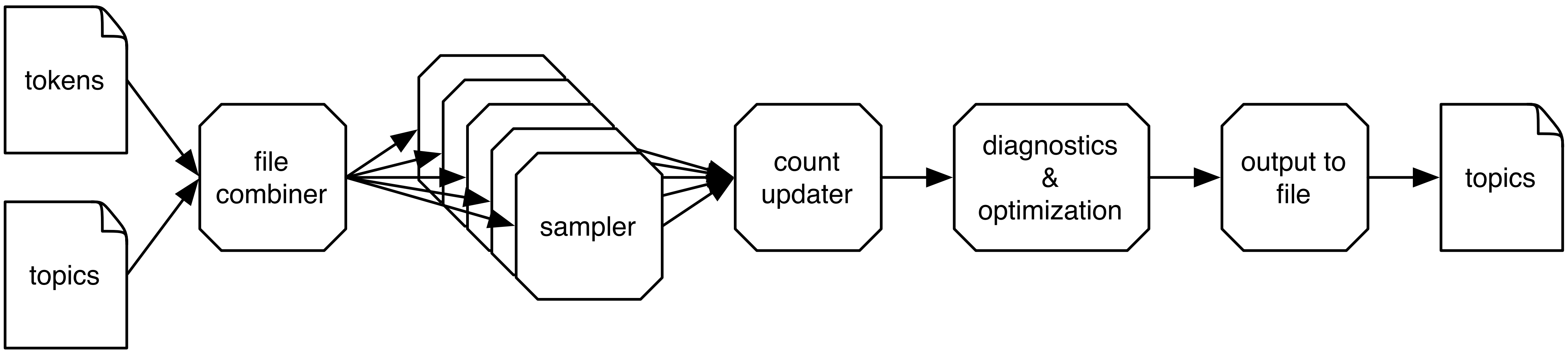


Out of core storage

- Very large state space



- Gibbs sampling requires us to traverse the data sequentially many times (think 1000x)
- Stream local data from disk and update coupling variable each time local data is accessed
- This is exact

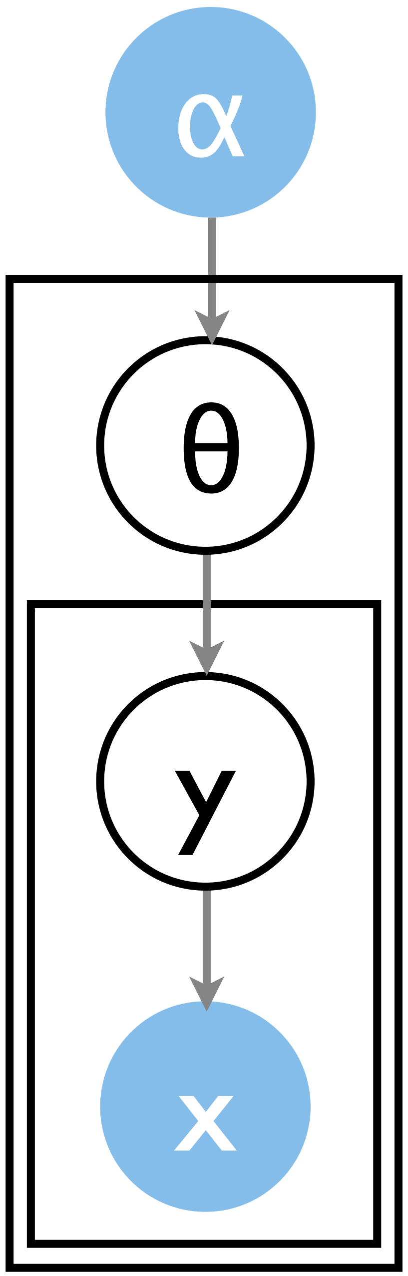


Part 6 - Advanced Modeling

Advances in Representation

Extensions to topic models

- Prior over document topic vector
  - Usually as Dirichlet distribution
  - Use correlation between topics (CTM)
  - Hierarchical structure over topics
- Document structure
  - Bag of words
  - n-grams (Li & McCallum)
  - Simplical Mixture (Girolami & Kaban)
- Side information
  - Upstream conditioning (Mimno & McCallum)
  - Downstream conditioning (Petterson et al.)
  - Supervised LDA (Blei and McAulliffe 2007; Lacoste, Sha and Jordan 2008; Zhu, Ahmed and Xing 2009)



Correlated topic models

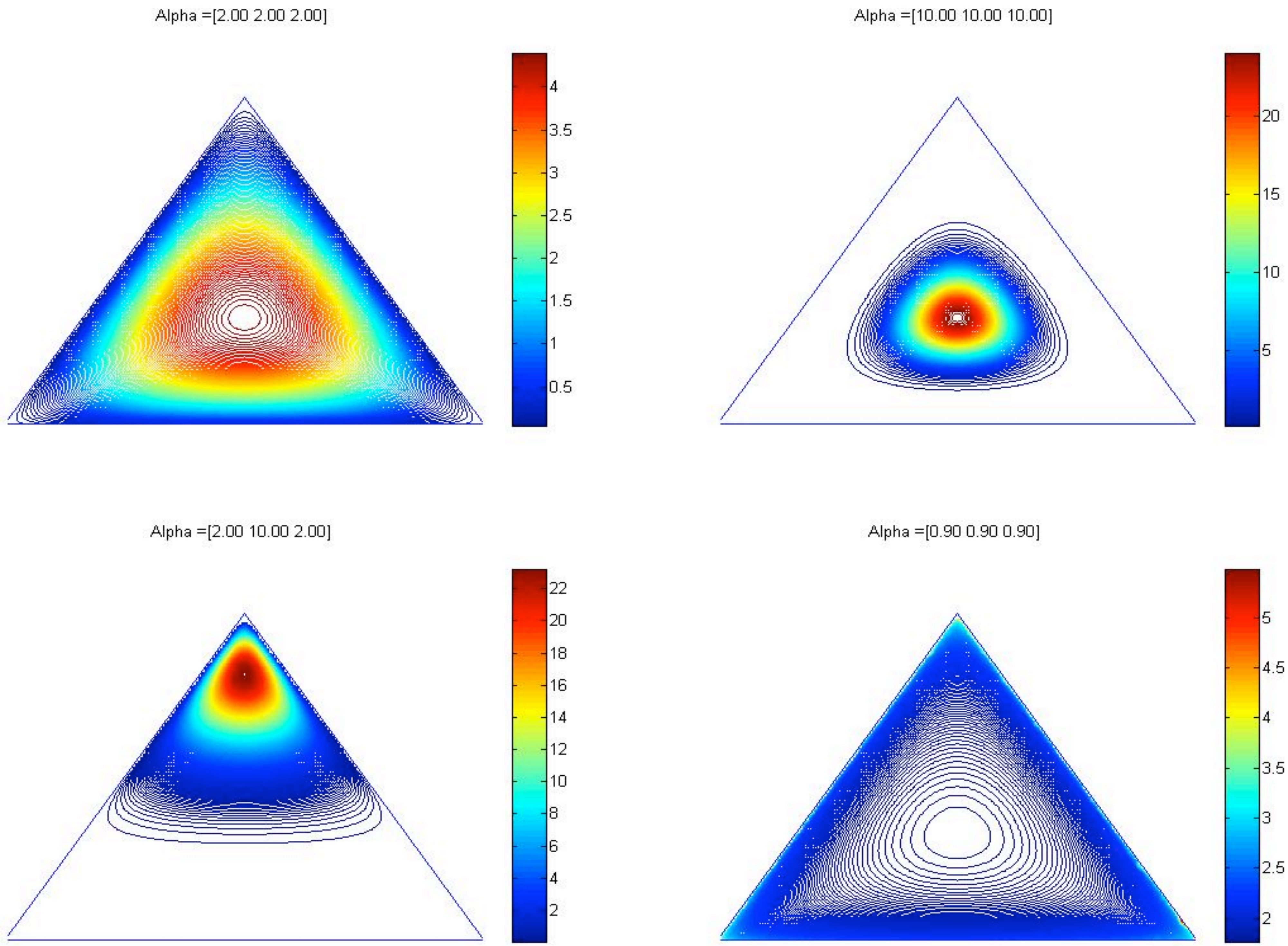
- Dirichlet distribution
  - Can only model which topics are hot
  - Does not model relationships between topics

Correlated topic models

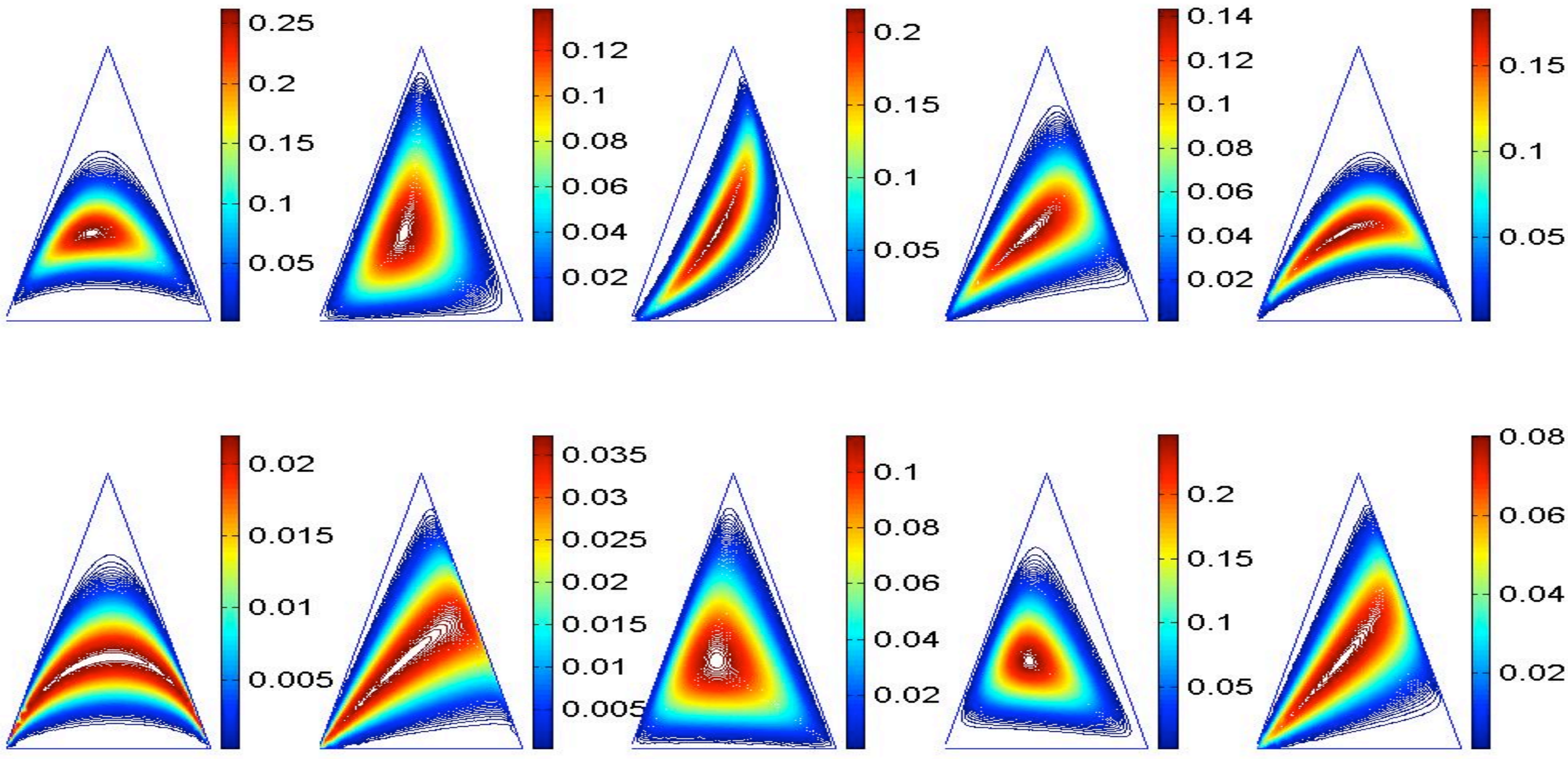
- Dirichlet distribution
  - Can only model which topics are hot
  - Does not model relationships between topics
- Key idea
  - We expect to see documents about sports and health but not about sports and politics
  - Uses a logistic normal distribution as a prior
- Conjugacy is no longer maintained
- Inference is harder than in LDA

Blei & Lafferty 2005; Ahmed & Xing 2007

Dirichlet prior on topics

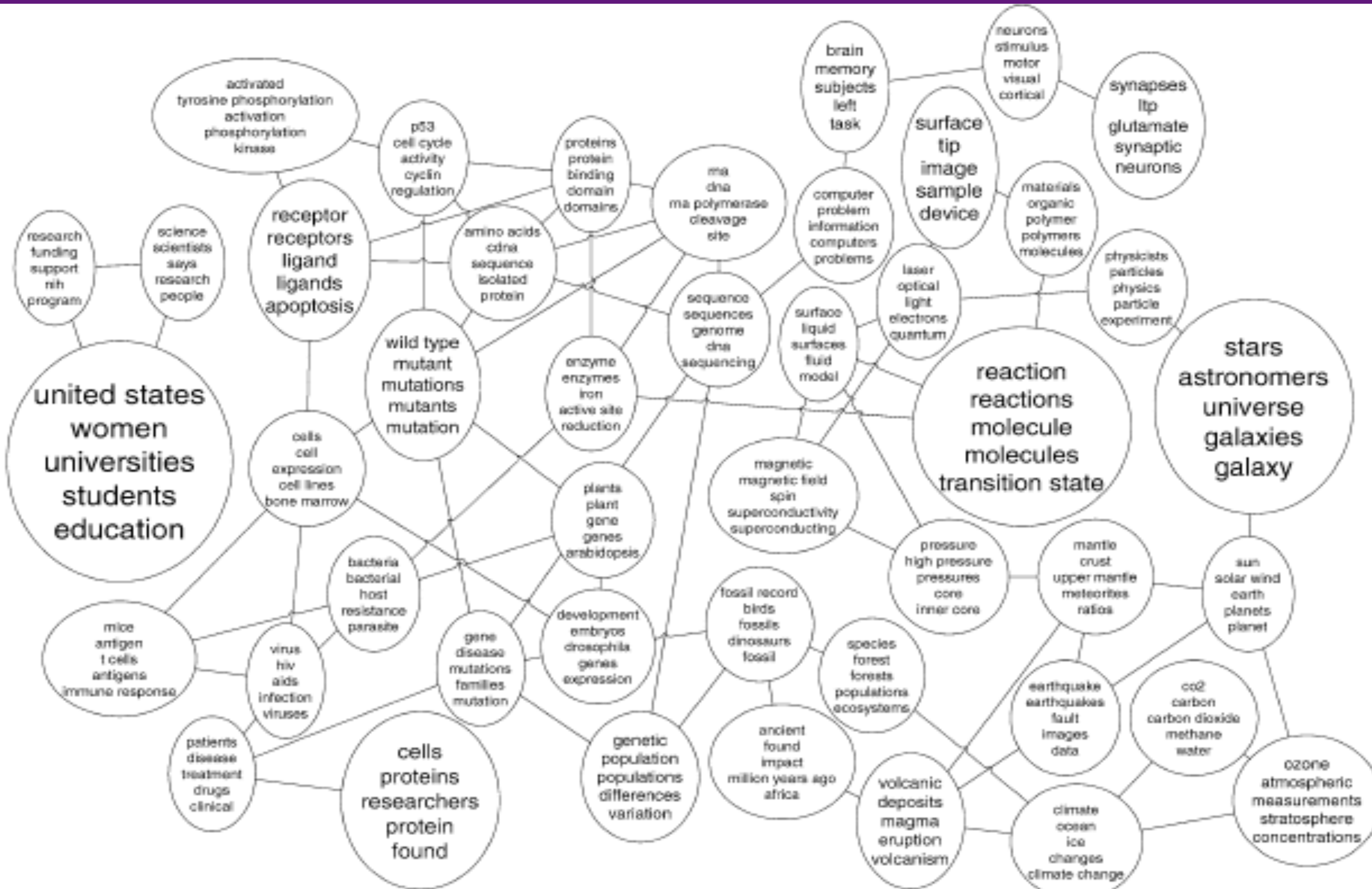


Log-normal prior on topics



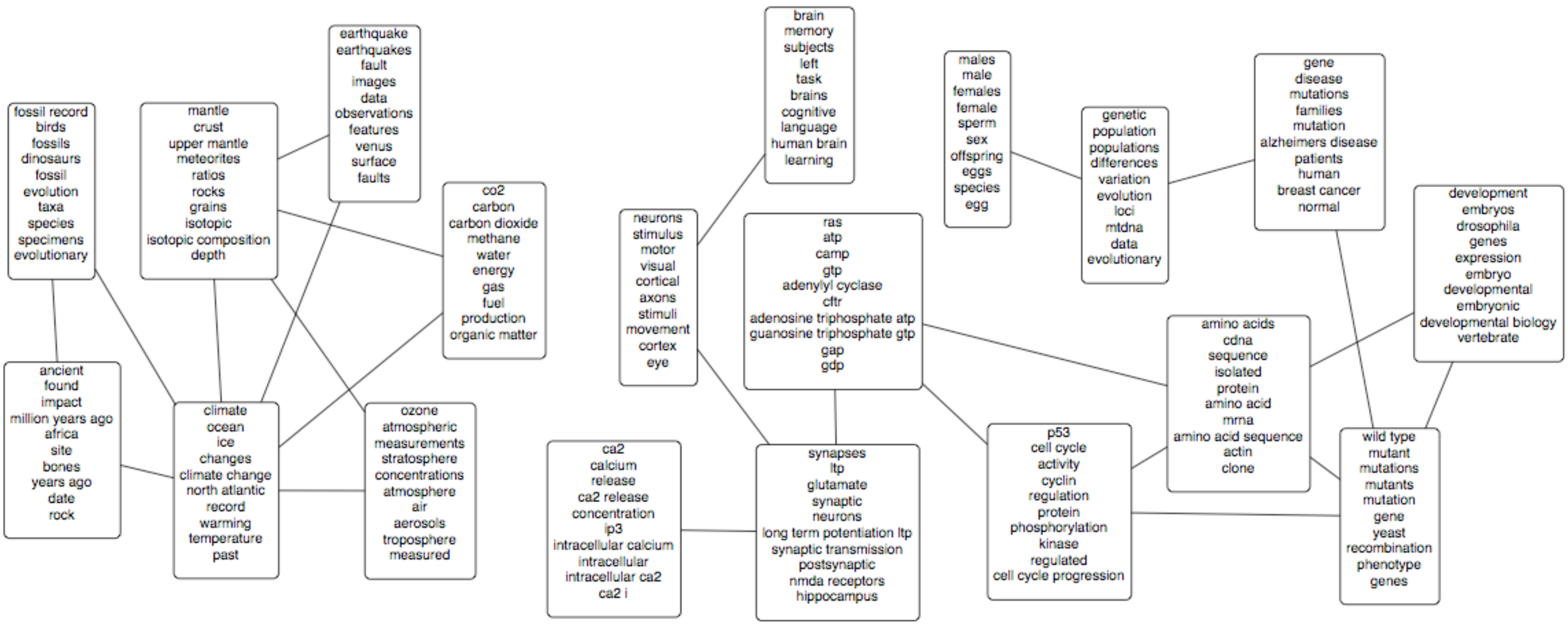
$\theta = e^{\eta - g(\eta)}$ **with** $\eta \sim \mathcal{N}(\mu, \Sigma)$

Correlated topics



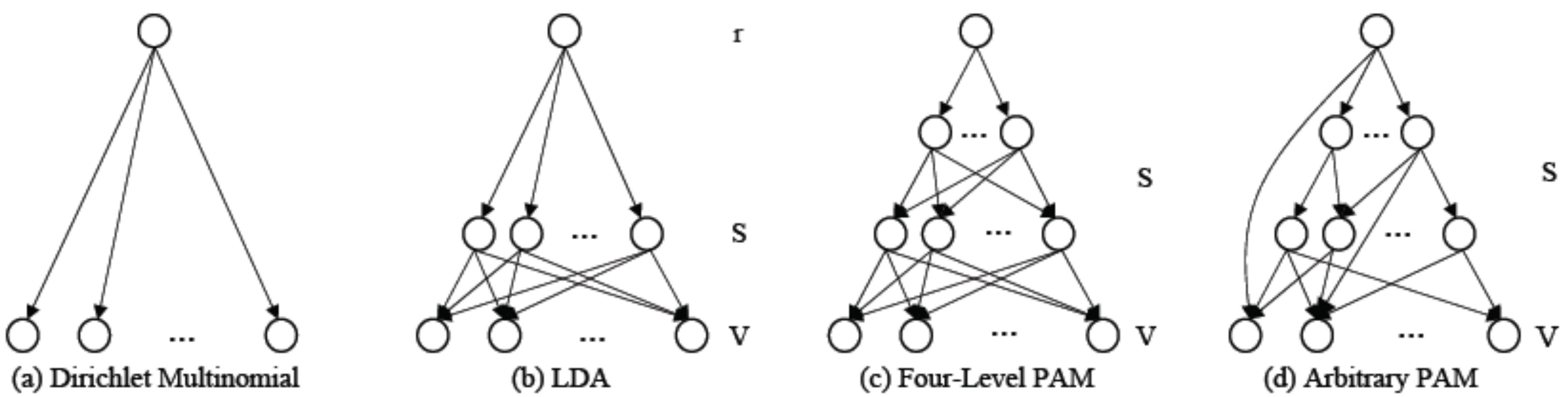
Blei and Lafferty 2005

Correlated topics



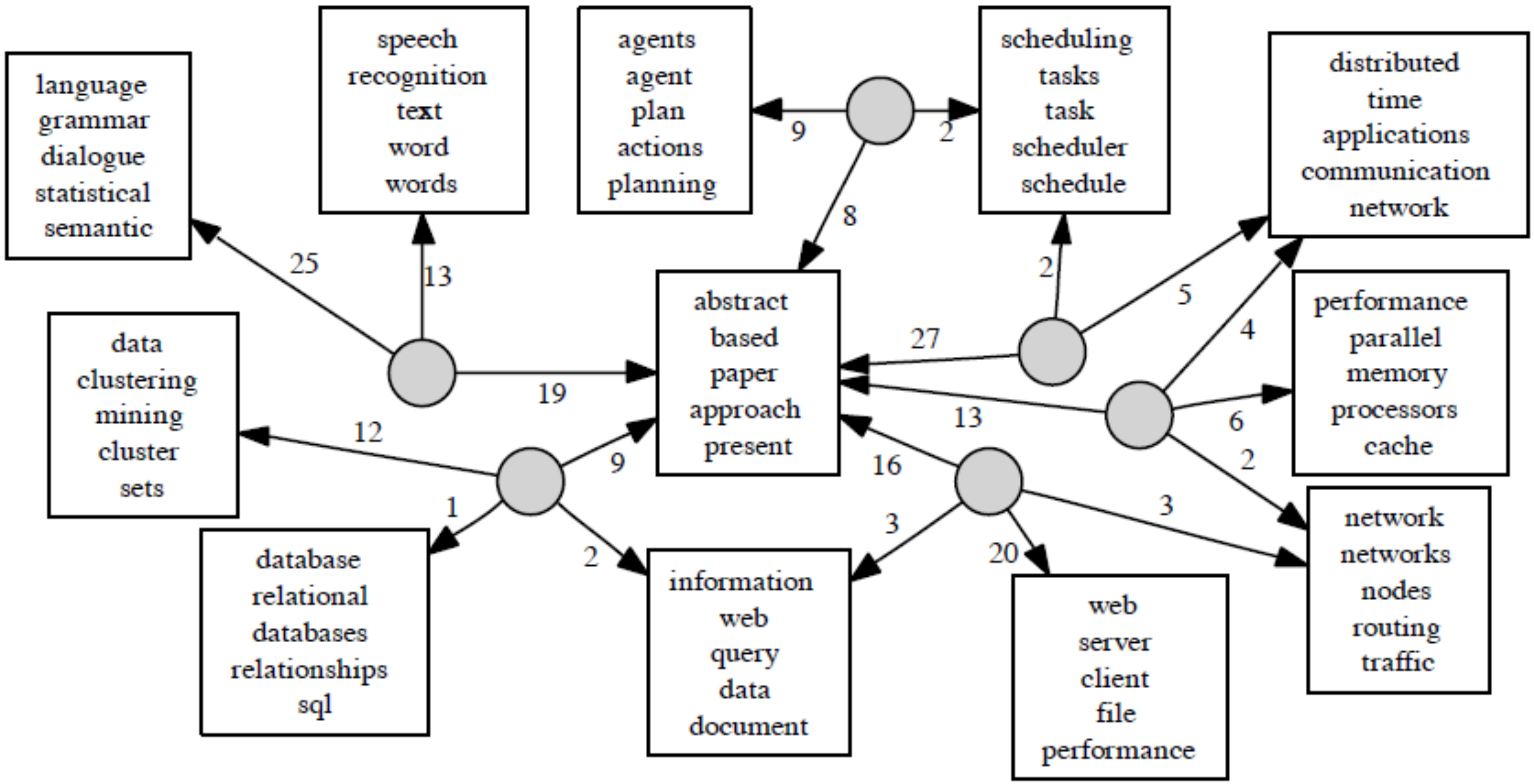
Pachinko Allocation

- Model the prior as a Directed Acyclic Graph
- Each document is modeled as multiple paths
- To sample a word, first select a path and then sample a word from the final topic
- The topics reside on the leaves of the tree



Li and McCallum 2006

Pachinko Allocation



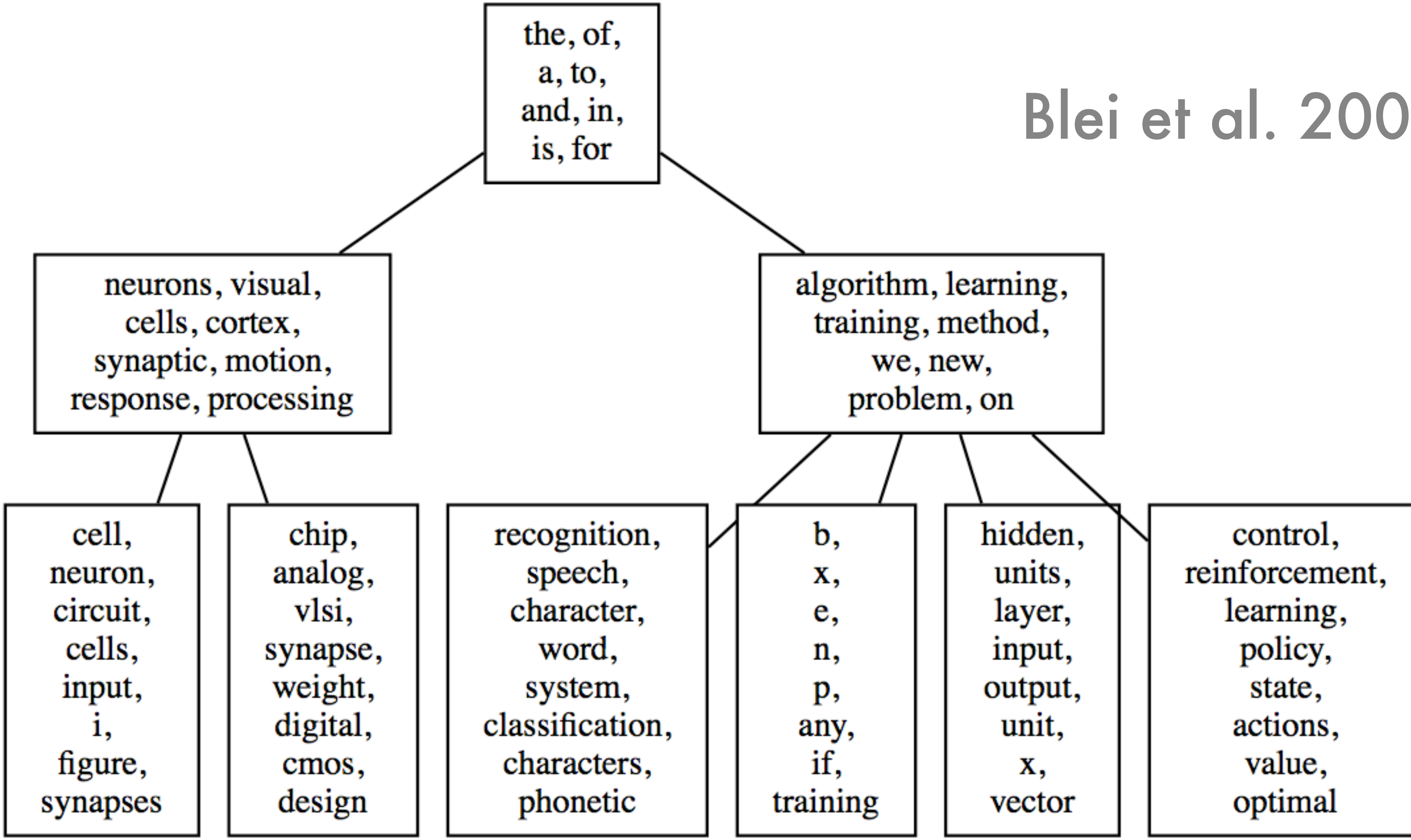
Li and McCallum 2006

Topic Hierarchies

- Topics can appear anywhere in the tree
- Each document is modeled as
  - Single path over the tree (Blei et al., 2004)
  - Multiple paths over the tree (Mimno et al.,2007)

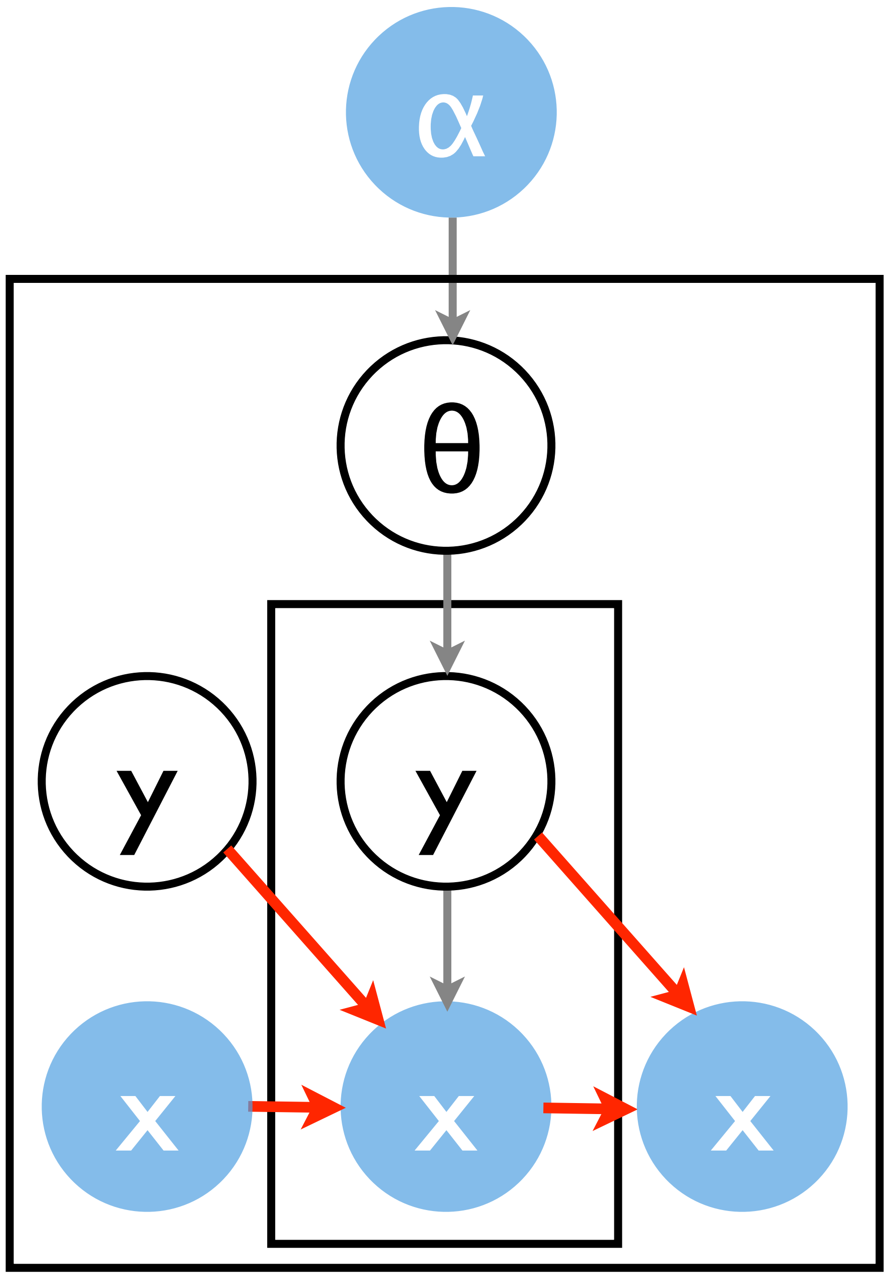
Topic Hierarchies

Blei et al. 2004



Topical n-grams

- Documents as bag of words
- Exploit sequential structure
- N-gram models
  - Capture longer phrases
  - Switch variables to determine segments
  - Dynamic programming needed



Girolami & Kaban, 2003; Wallach, 2006; Wang & McCallum, 2007

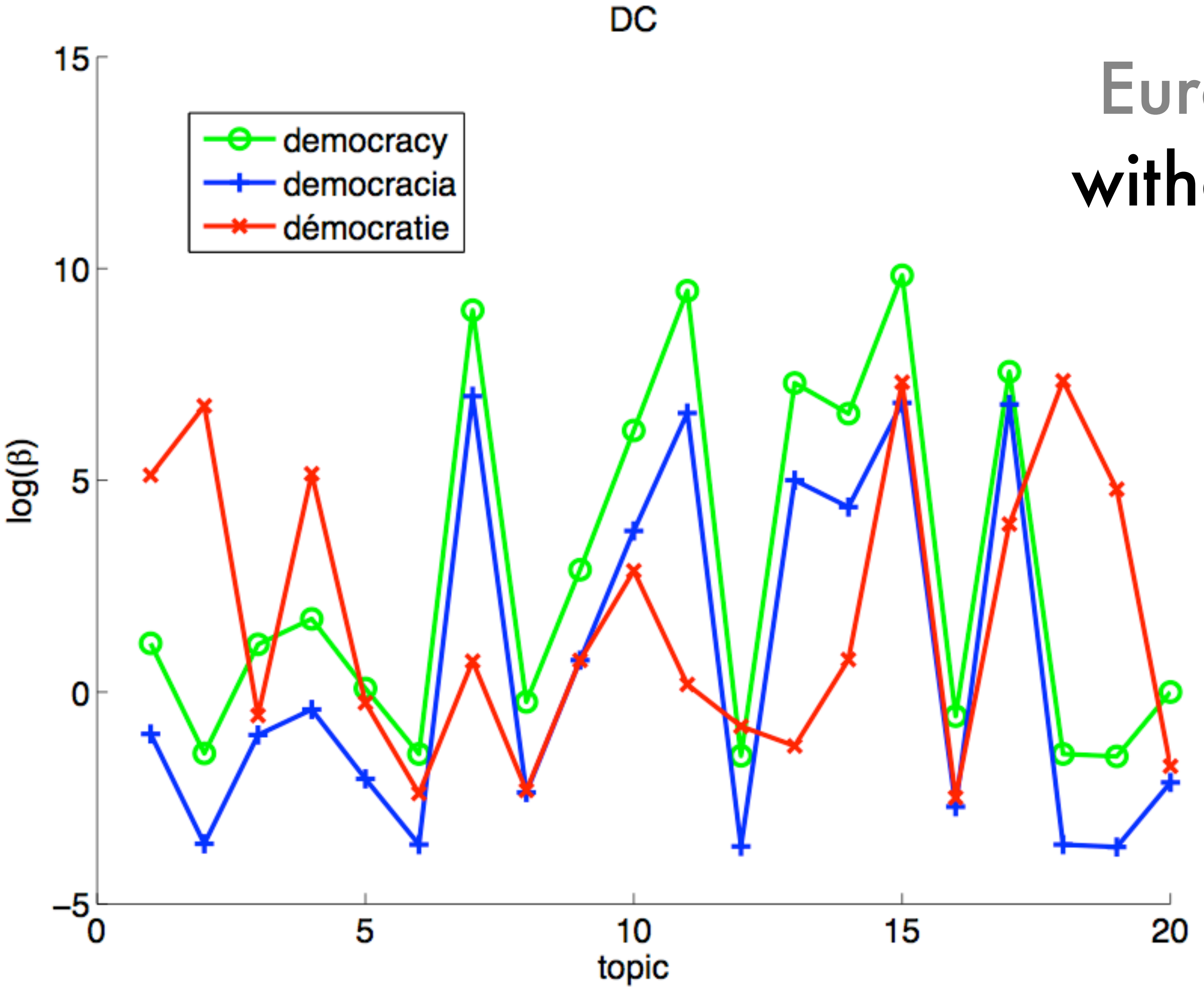
Topic n-grams

Speech Recognition			Support Vector Machines		
LDA	n-gram (2+)	n-gram (1)	LDA	n-gram (2+)	n-gram (1)
recognition	speech recognition	speech	kernel	support vectors	kernel
system	training data	word	linear	test error	training
word	neural network	training	vector	support vector machines	support
face	error rates	system	support	training error	margin
context	neural net	recognition	set	feature space	svm
character	hidden markov model	hmm	nonlinear	training examples	solution
hmm	feature vectors	speaker	data	decision function	kernels
based	continuous speech	performance	algorithm	cost functions	regularization
frame	training procedure	phoneme	space	test inputs	adaboost
segmentation	continuous speech recognition	acoustic	pca	kkt conditions	test
training	gamma filter	words	function	leave-one-out procedure	data
characters	hidden control	context	problem	soft margin	generalization
set	speech production	systems	margin	bayesian transduction	examples
probabilities	neural nets	frame	vectors	training patterns	cost
features	input representation	trained	solution	training points	convex
faces	output layers	sequence	training	maximum margin	algorithm
words	training algorithm	phonetic	svm	strictly convex	working
frames	test set	speakers	kernels	regularization operators	feature
database	speech frames	mlp	matrix	base classifiers	sv
mlp	speaker dependent	hybrid	machines	convex optimization	functions

Side information

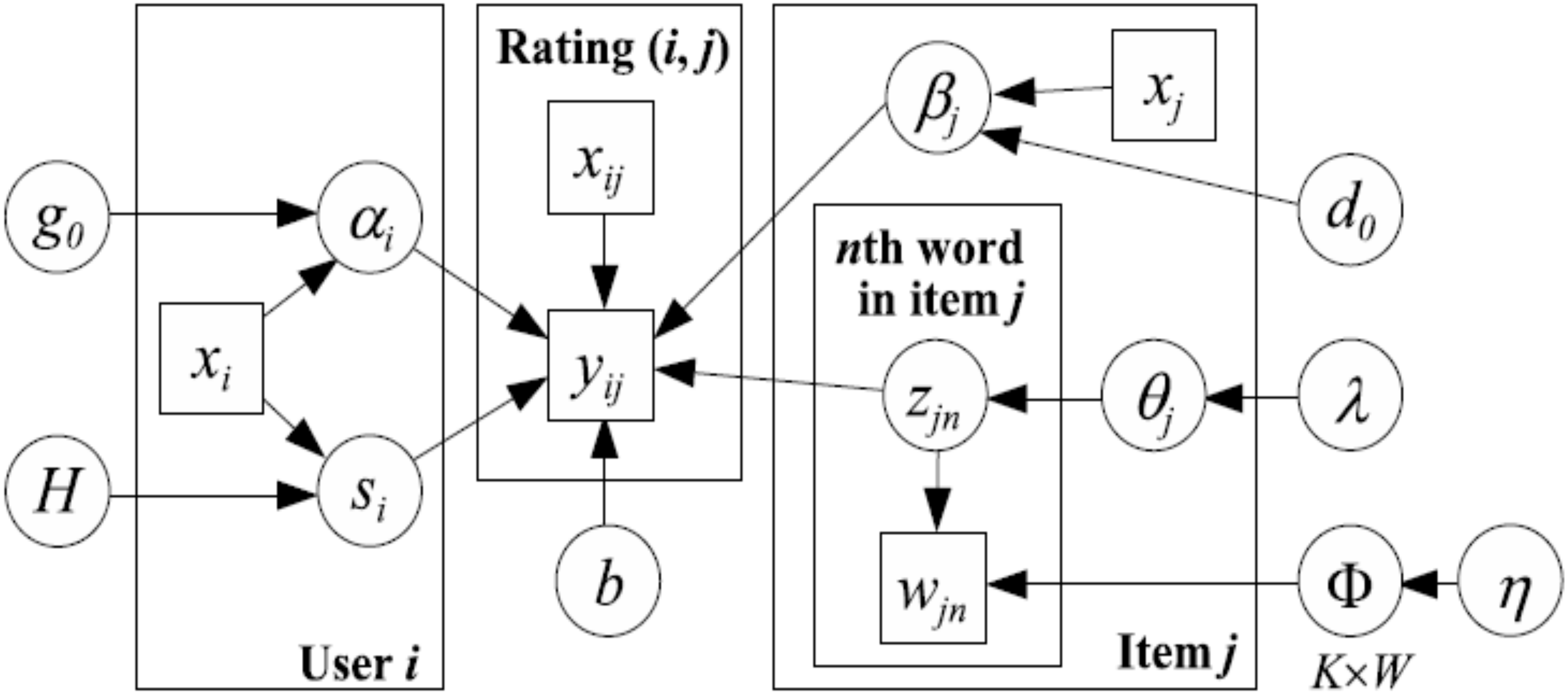
- Upstream conditioning (Mimno et al., 2008)
  - Document features are informative for topics
  - Estimate topic distribution e.g. based on authors, links, timestamp
- Downstream conditioning (Petterson et al., 2010)
  - Word features are informative on topics
  - Estimate topic distribution for words e.g. based on dictionary, lexical similarity, distributional similarity
- Class labels (Blei and McAulliffe 2007; Lacoste, Sha and Jordan 2008; Zhu, Ahmed and Xing 2009)
  - Joint model of unlabeled data and labels
  - Joint likelihood - semisupervised learning done right!

Downstream conditioning



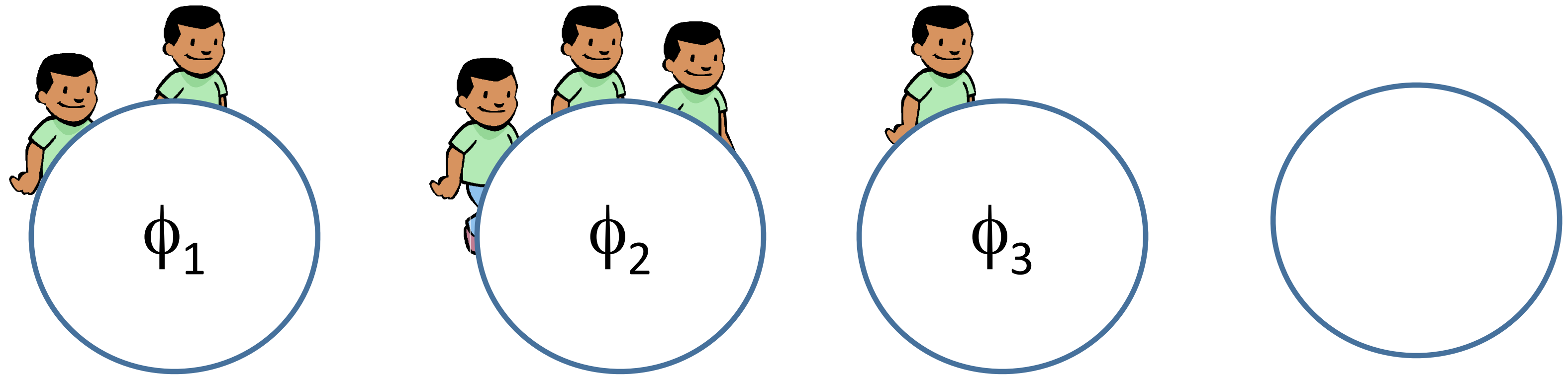
Europarl corpus
without alignment

Recommender Systems



Agarwal & Chen, 2010

Chinese Restaurant Process



Problem

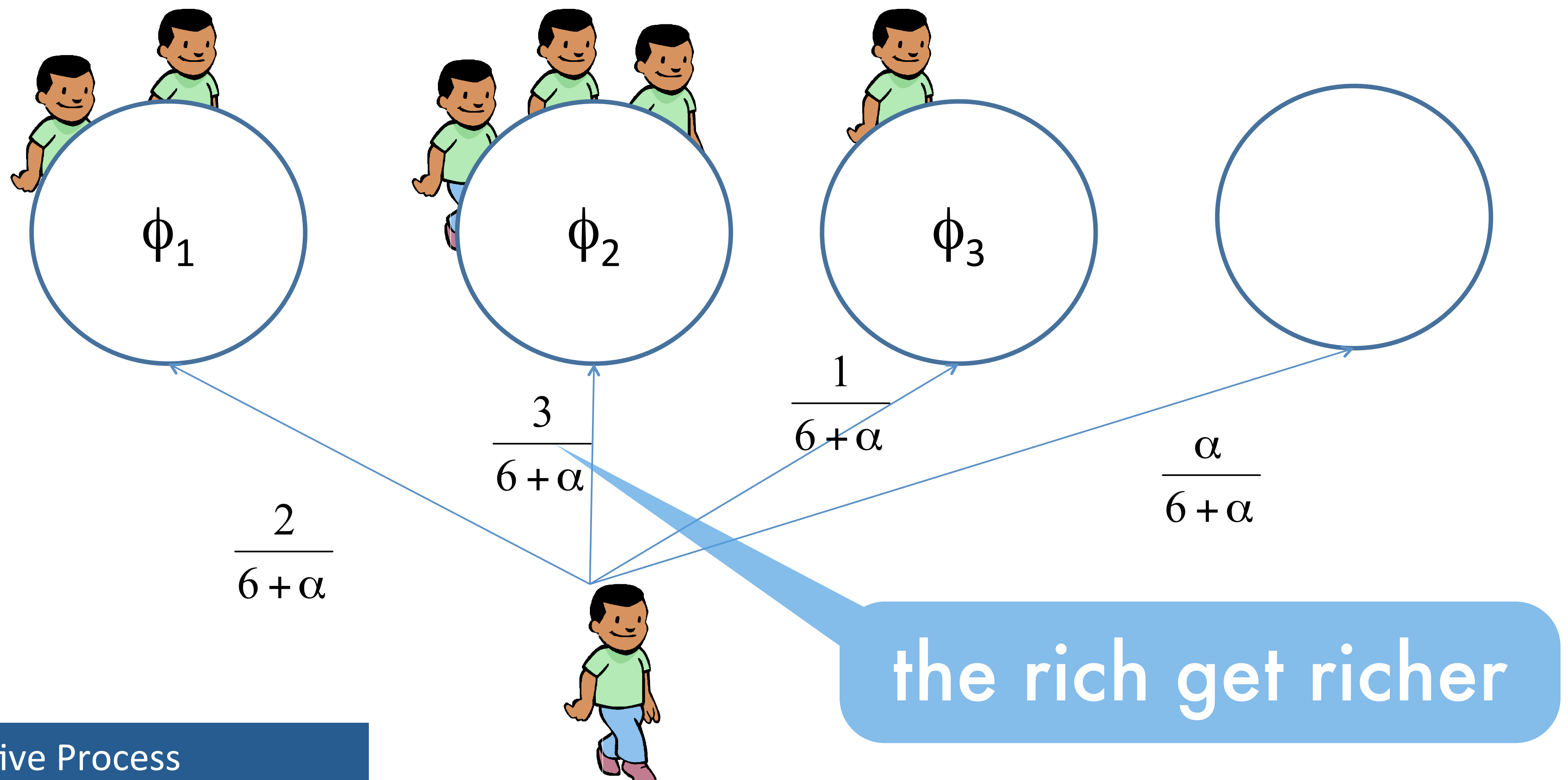
- How many clusters should we pick?
- How about a prior for infinitely many clusters?
- Finite model

$$p(y|Y, \alpha) = \frac{n(y) + \alpha_y}{n + \sum_{y'} \alpha_{y'}}$$

- Infinite model
  Assume that the total smoother weight is constant

$$p(y|Y, \alpha) = \frac{n(y)}{n + \sum_{y'} \alpha_{y'}} \text{ and } p(\mathrm{new}|Y, \alpha) = \frac{\alpha}{n + \alpha}$$

Chinese Restaurant Metaphor



Generative Process

-For data point x_i

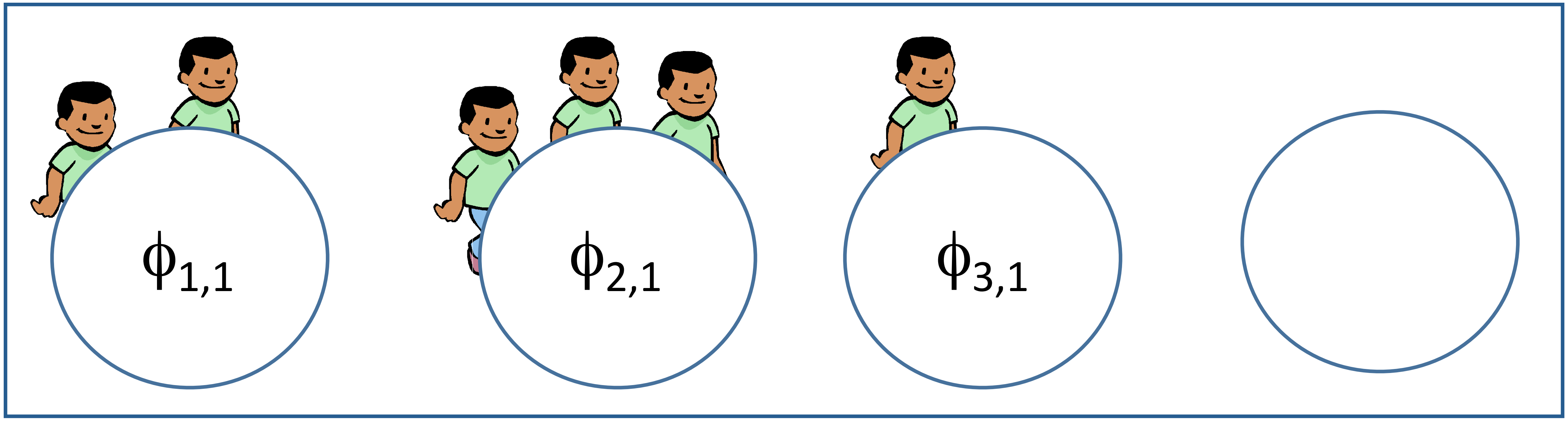
- Choose table $j \propto m_j$ and Sample $x_i \sim f(\phi_j)$
- Choose a new table $K+1 \propto \alpha$
  - Sample $\phi_{K+1} \sim G_0$ and Sample $x_i \sim f(\phi_{K+1})$

Pitman; Antoniak; Ishwaran; Jordan et al.; Teh et al.;

Evolutionary Clustering

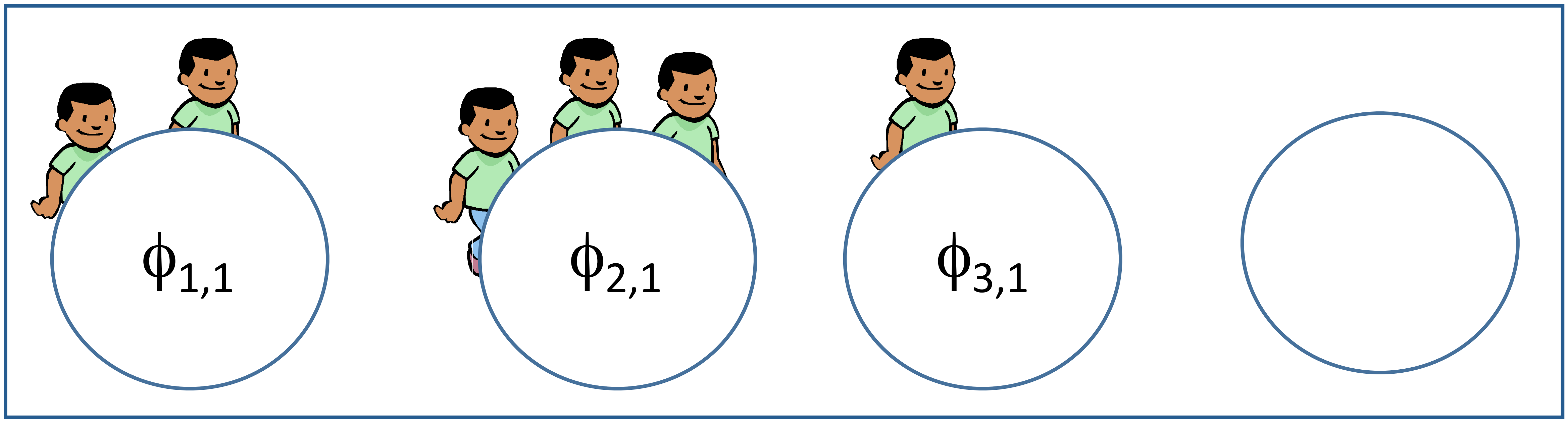
- Time series of objects, e.g. news stories
- Stories appear / disappear
- Want to keep track of clusters automatically

Recurrent Chinese Restaurant Process



T=1

Recurrent Chinese Restaurant Process

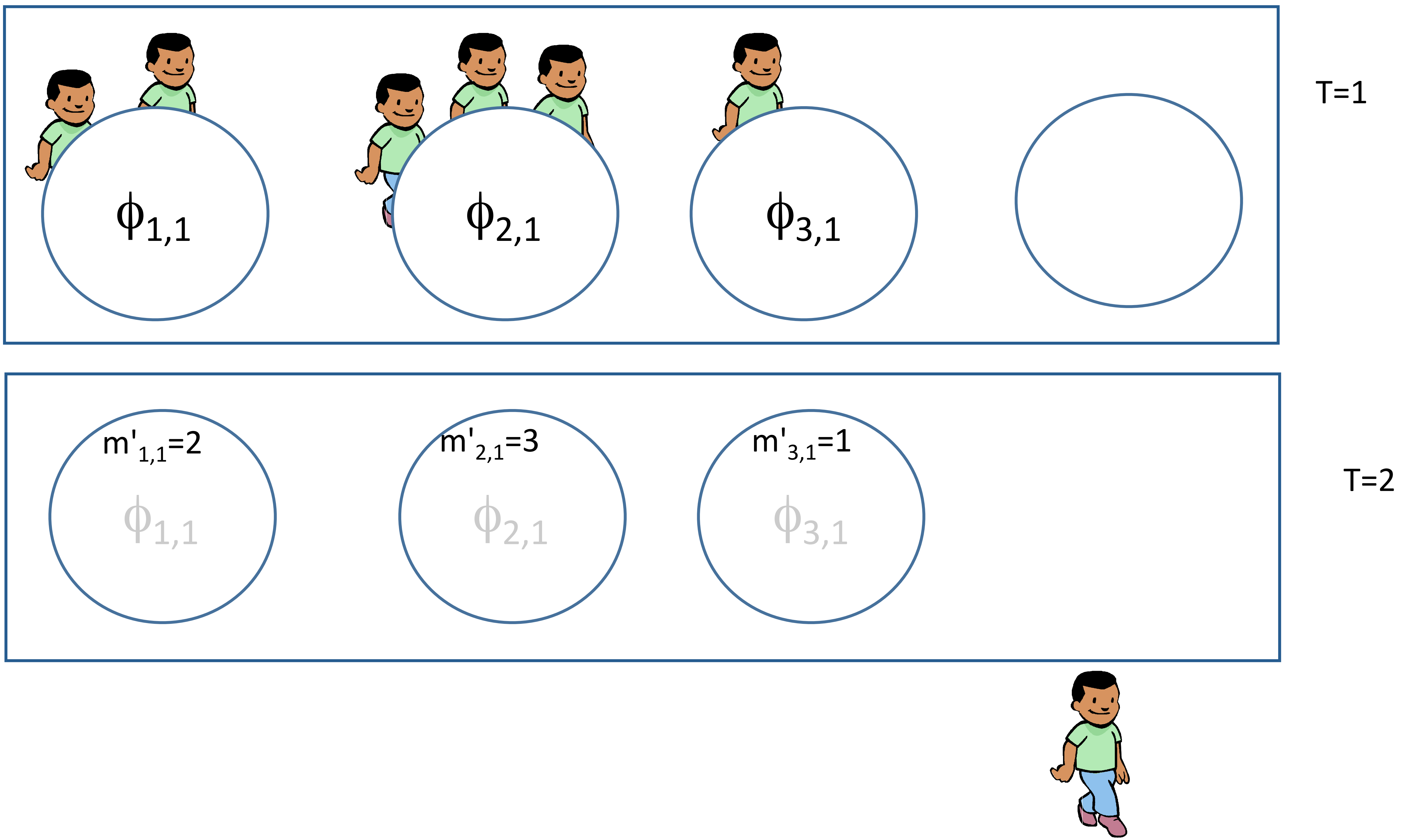


T=1

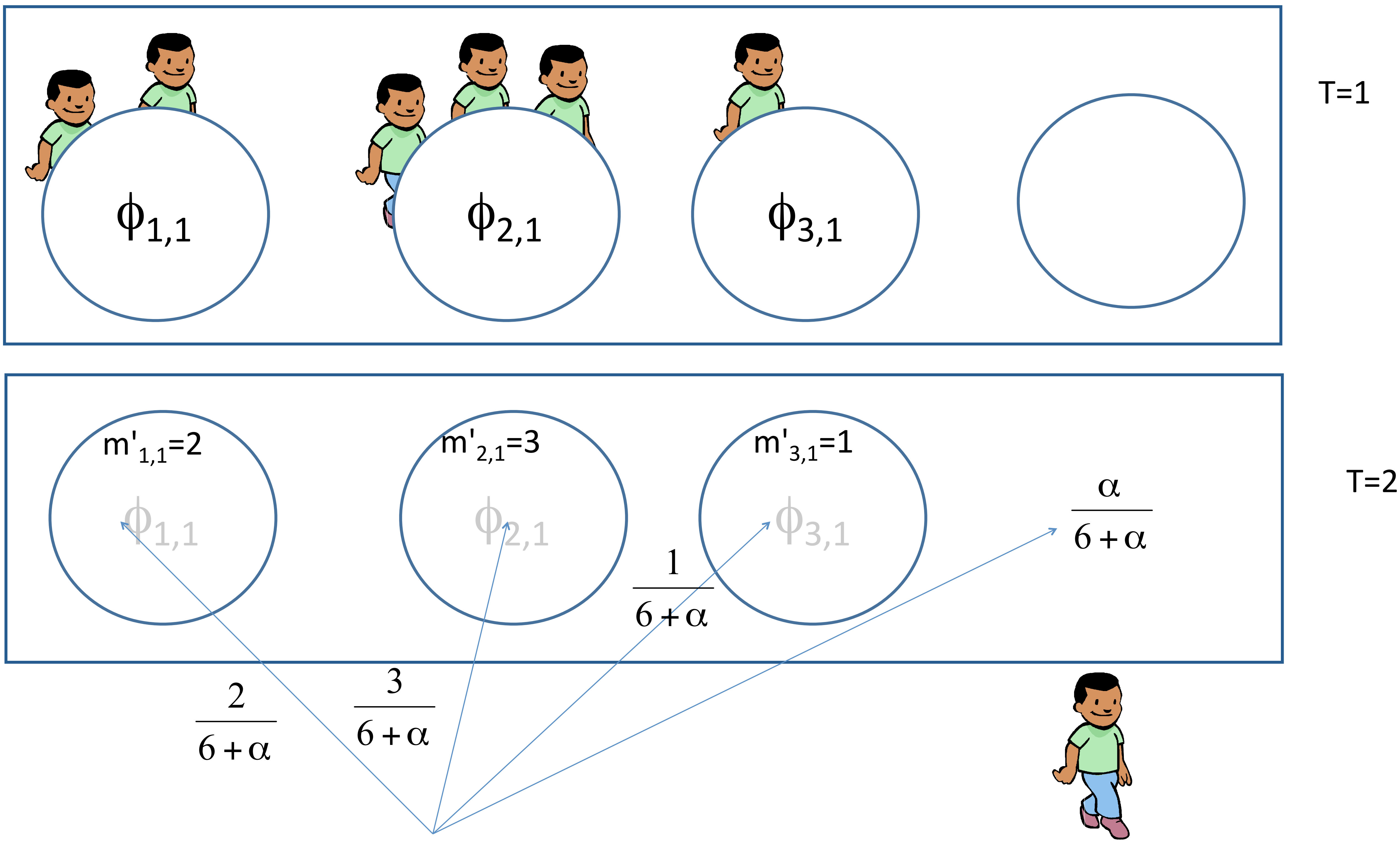
$m'_{1,1}=2$ $m'_{2,1}=3$ $m'_{3,1}=1$

T=2

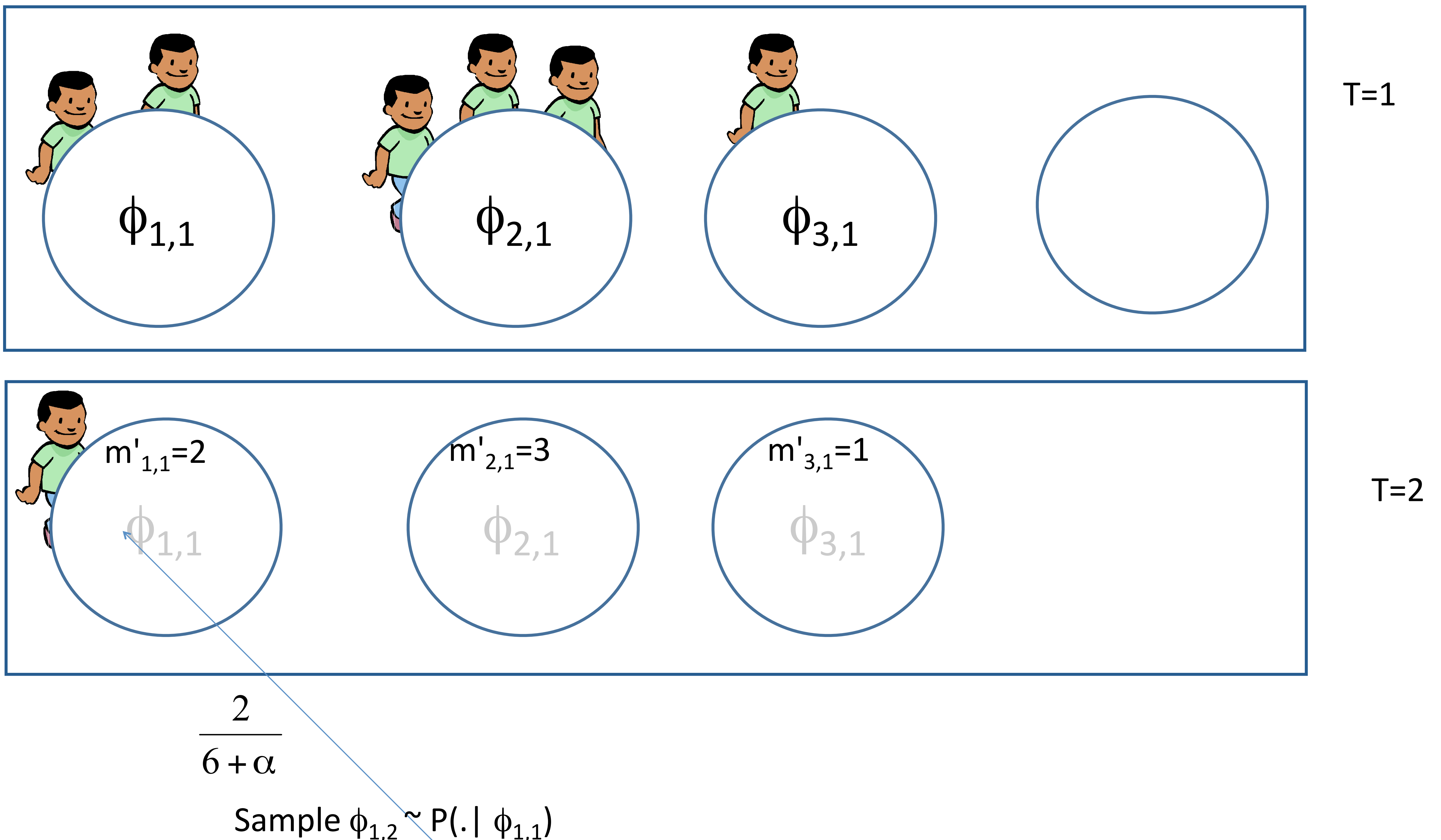
Recurrent Chinese Restaurant Process



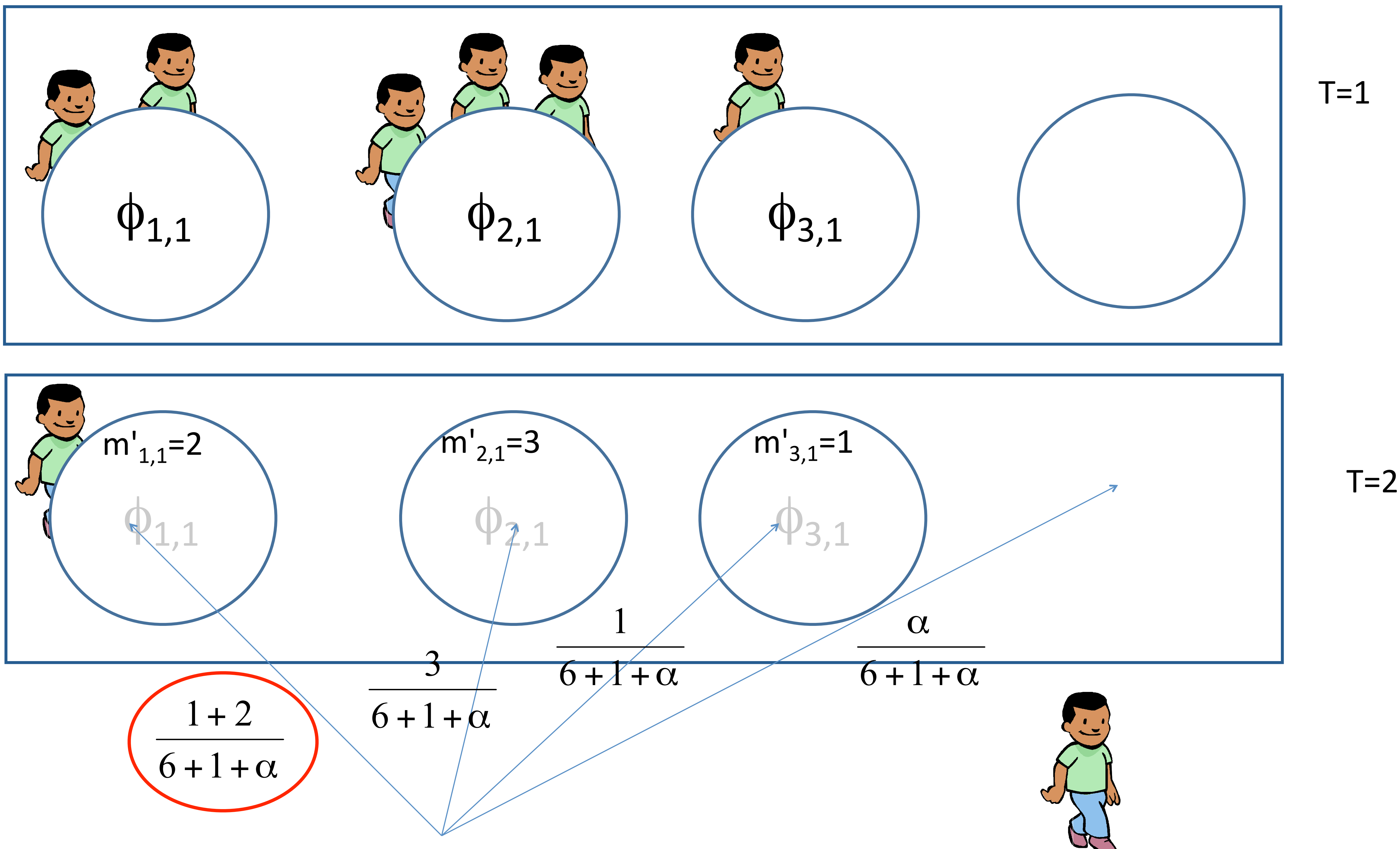
Recurrent Chinese Restaurant Process



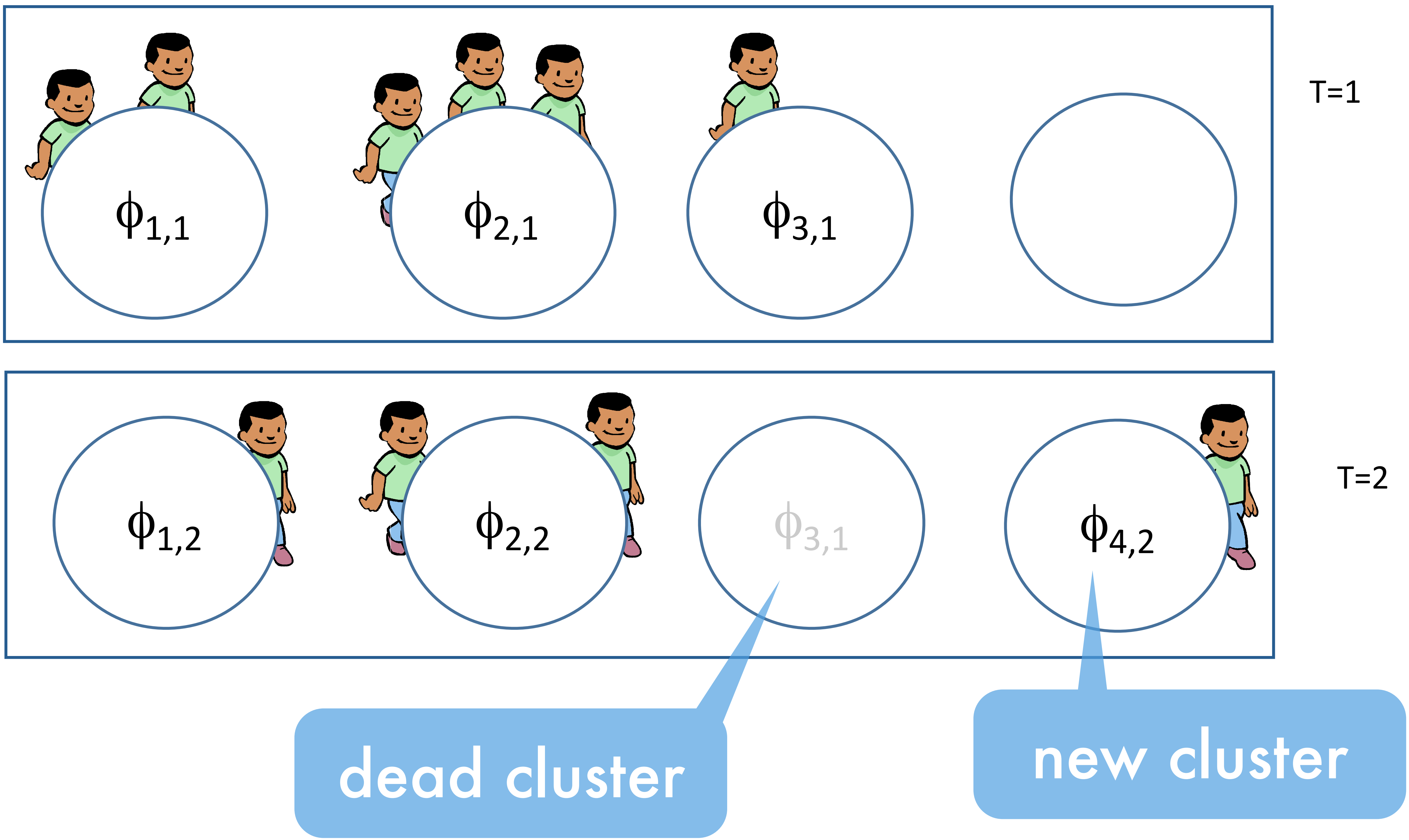
Recurrent Chinese Restaurant Process



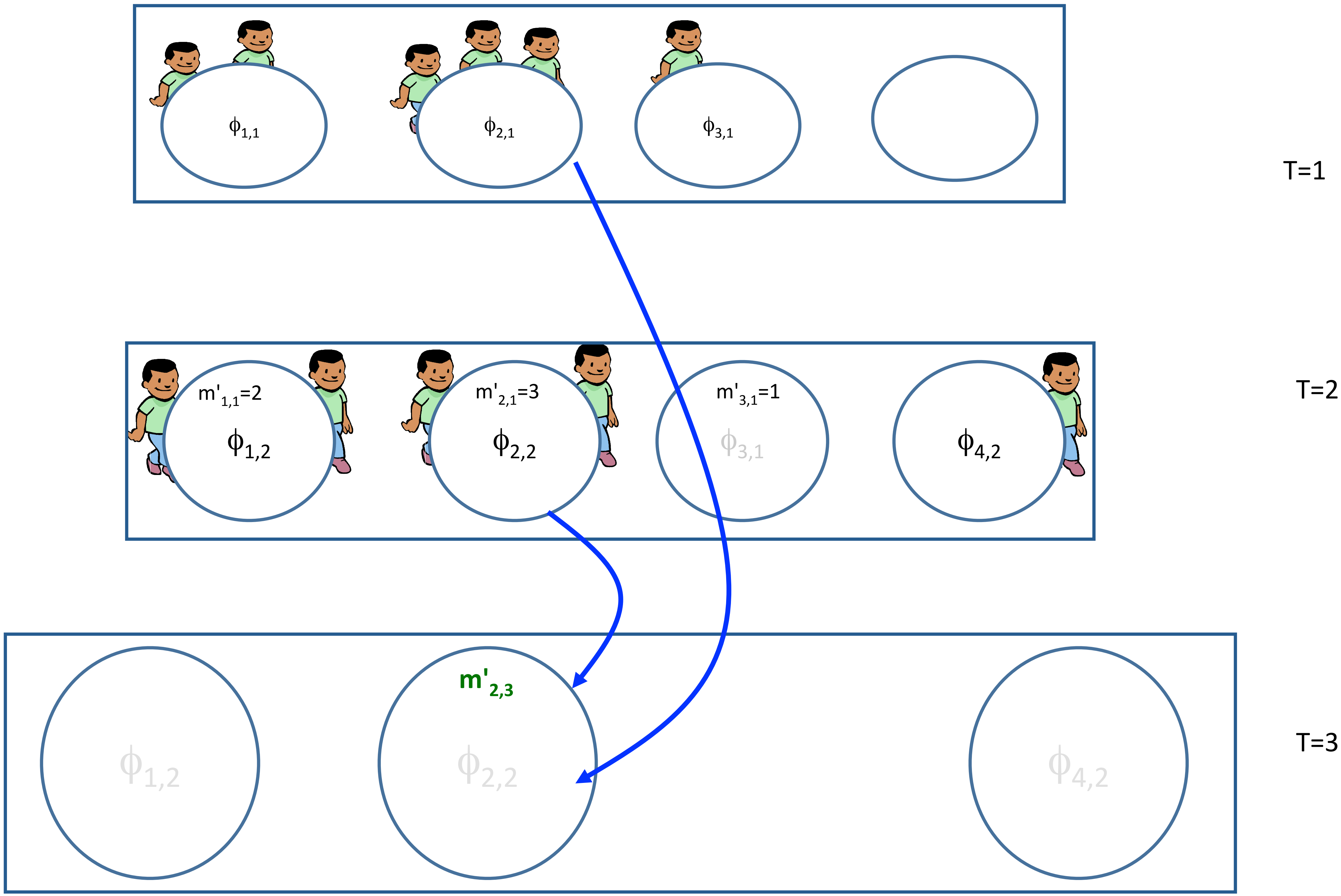
Recurrent Chinese Restaurant Process



Recurrent Chinese Restaurant Process



Longer History



TDPM Generative Power

DPM

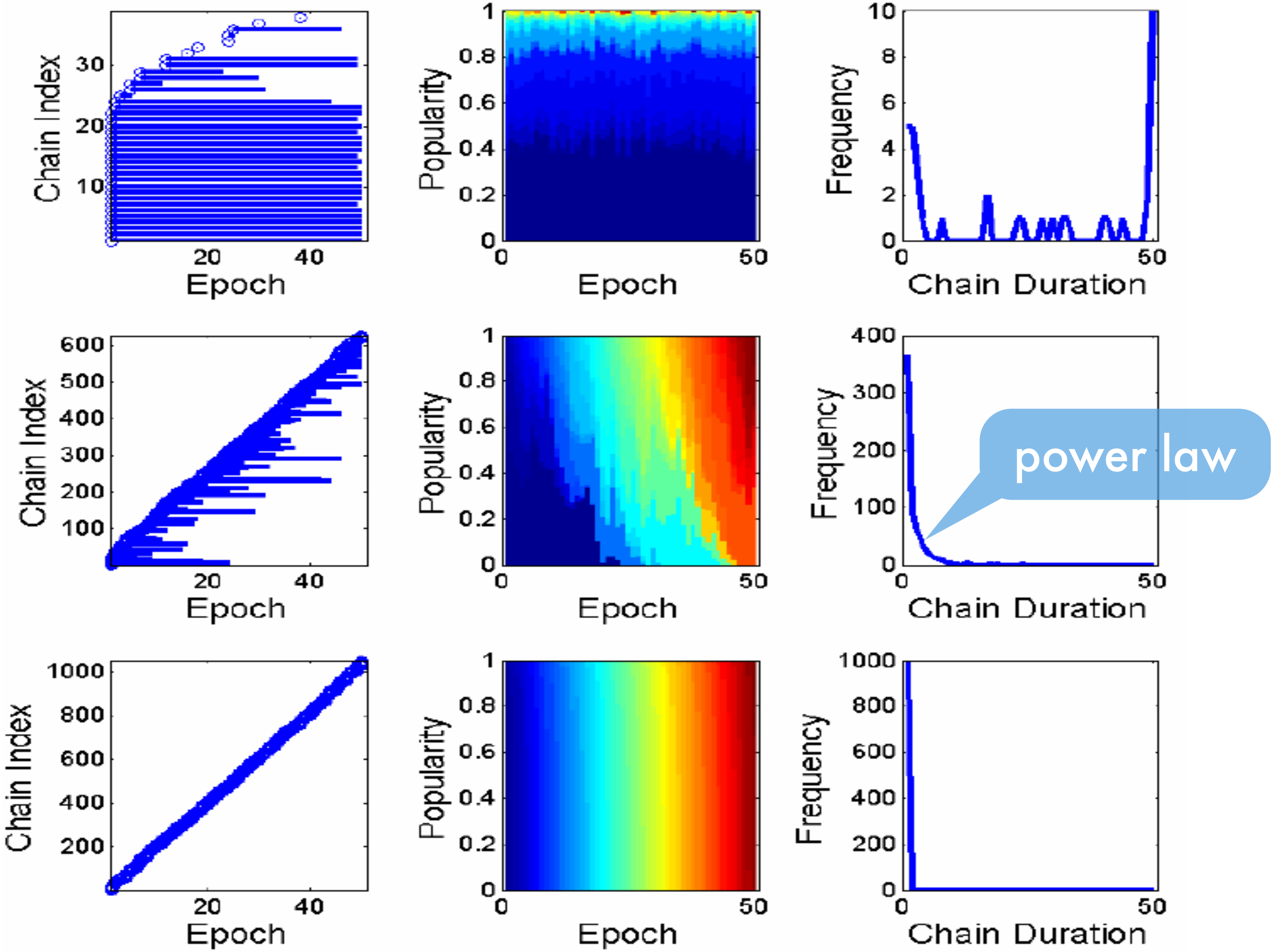
W= ∞
λ = ∞

TDPM

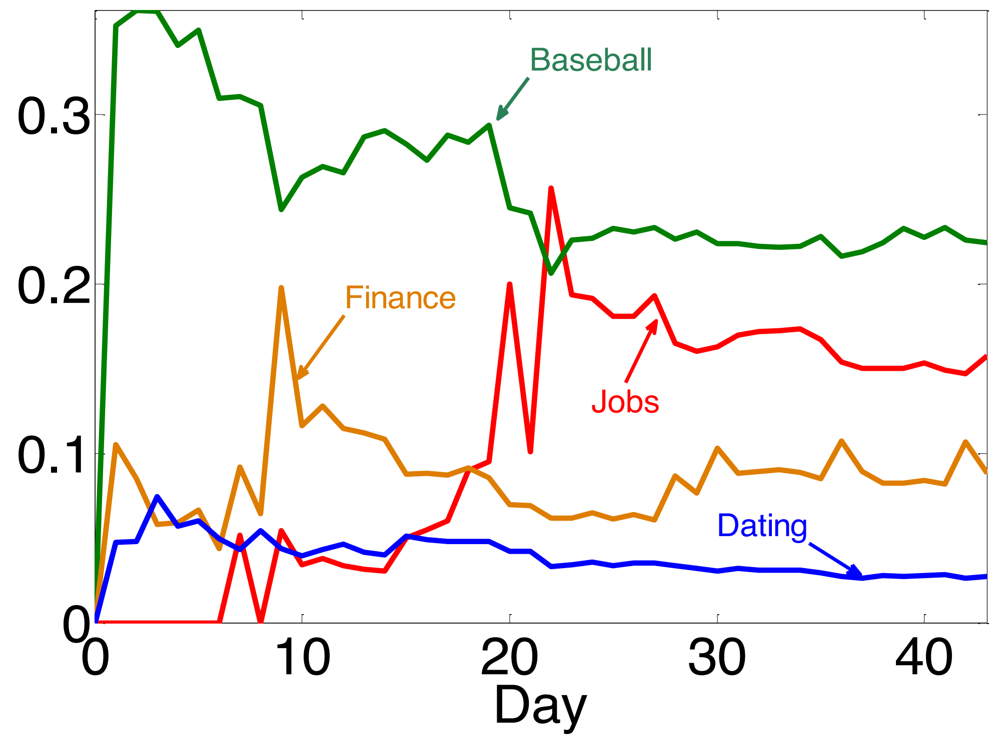
W=4
λ = .4

Independent DPMs

W= 0
λ = ? (any)

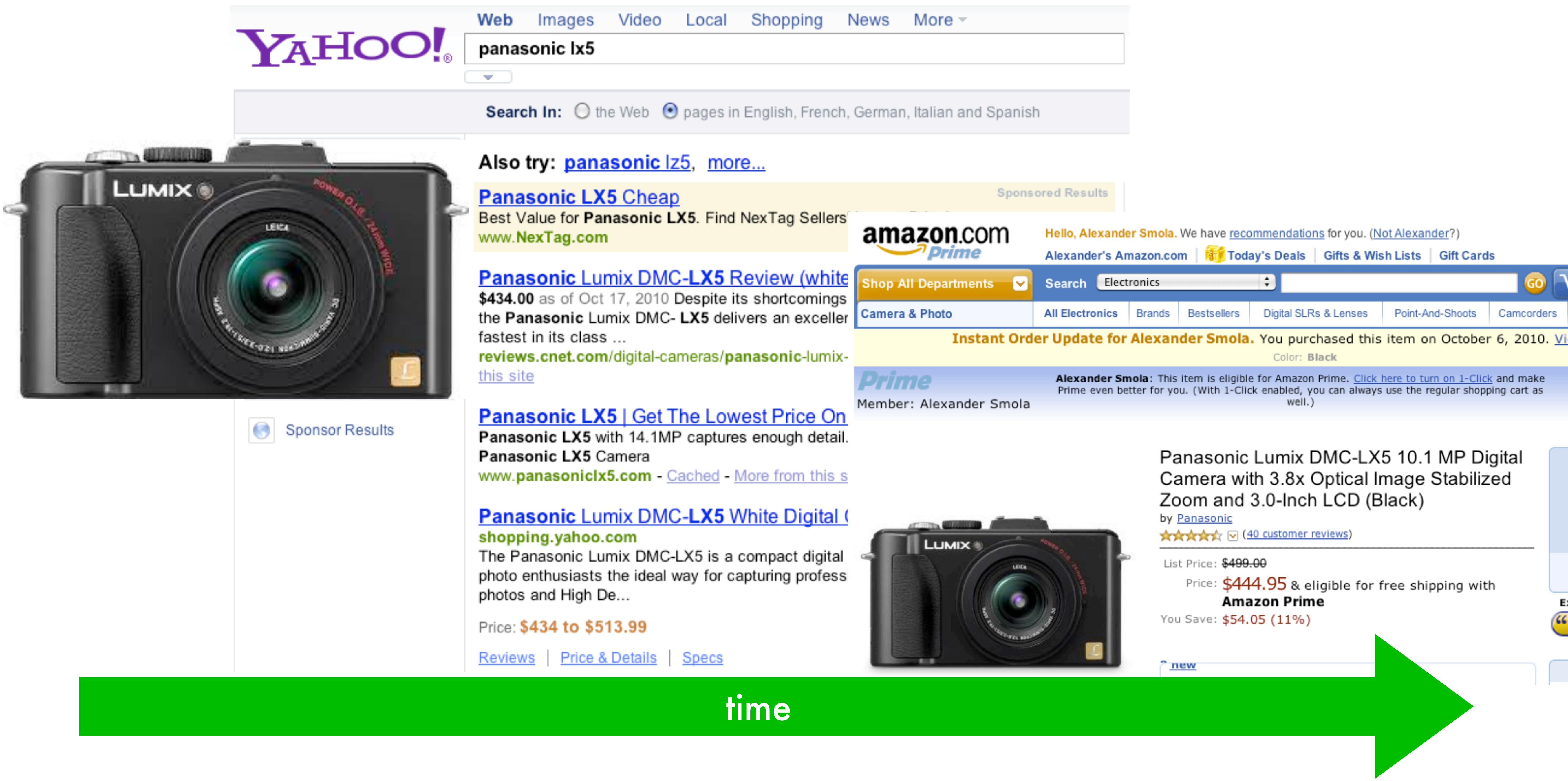


User modeling

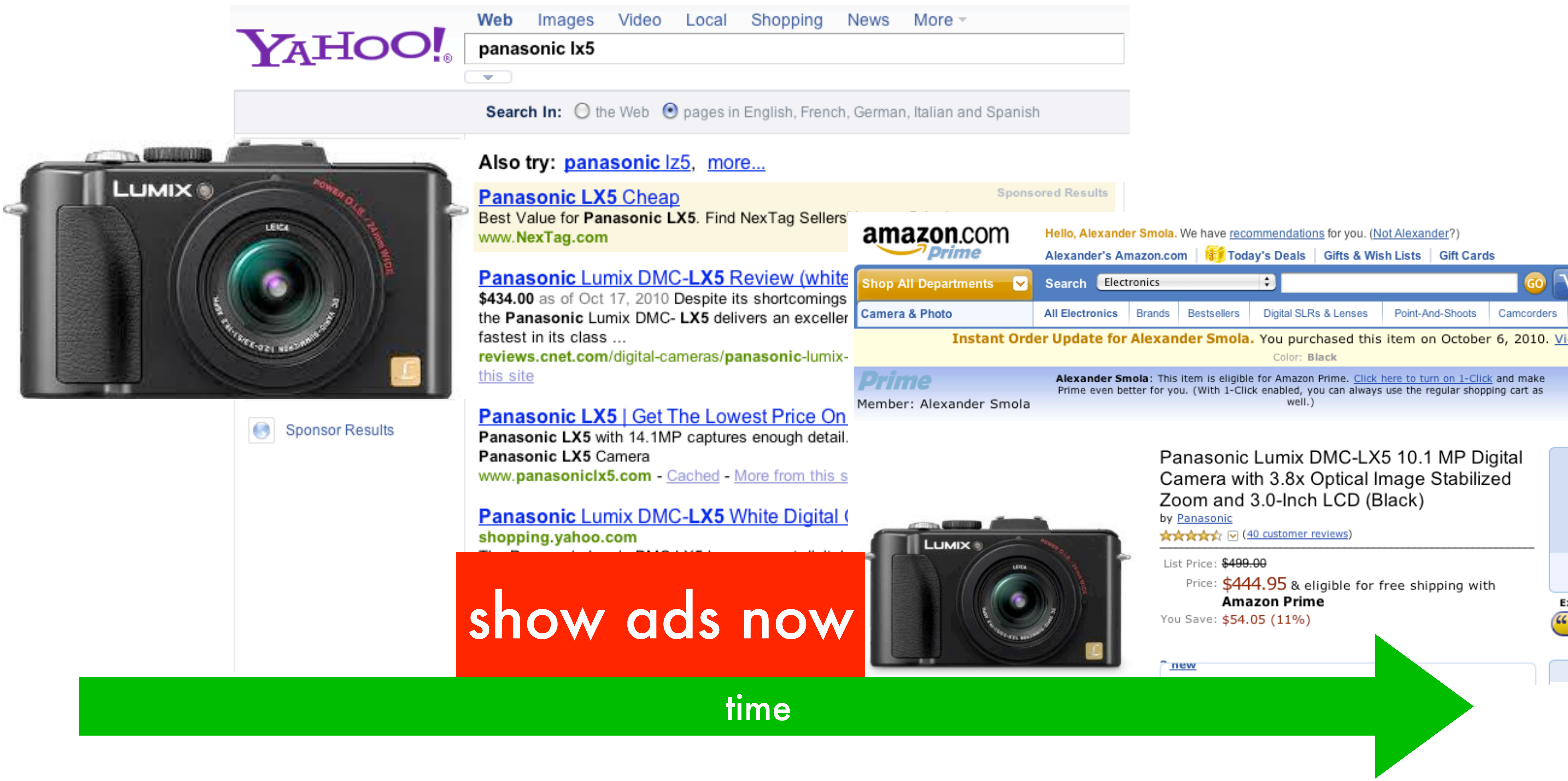


Buying a camera

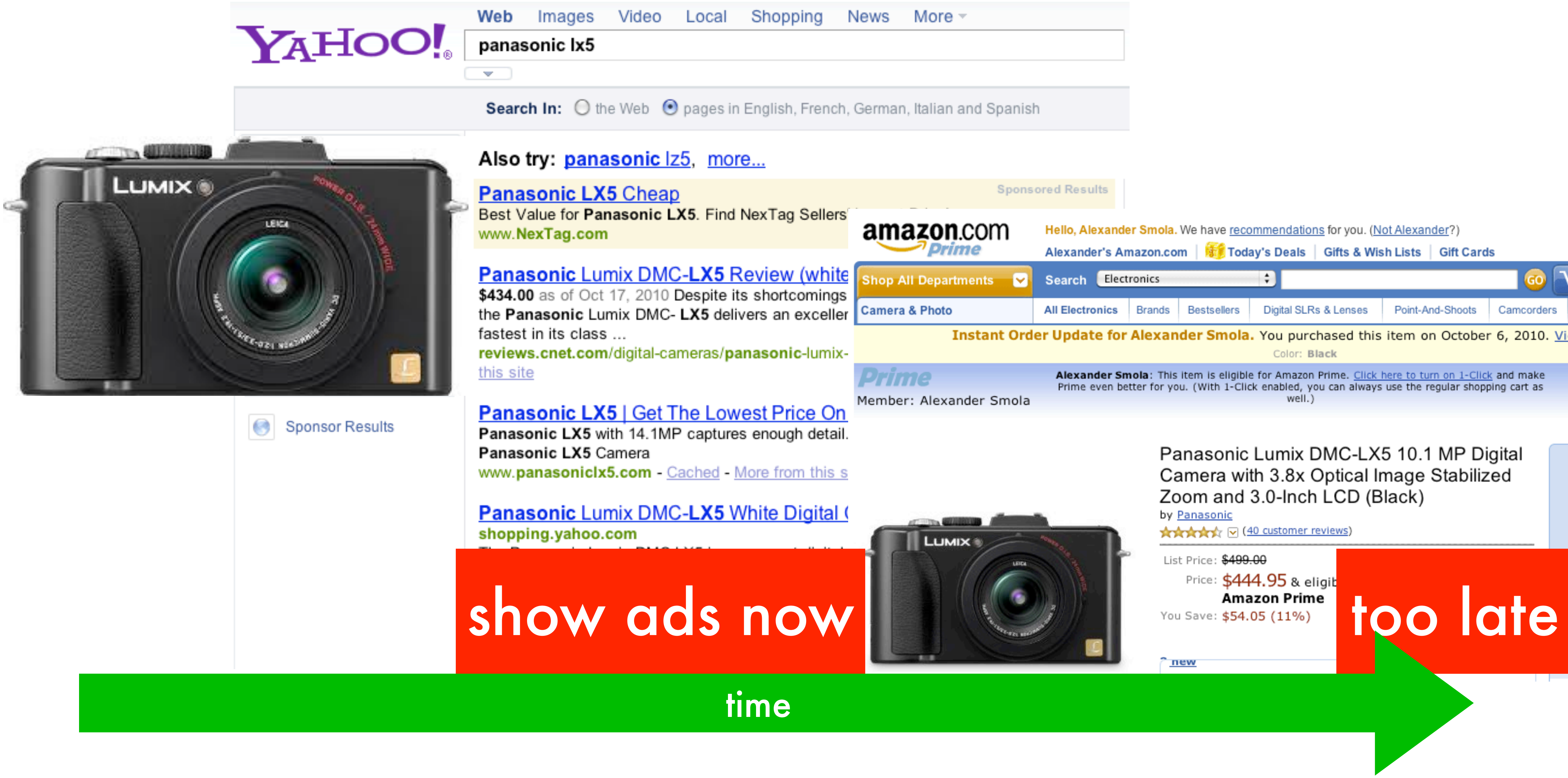
Buying a camera



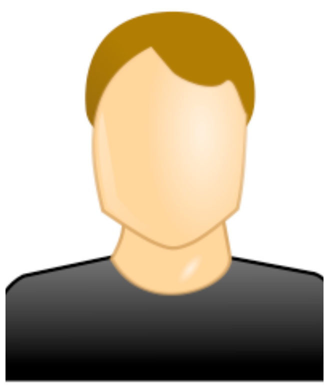
Buying a camera



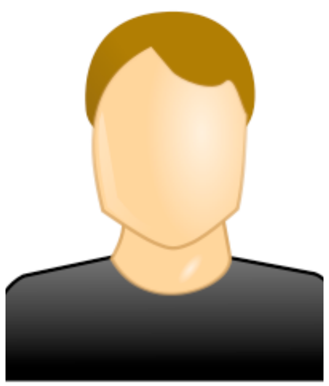
Buying a camera

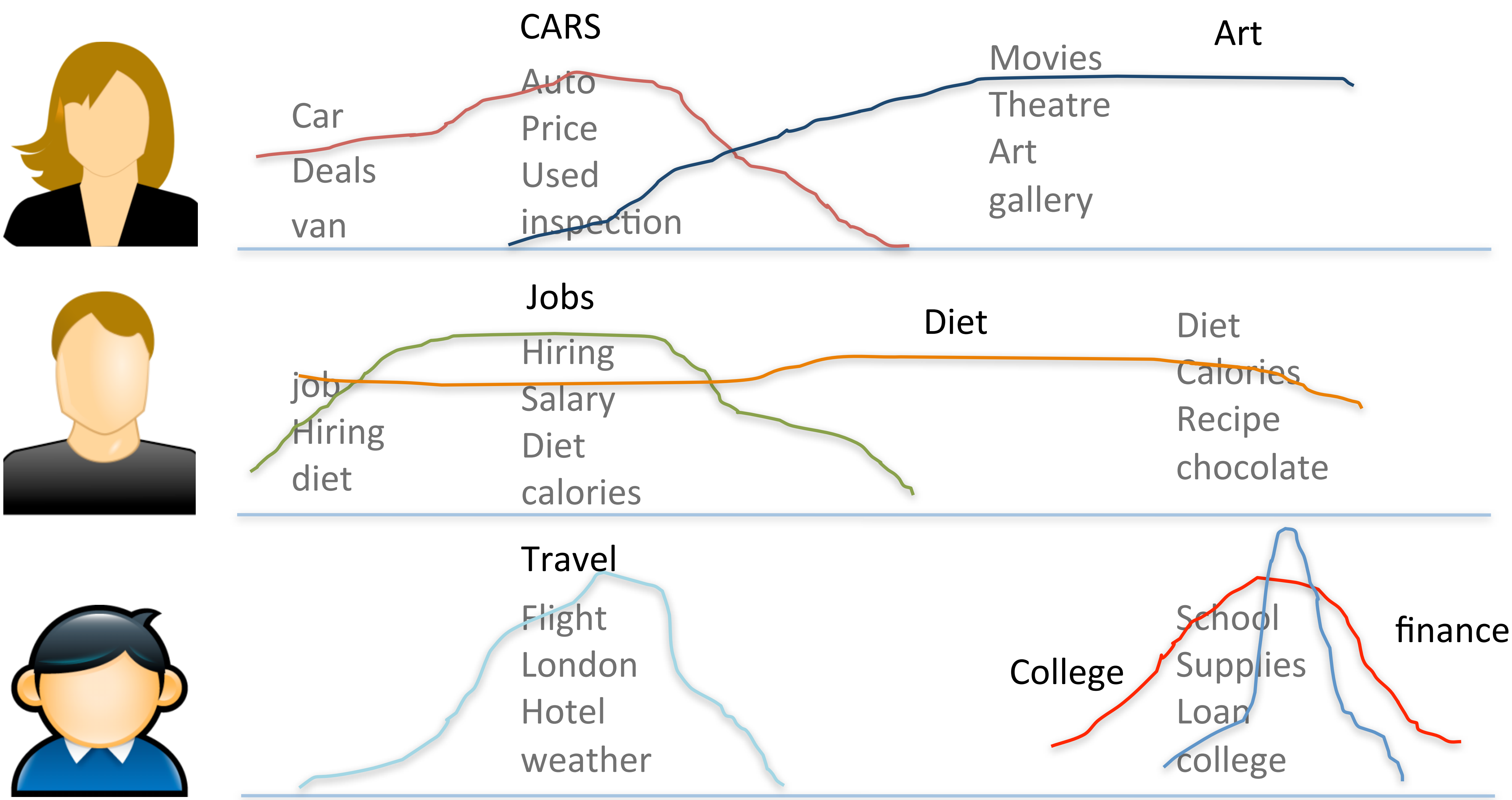


Car
Deals
van



job
Hiring
diet

	Car Deals van	Auto Price Used inspection	Movies Theatre Art gallery	
	job Hiring diet	Hiring Salary Diet calories		Diet Calories Recipe chocolate
		Flight London Hotel weather		School Supplies Loan college


CARS
Art
Car
Deals
van
Auto
Price
Used
inspection
Movies
Theatre
Art
gallery
Jobs
Diet
job
Hiring
diet
Hiring
Salary
Diet
calories
Diet
Calories
Recipe
chocolate
Travel
Flight
London
Hotel
weather
College
School
Supplies
Loan
college
finance

User modeling

Input

- Queries issued by the user or Tags of watched content
- Snippet of page examined by user
- Time stamp of each action (day resolution)

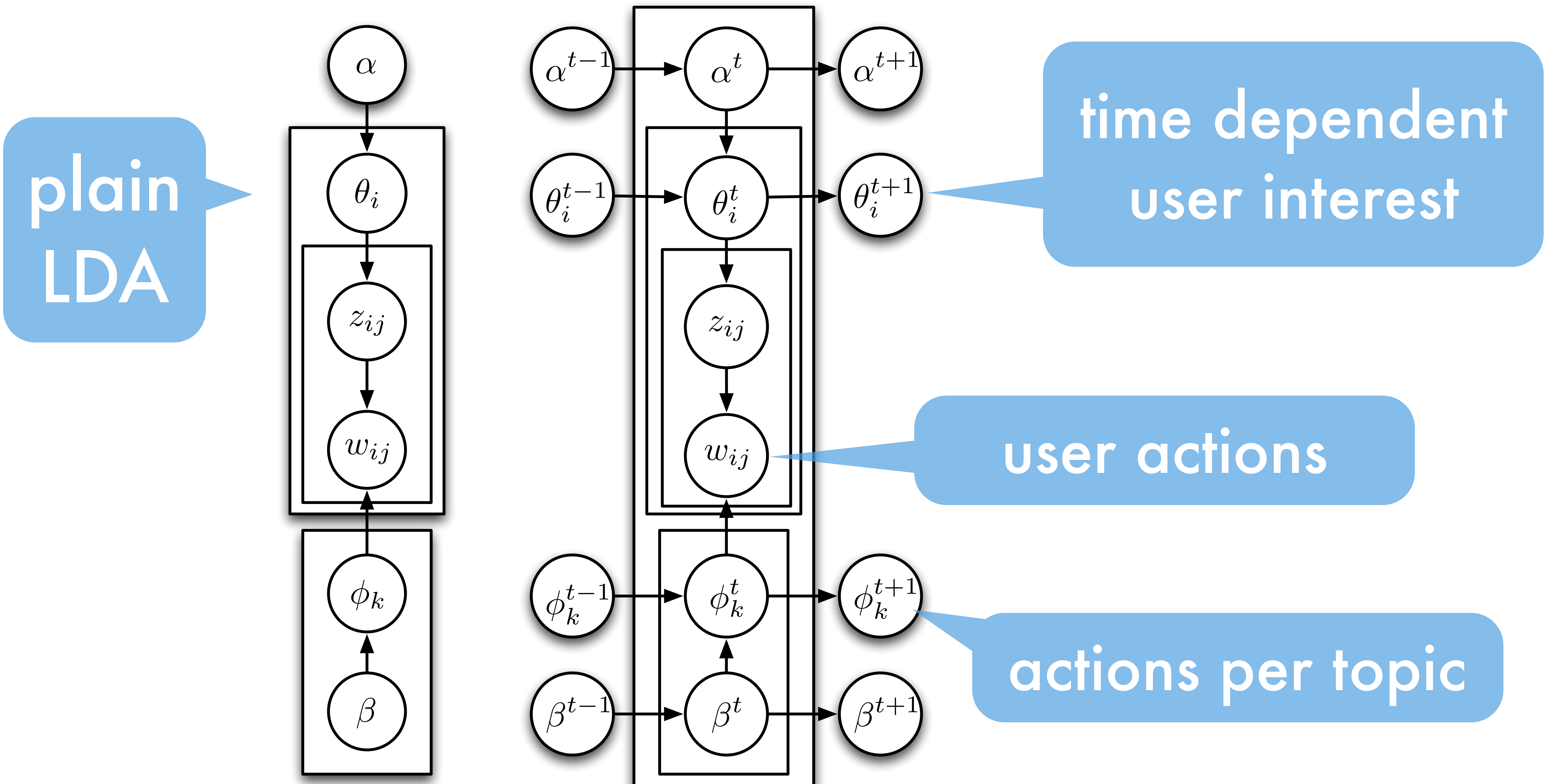
Output

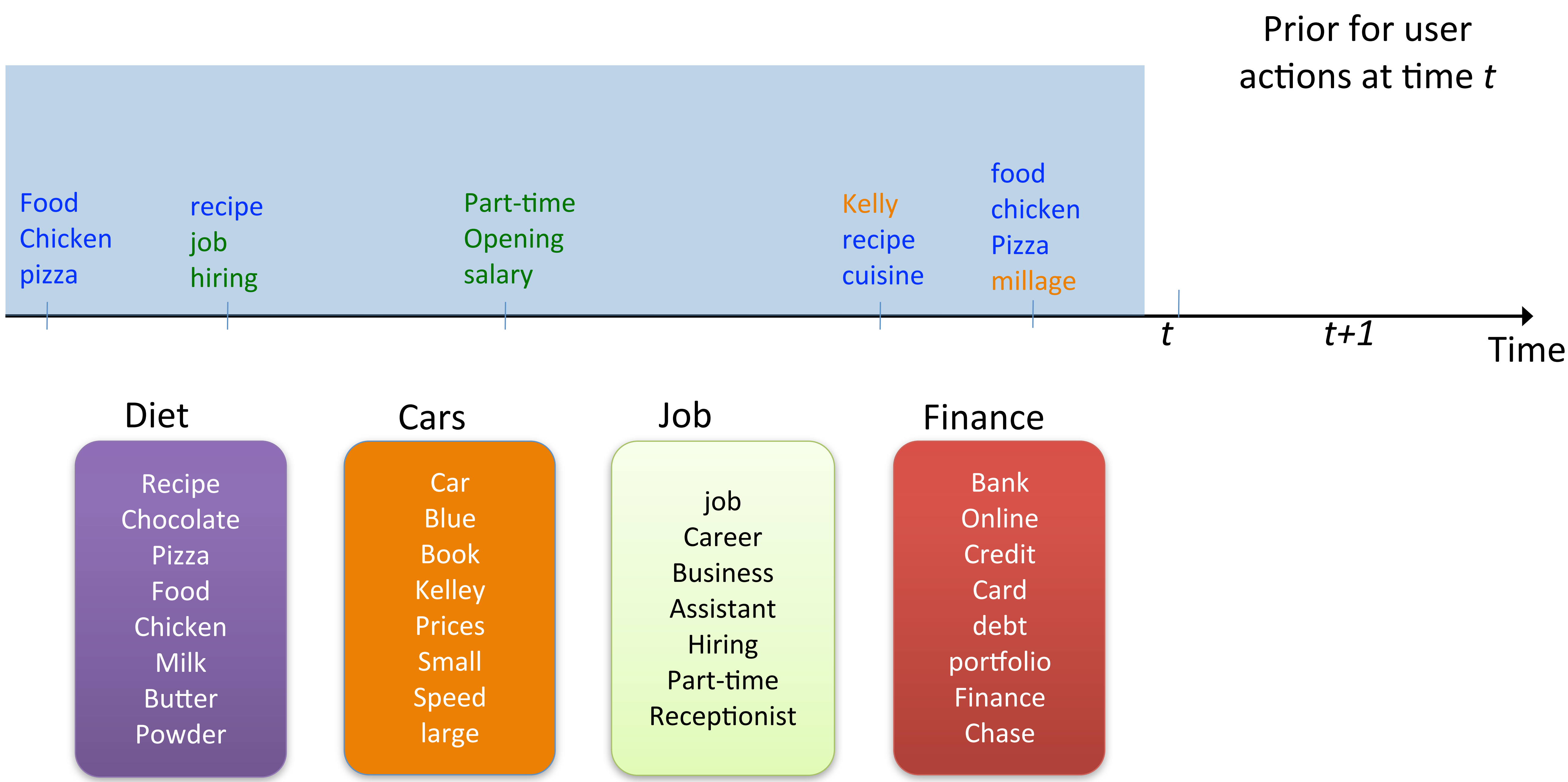
- Users' daily distribution over intents
- Dynamic intent representation

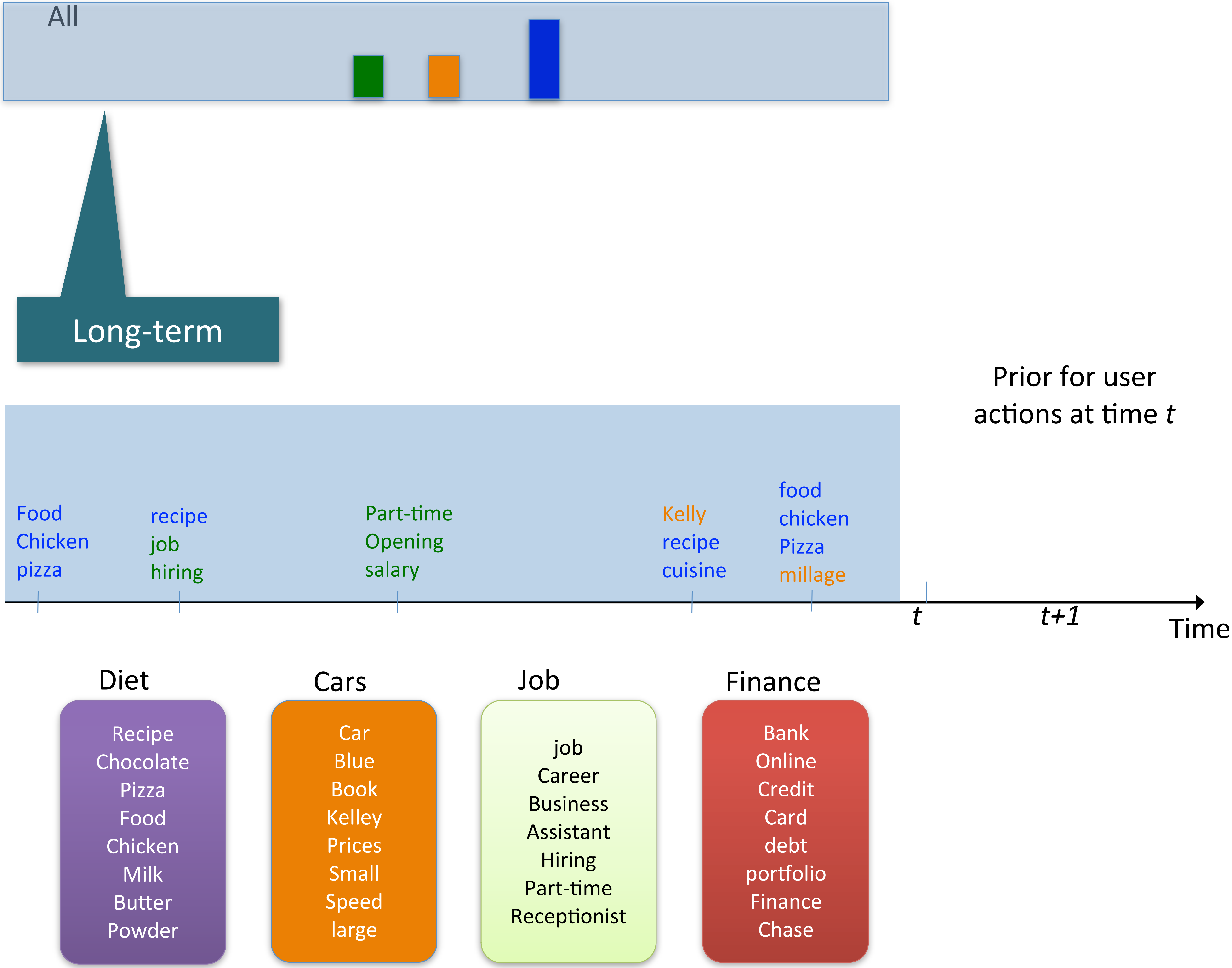
Time dependent models

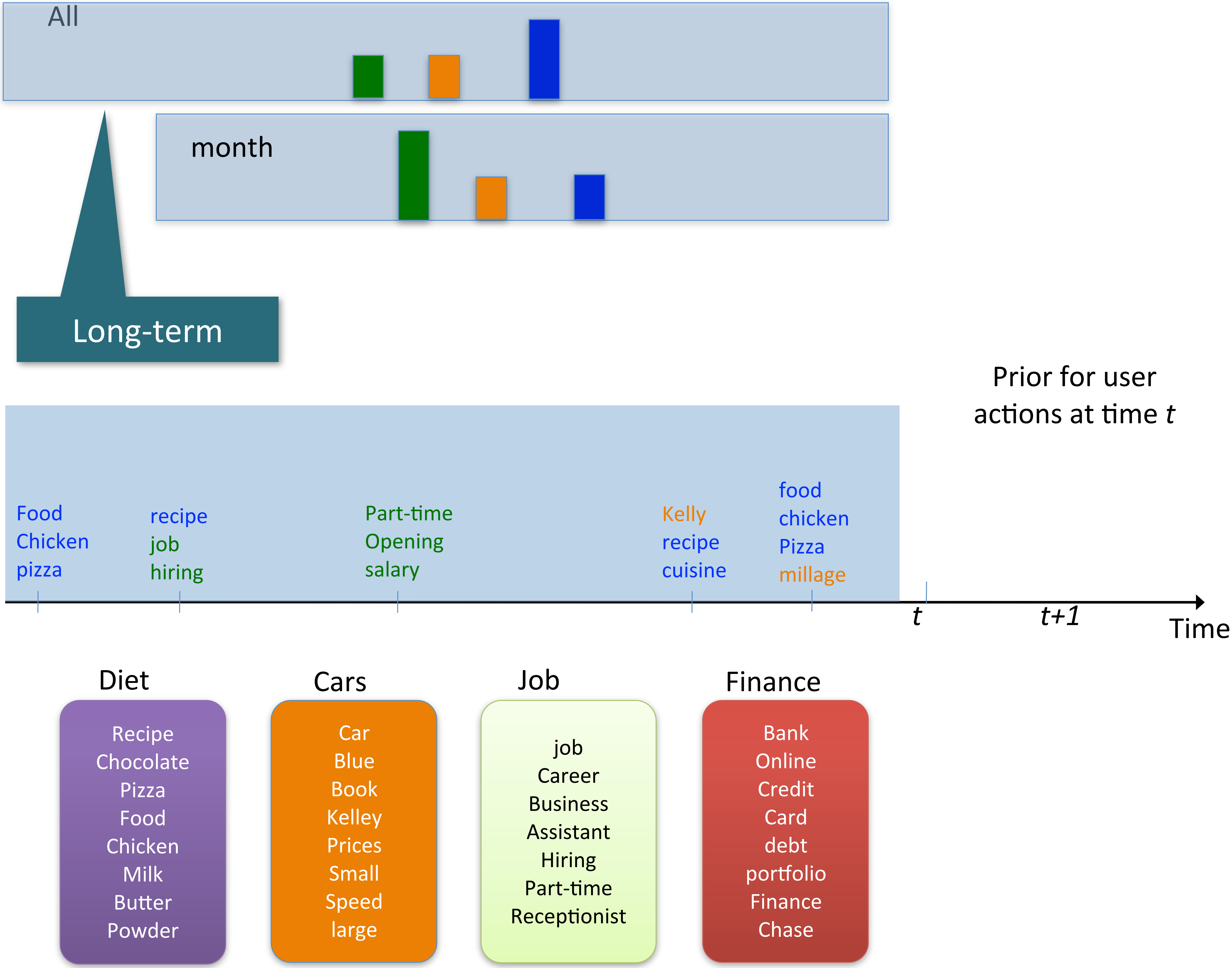
- LDA for topical model of users where
  - User interest distribution changes over time
  - Topics change over time
- This is like a Kalman filter except that
  - Don’t know what to track (a priori)
  - Can’t afford a Rauch-Tung-Striebel smoother
  - Much more messy than plain LDA

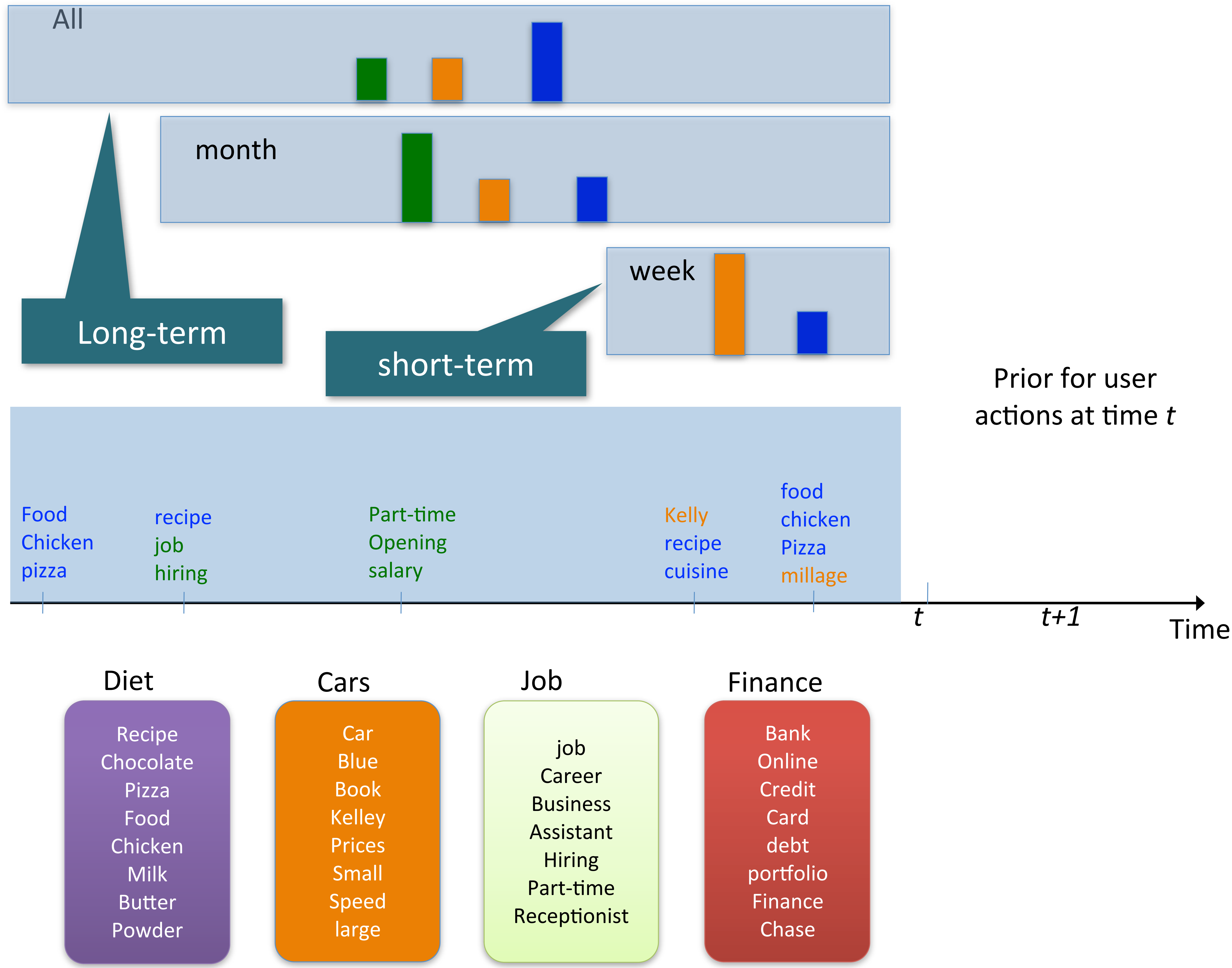
Graphical Model



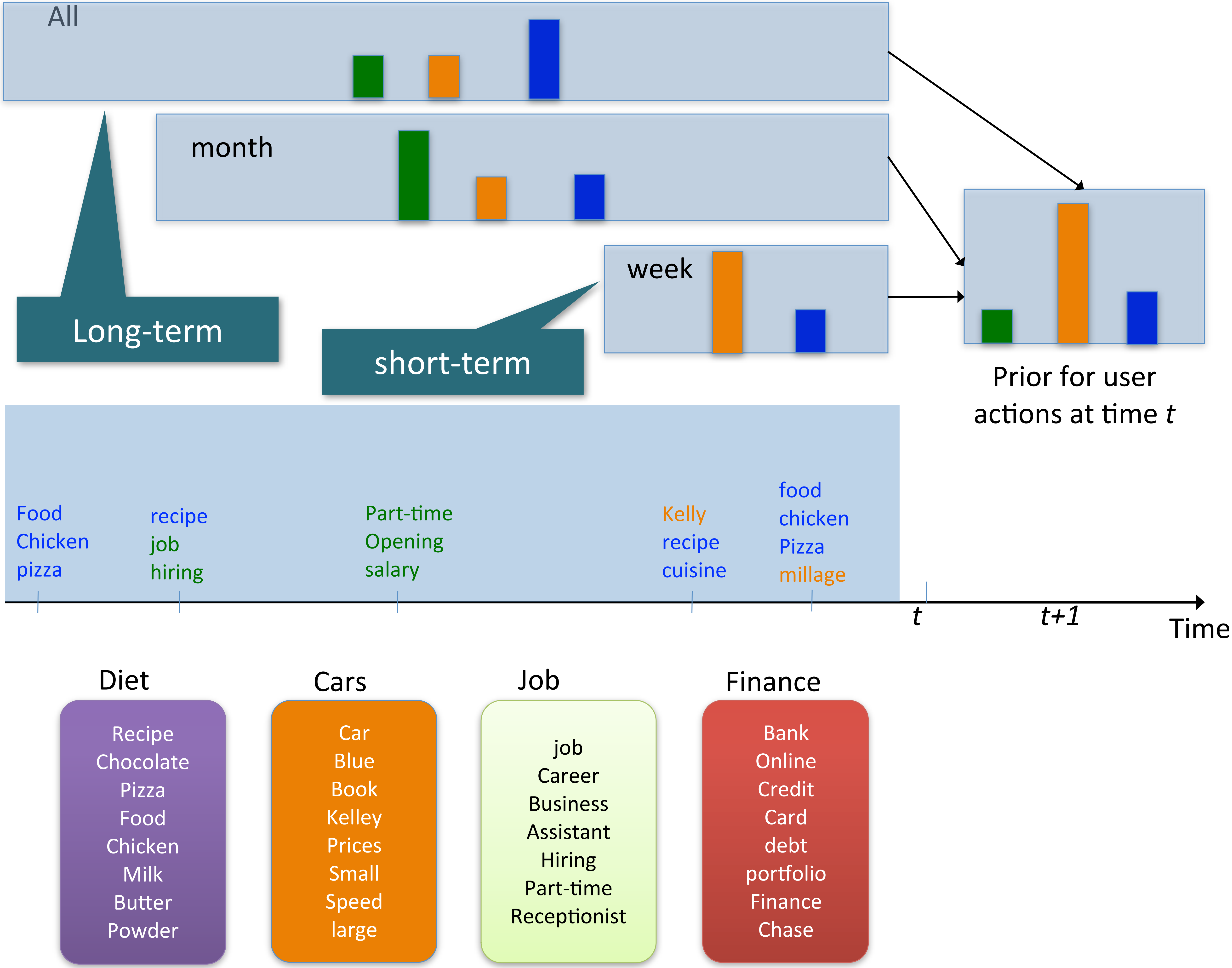

Prior for user actions at time t
Food Chicken pizza
recipe job hiring
Part-time Opening salary
Kelly recipe cuisine
food chicken Pizza millage
t
t+1
Time
Diet
Recipe Chocolate Pizza Food Chicken Milk Butter Powder
Cars
Car Blue Book Kelley Prices Small Speed large
Job
job Career Business Assistant Hiring Part-time Receptionist
Finance
Bank Online Credit Card debt portfolio Finance Chase

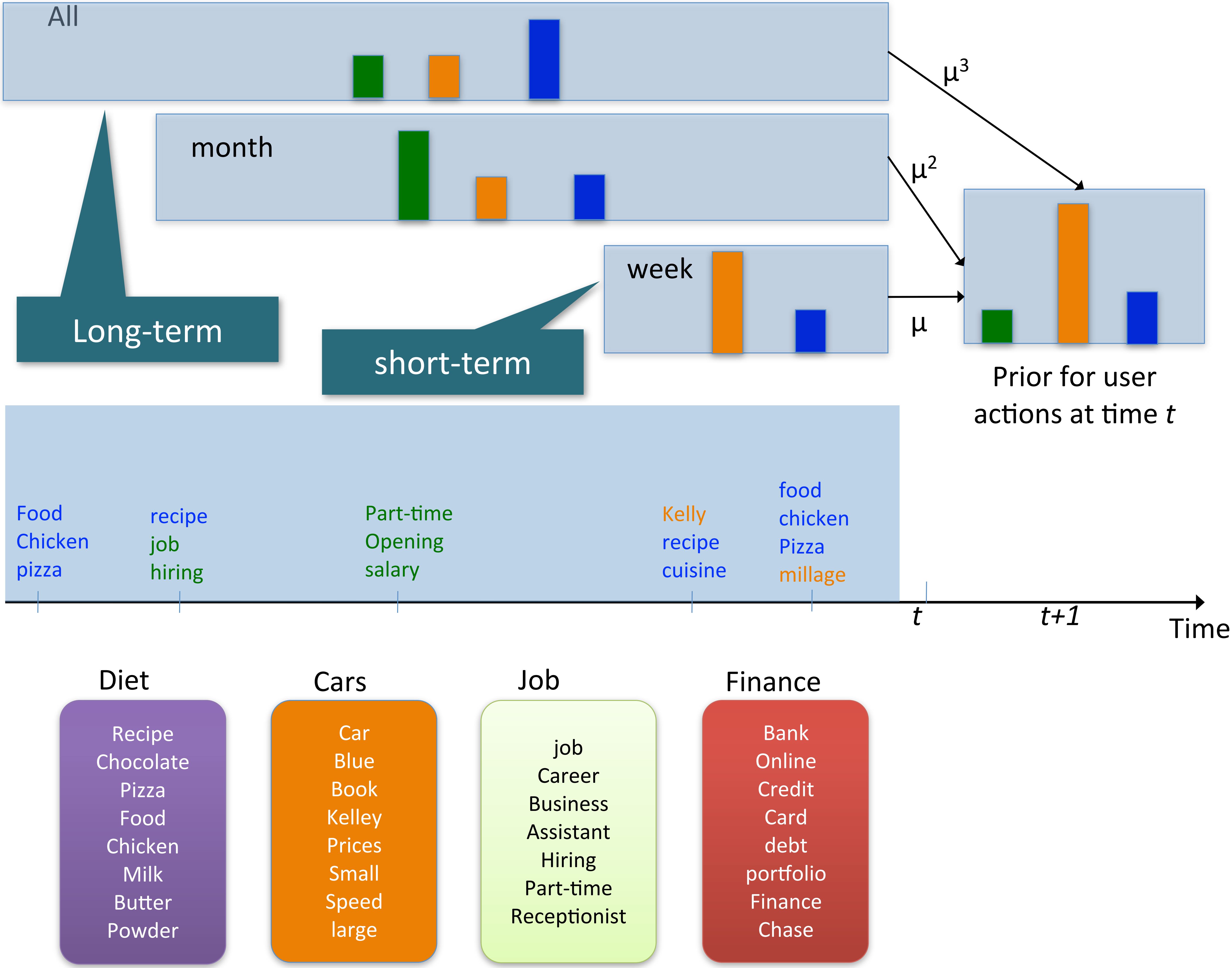

All
Long-term
Prior for user actions at time *t*
Food
Chicken
pizza
recipe
job
hiring
Part-time
Opening
salary
Kelly
recipe
cuisine
food
chicken
Pizza
millage
t
t+1
Time
Diet
Recipe
Chocolate
Pizza
Food
Chicken
Milk
Butter
Powder
Cars
Car
Blue
Book
Kelley
Prices
Small
Speed
large
Job
job
Career
Business
Assistant
Hiring
Part-time
Receptionist
Finance
Bank
Online
Credit
Card
debt
portfolio
Finance
Chase


All
month
Long-term
Prior for user actions at time t
Food Chicken pizza
recipe job hiring
Part-time Opening salary
Kelly recipe cuisine
food chicken Pizza millage
t
t+1
Time
Diet
Recipe Chocolate Pizza Food Chicken Milk Butter Powder
Cars
Car Blue Book Kelley Prices Small Speed large
Job
job Career Business Assistant Hiring Part-time Receptionist
Finance
Bank Online Credit Card debt portfolio Finance Chase

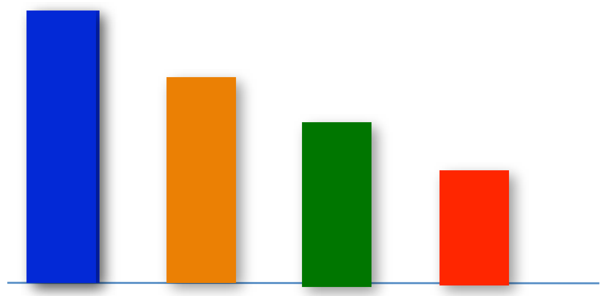


All
month
week
Long-term
short-term
Prior for user actions at time *t*
Food
Chicken
pizza
recipe
job
hiring
Part-time
Opening
salary
Kelly
recipe
cuisine
food
chicken
Pizza
millage
t
t+1
Time
Diet
Recipe
Chocolate
Pizza
Food
Chicken
Milk
Butter
Powder
Cars
Car
Blue
Book
Kelley
Prices
Small
Speed
large
Job
job
Career
Business
Assistant
Hiring
Part-time
Receptionist
Finance
Bank
Online
Credit
Card
debt
portfolio
Finance
Chase

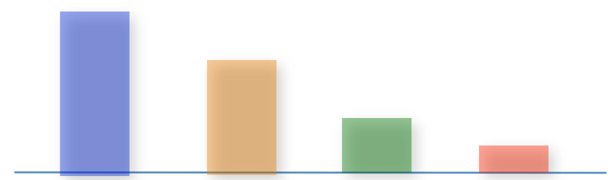

All
month
week
Long-term
short-term
Prior for user actions at time t
Food
Chicken
pizza
recipe
job
hiring
Part-time
Opening
salary
Kelly
recipe
cuisine
food
chicken
Pizza
millage
t
t+1
Time
Diet
Recipe
Chocolate
Pizza
Food
Chicken
Milk
Butter
Powder
Cars
Car
Blue
Book
Kelley
Prices
Small
Speed
large
Job
job
Career
Business
Assistant
Hiring
Part-time
Receptionist
Finance
Bank
Online
Credit
Card
debt
portfolio
Finance
Chase


All
month
week
μ^3
μ^2
μ
Long-term
short-term
Prior for user actions at time t
Food
Chicken
pizza
recipe
job
hiring
Part-time
Opening
salary
Kelly
recipe
cuisine
food
chicken
Pizza
millage
t
$t+1$
Time
Diet
Recipe
Chocolate
Pizza
Food
Chicken
Milk
Butter
Powder
Cars
Car
Blue
Book
Kelley
Prices
Small
Speed
large
Job
job
Career
Business
Assistant
Hiring
Part-time
Receptionist
Finance
Bank
Online
Credit
Card
debt
portfolio
Finance
Chase

At time t



At time t+1

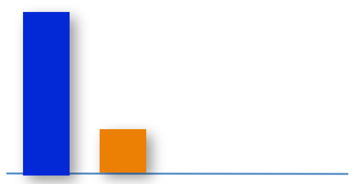


Recipe
Chocolate
Pizza
Food
Chicken
Milk
Butter
Powder

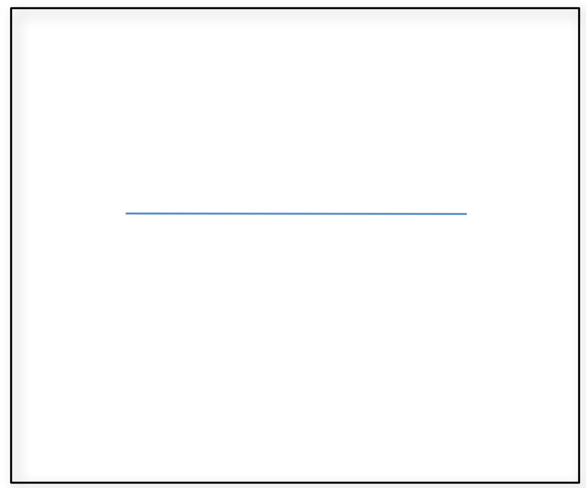
Car
Altima
Accord
Blue
Book
Kelley
Prices
Small
Speed

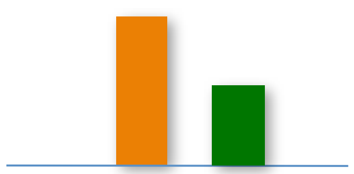
job
Career
Business
Assistant
Hiring
Part-time
Receptionist

Bank
Online
Credit
Card
debt
portfolio
Finance
Chase

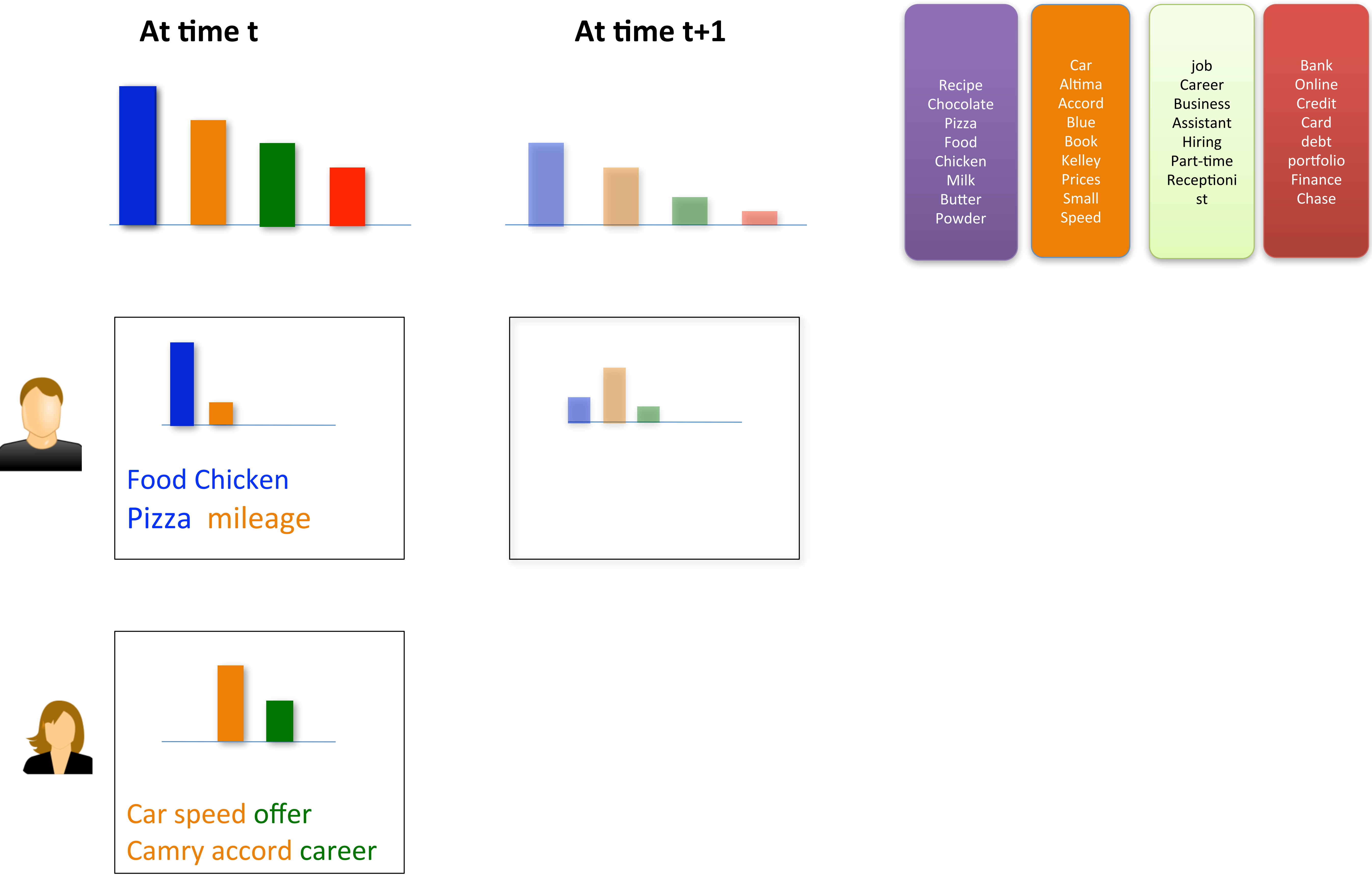


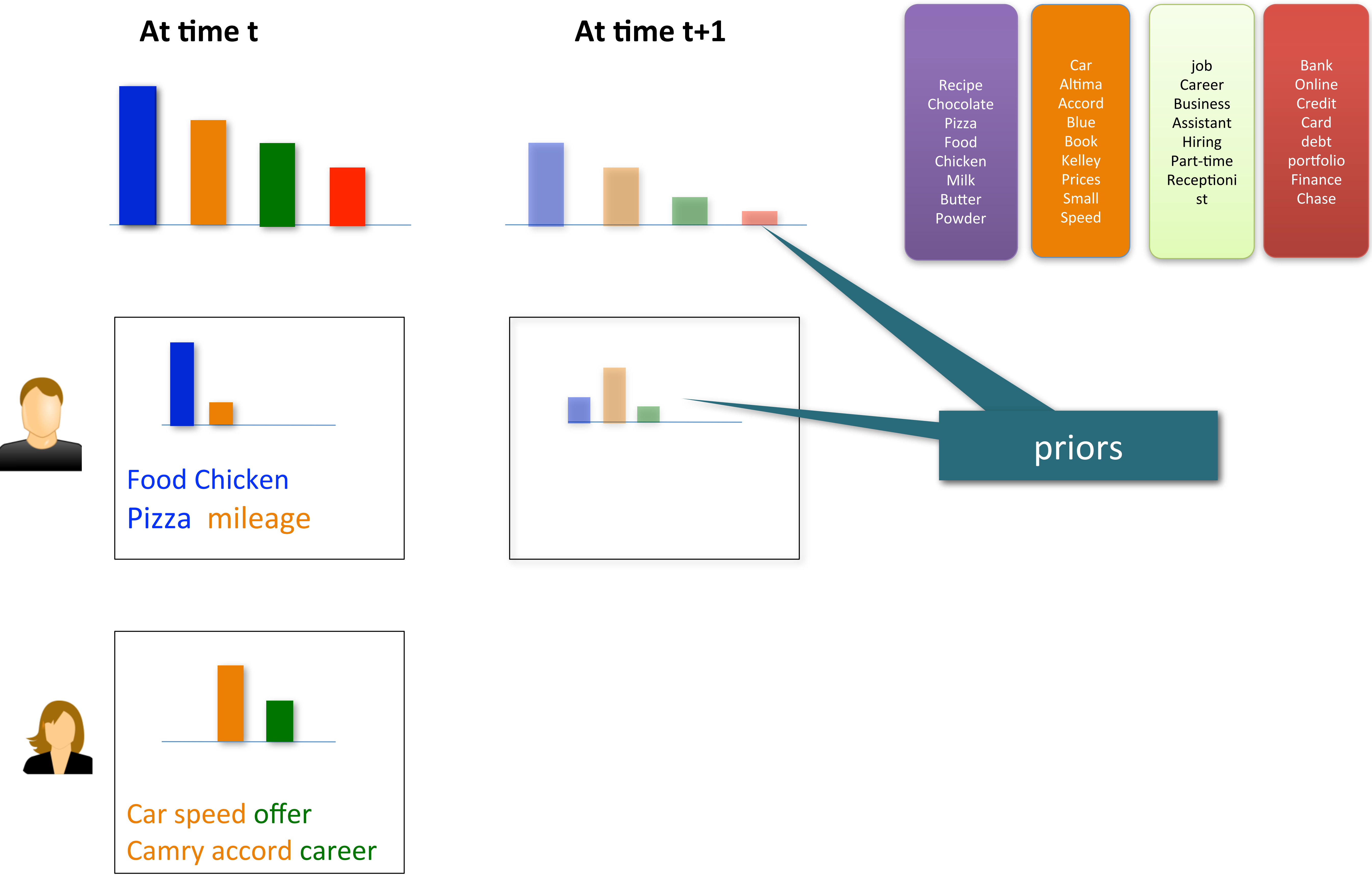
Food Chicken
Pizza mileage

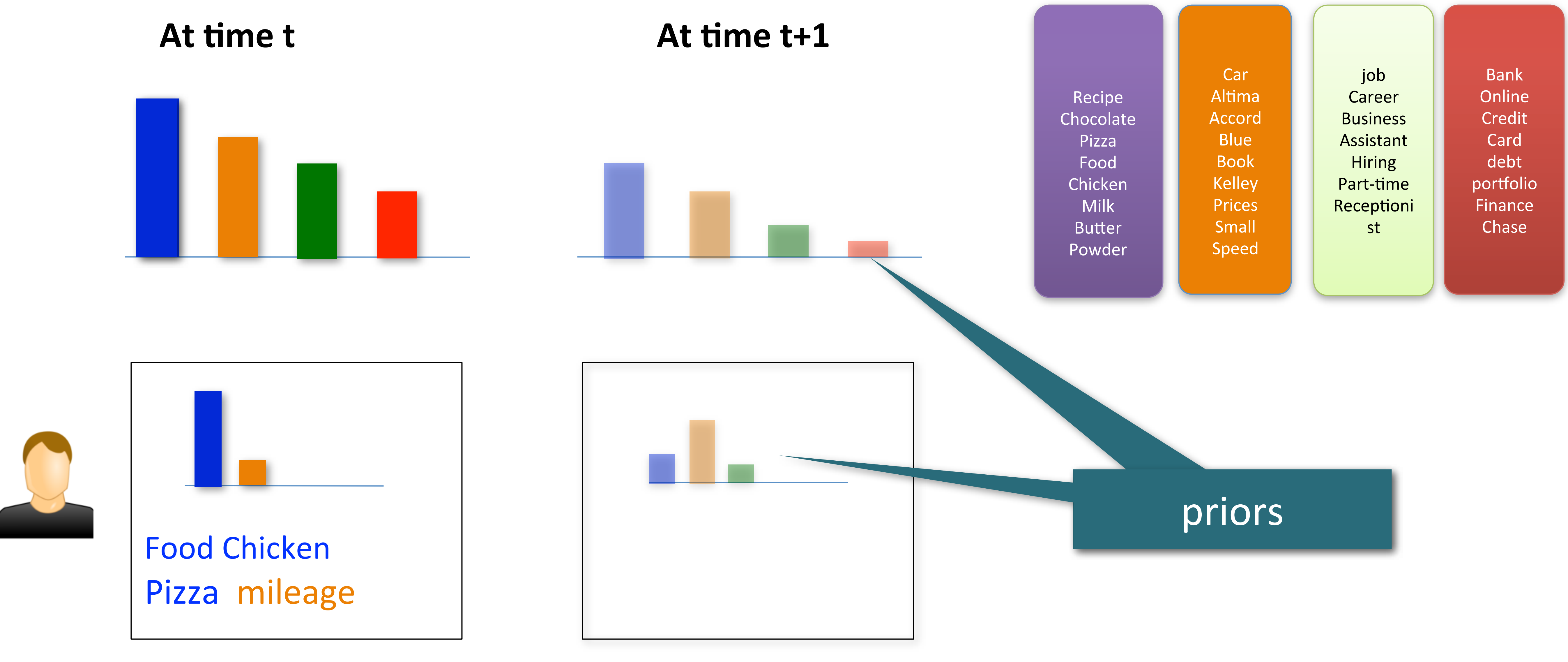


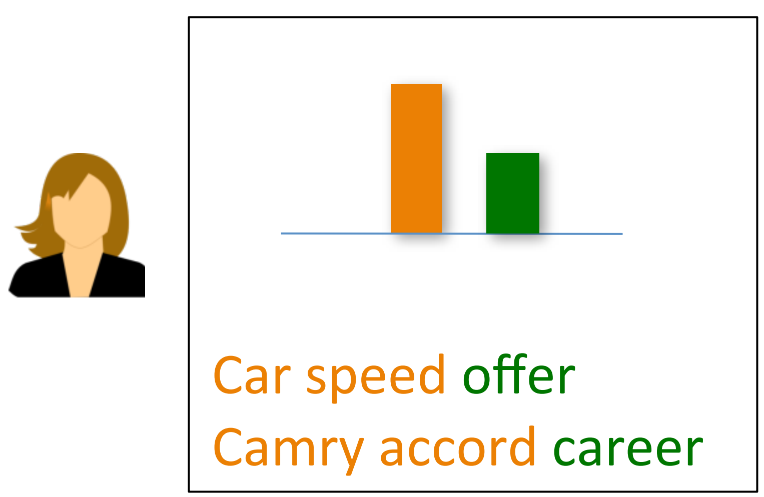


Car speed offer
Camry accord career


At time t
At time t+1
Recipe
Chocolate
Pizza
Food
Chicken
Milk
Butter
Powder
Car
Altima
Accord
Blue
Book
Kelley
Prices
Small
Speed
job
Career
Business
Assistant
Hiring
Part-time
Receptioni
st
Bank
Online
Credit
Card
debt
portfolio
Finance
Chase
Food Chicken
Pizza mileage
Car speed offer
Camry accord career

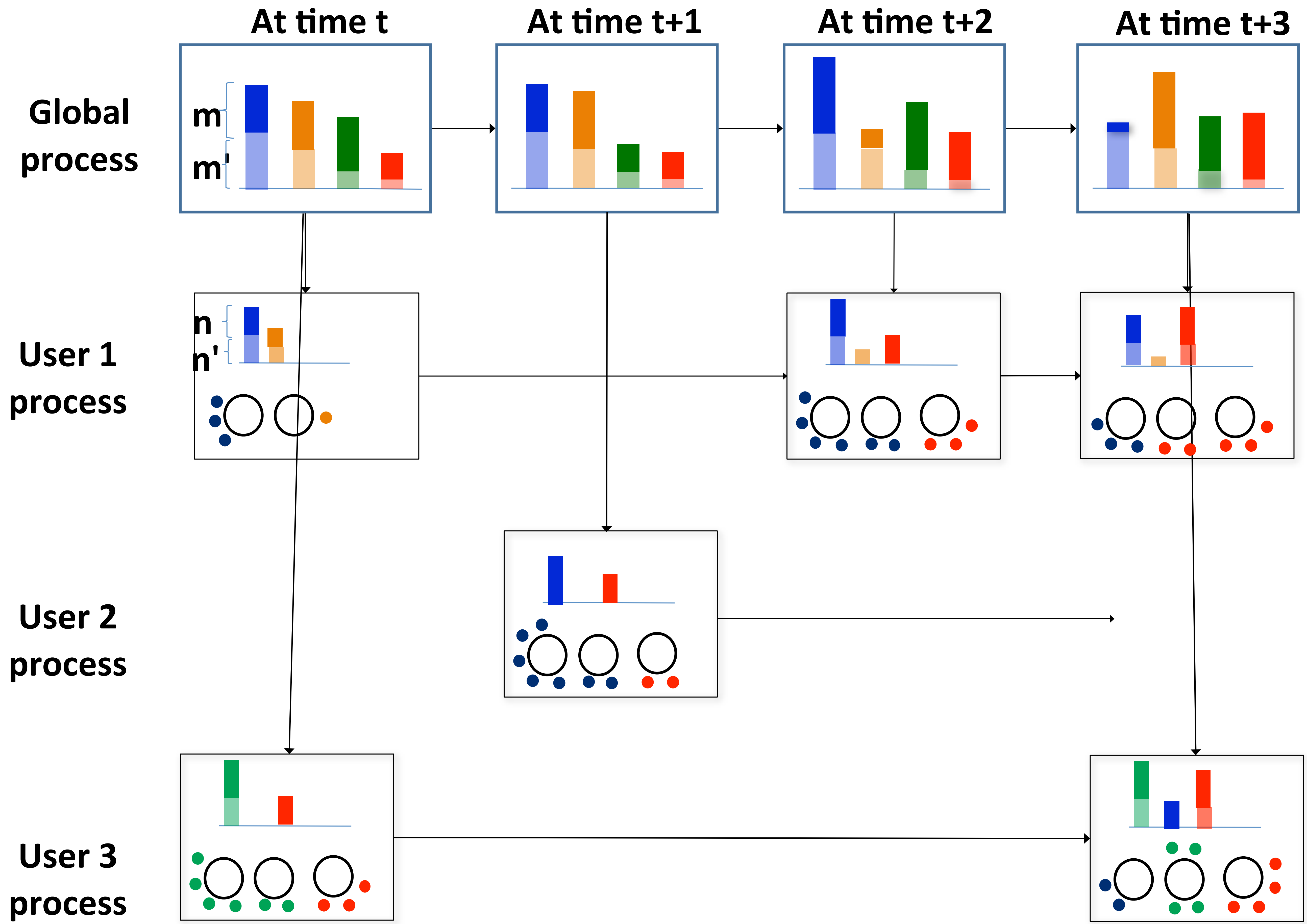

At time t
At time t+1
Recipe Chocolate Pizza Food Chicken Milk Butter Powder
Car Altima Accord Blue Book Kelley Prices Small Speed
job Career Business Assistant Hiring Part-time Receptionist
Bank Online Credit Card debt portfolio Finance Chase
priors
Food Chicken
Pizza mileage
Car speed offer
Camry accord career



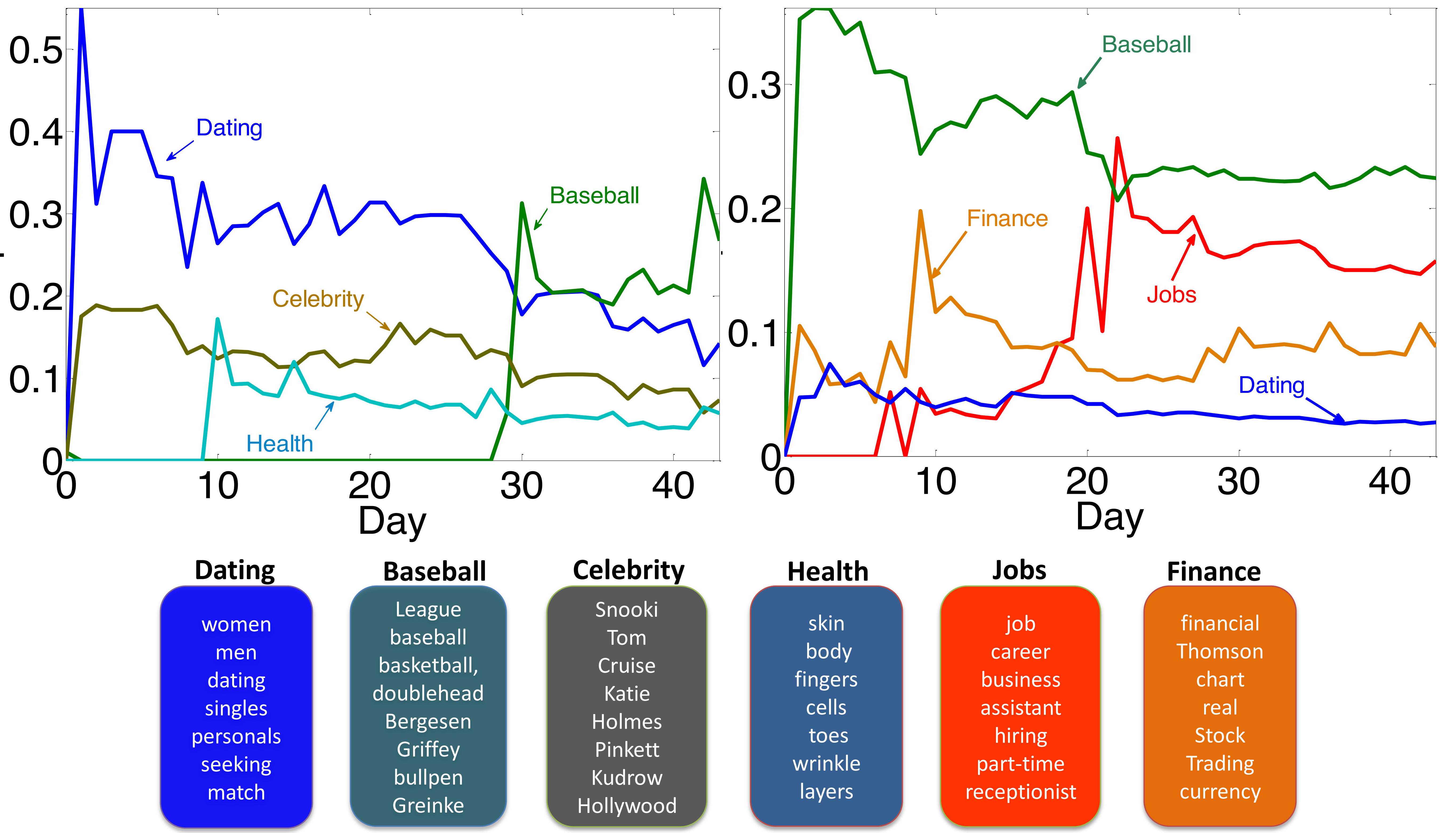


Generative Process

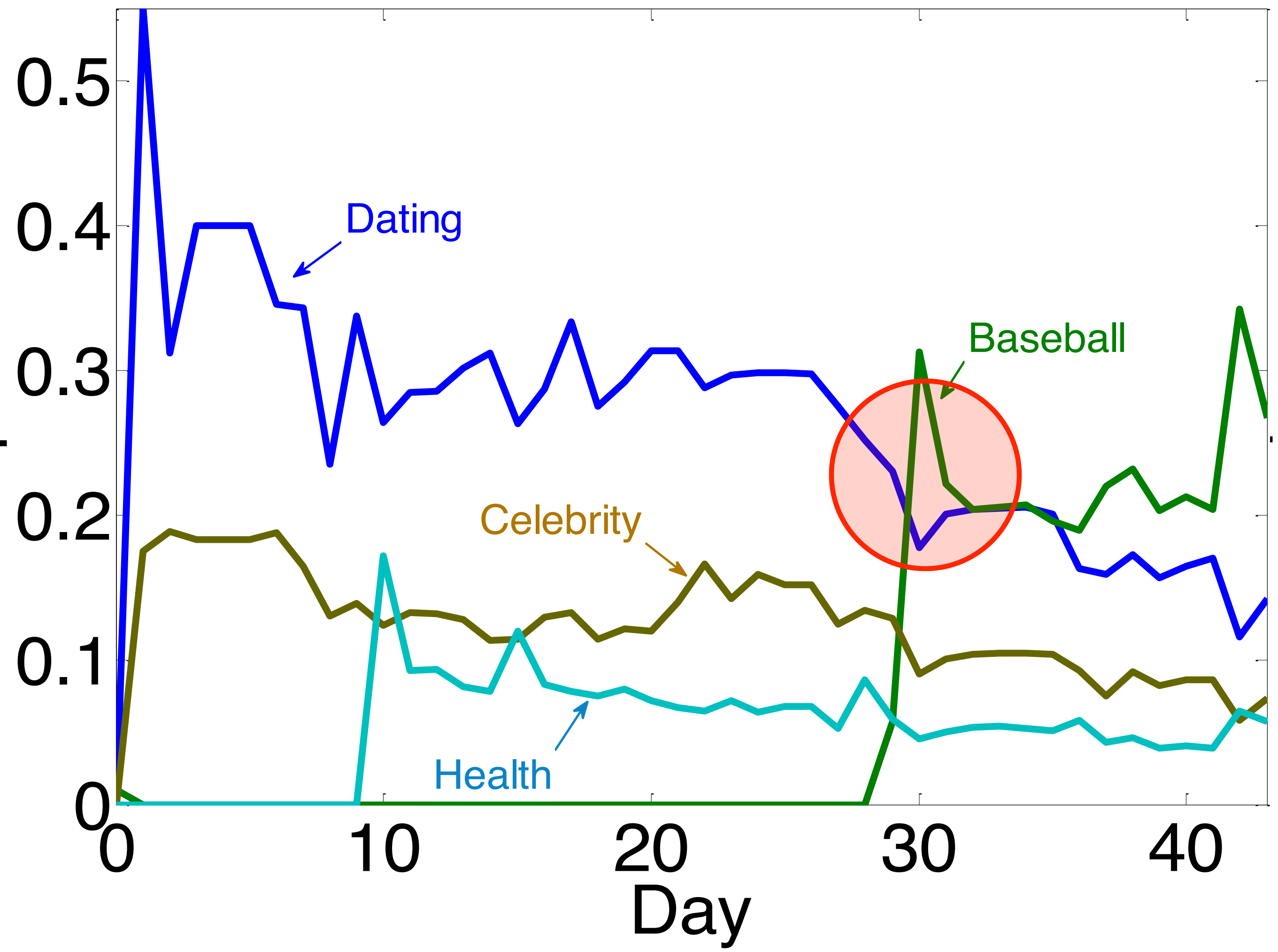
- For each user interaction
  - Choose an intent from local distribution
    - Sample word from the topic's word-distribution
  - Choose a new intent $\propto \alpha$
    - Sample a new intent from the global distribution
      - Sample word from the new topic word-distribution

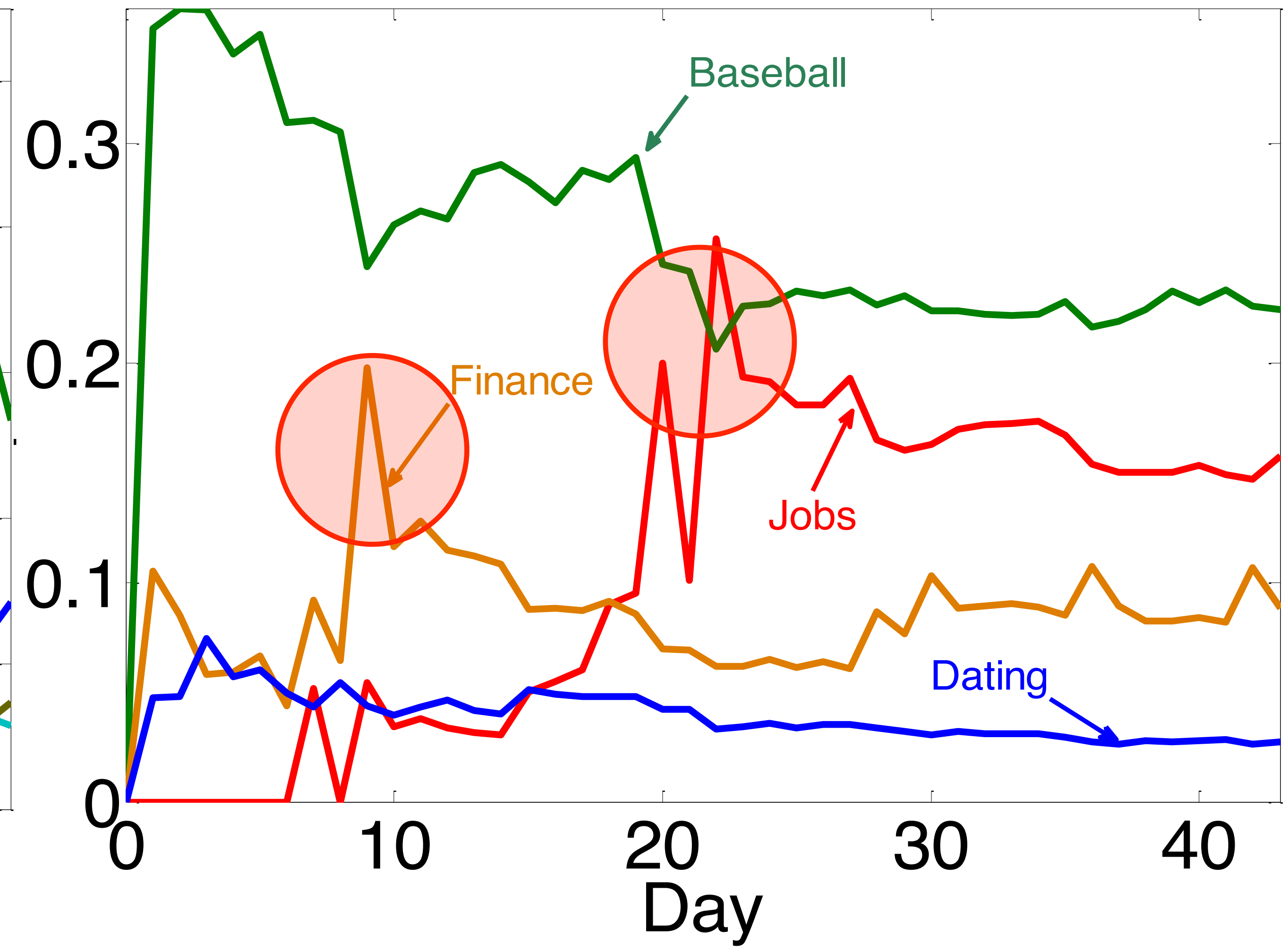

At time t
At time t+1
At time t+2
At time t+3
Global
process
m
m'
User 1
process
n
n'
User 2
process
User 3
process

Sample users



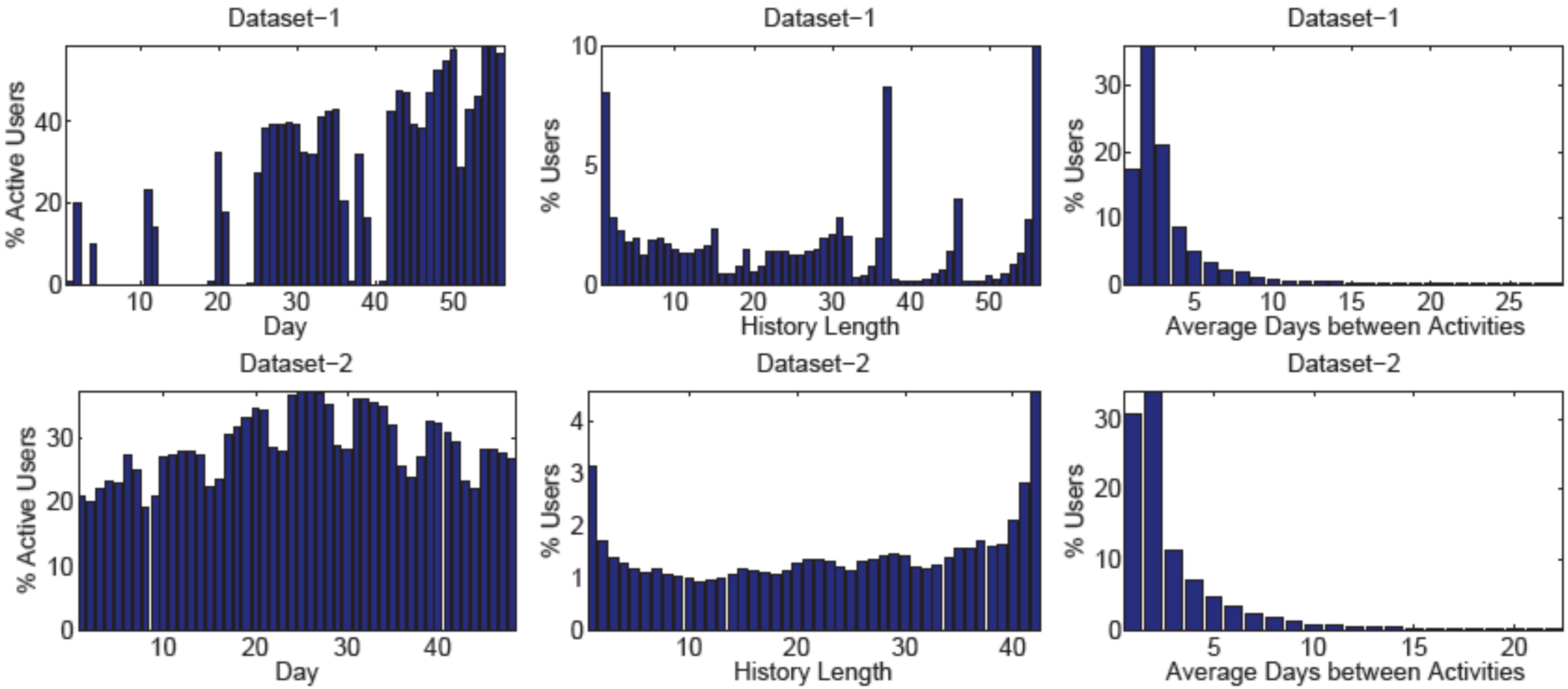
Sample users





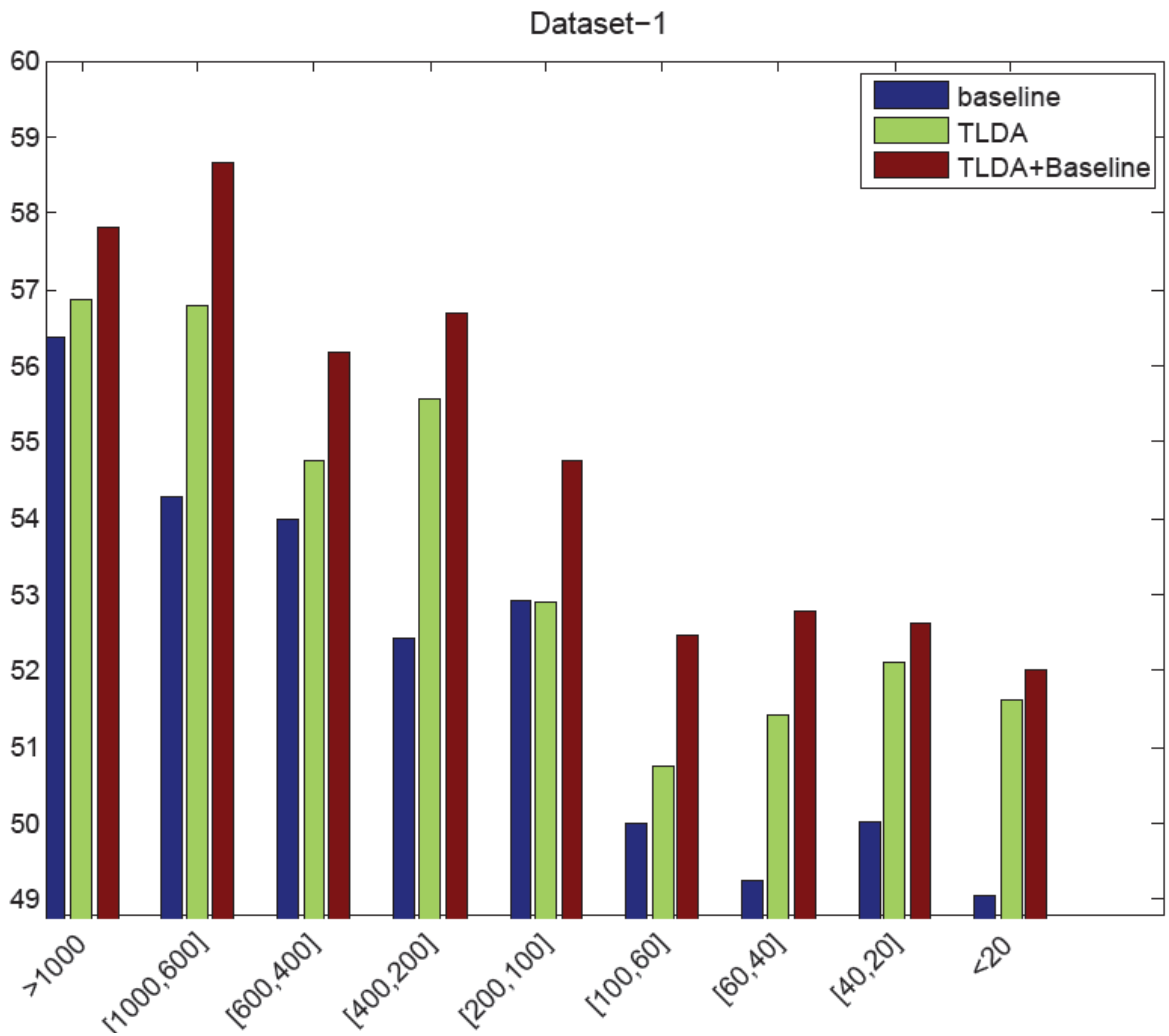
Dating	Baseball	Celebrity	Health	Jobs	Finance
women	League	Snooki	skin	job	financial
men	baseball	Tom	body	career	Thomson
dating	basketball,	Cruise	fingers	business	chart
singles	doublehead	Katie	cells	assistant	real
personals	Bergesen	Holmes	toes	hiring	Stock
seeking	Griffey	Pinkett	wrinkle	part-time	Trading
match	bullpen	Kudrow	layers	receptionist	currency
	Greinke	Hollywood			

Data

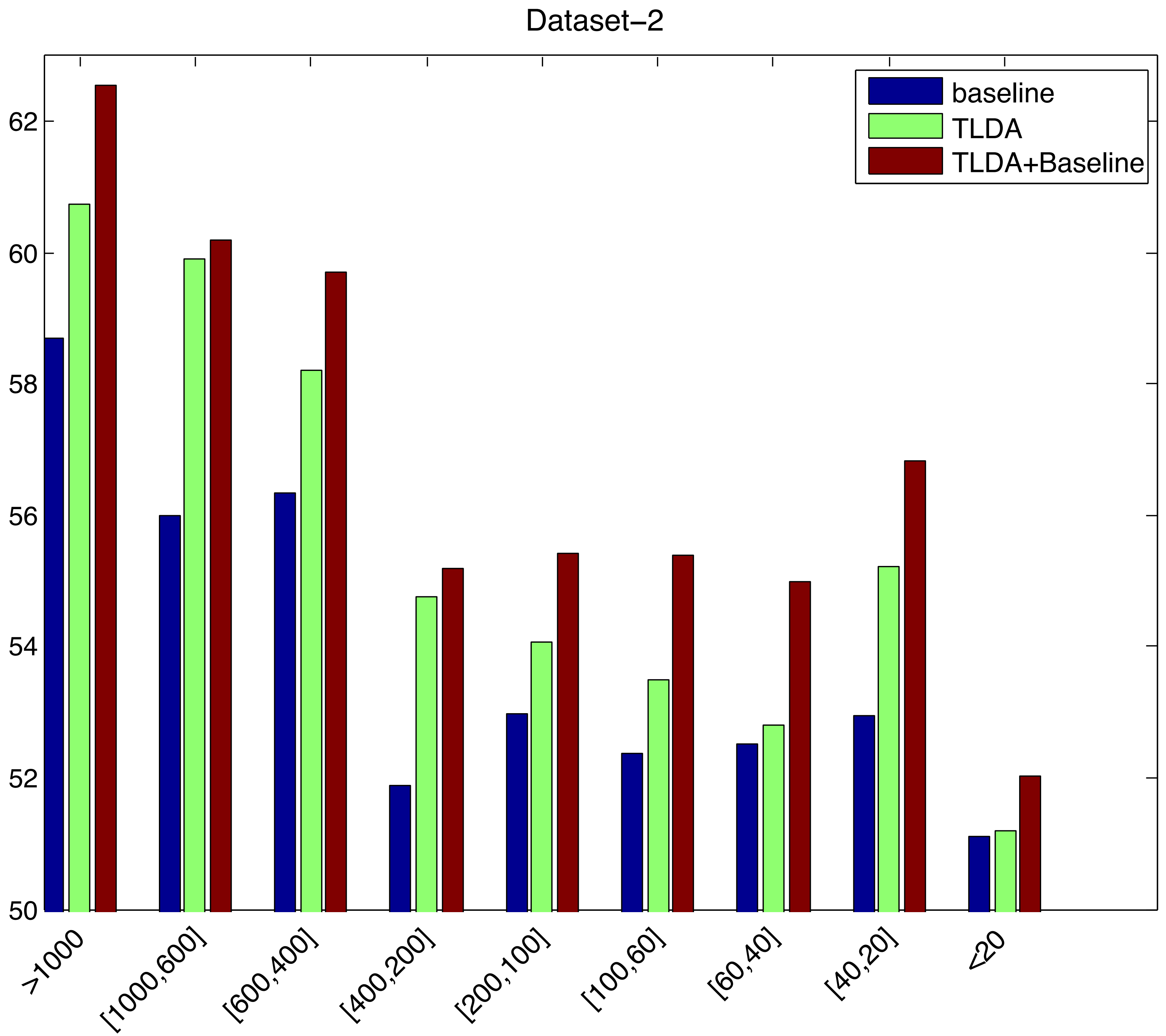


dataset	# days	# users	# campaigns	size
1	56	13.34M	241	242GB
2	44	33.5M	216	435GB

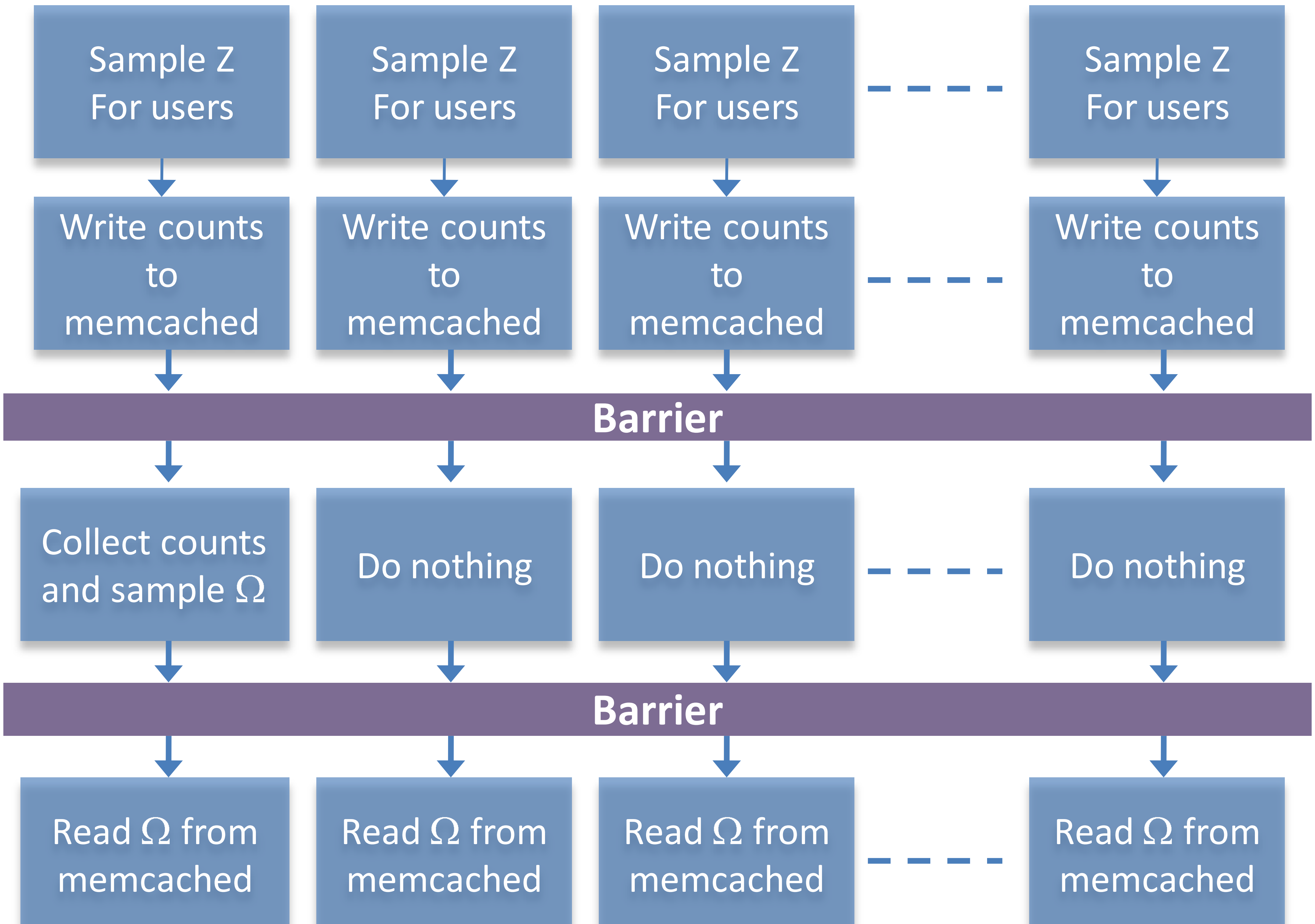
ROC score improvement



ROC score improvement



LDA for user profiling



News

News Stream

News Stream

Add-ons turn tax cut bill into 'Christmas t…'

AP – 1 hr 32 mins ago

WASHINGTON – In the …

AP

Republicans is bec… and lawmakers. Bu… Bill Clinton even ba…

Full Story »

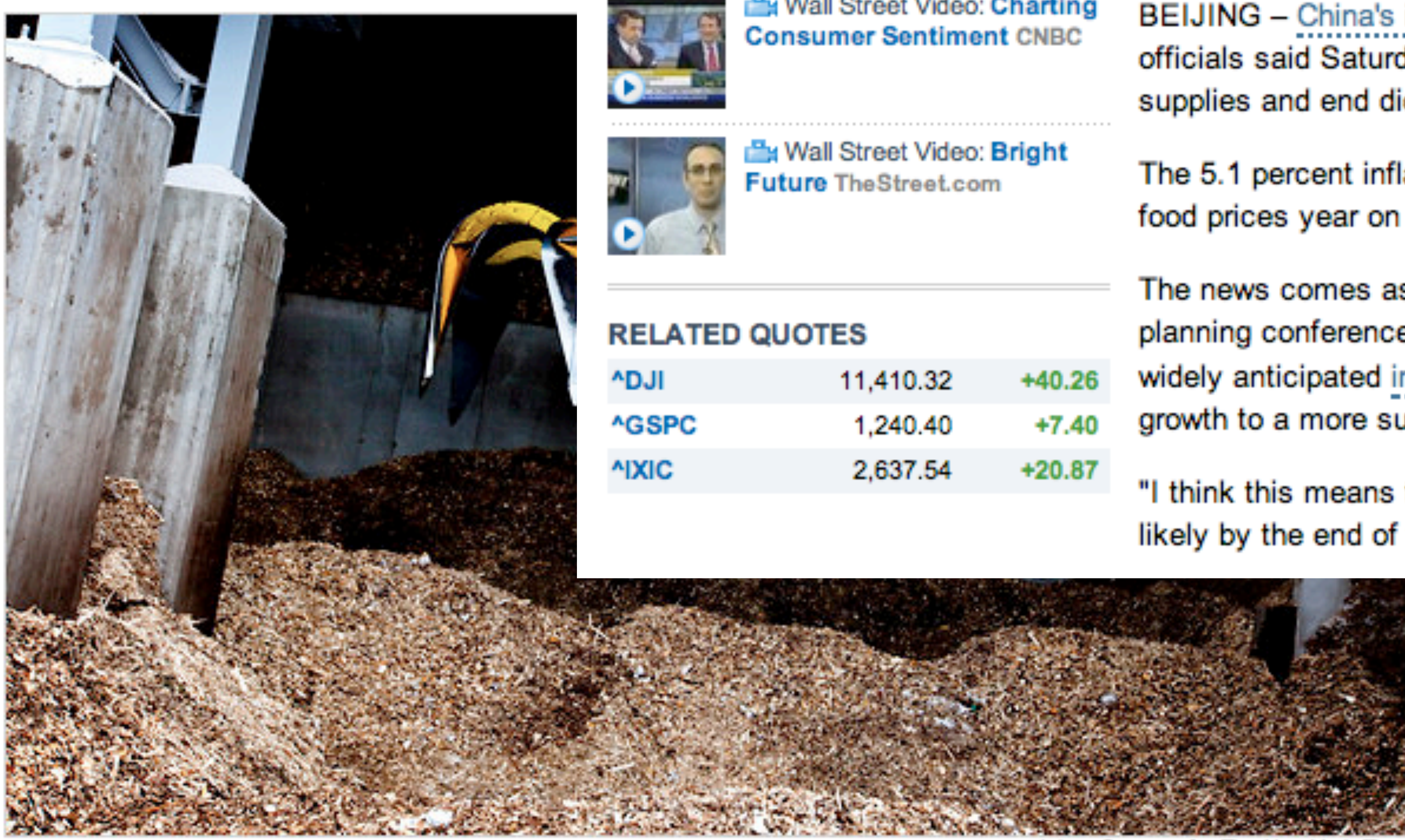
Video: Gibbs: I Ha…

Slideshow: Preside…

Related: Tax fight …

BEYOND FOSSIL FUELS

Using Waste, Swedish …



Johan Spanner for The New York Times

As part of its citywide system, Kristianstad burns wood waste like tree prunings and scraps from flooring factories to power an underground district heating grid.

China says inflation up 5.1 perce… …ber

Associated Press

Buzz up! 19 votes

Share

Print

Wall Street Video: Charting Consumer Sentiment CNBC

Wall Street Video: Bright Future TheStreet.com

RELATED QUOTES

^DJI	11,410.32	+40.26
^GSPC	1,240.40	+7.40
^IXIC	2,637.54	+20.87

By CARA ANNA, Associated Press

BEIJING – China's inflation su… …ovember, officials said Saturday, despit… …ase food supplies and end diesel short…

The 5.1 percent inflation rate was driven by a 11.7 percent jump in food prices year on year.

The news comes as China's leaders meet for the top economic planning conference of the year and as financial markets watch for a widely anticipated interest rate hike to help bring rapid economic growth to a more sustainable level.

"I think this means that an interest rate hike of 25 basis points is very likely by the end of the year," said CLSA analyst Andy Rothman.

Suit to Recover Madoff's Money Calls Austrian an Accomplice

By DIANA B. HENRIQUES and PETER LATTMAN

Sonja Kohn, an Austrian banker, is accused of masterminding a 23-year conspiracy that played a central role in financing the gigantic Ponzi scheme.

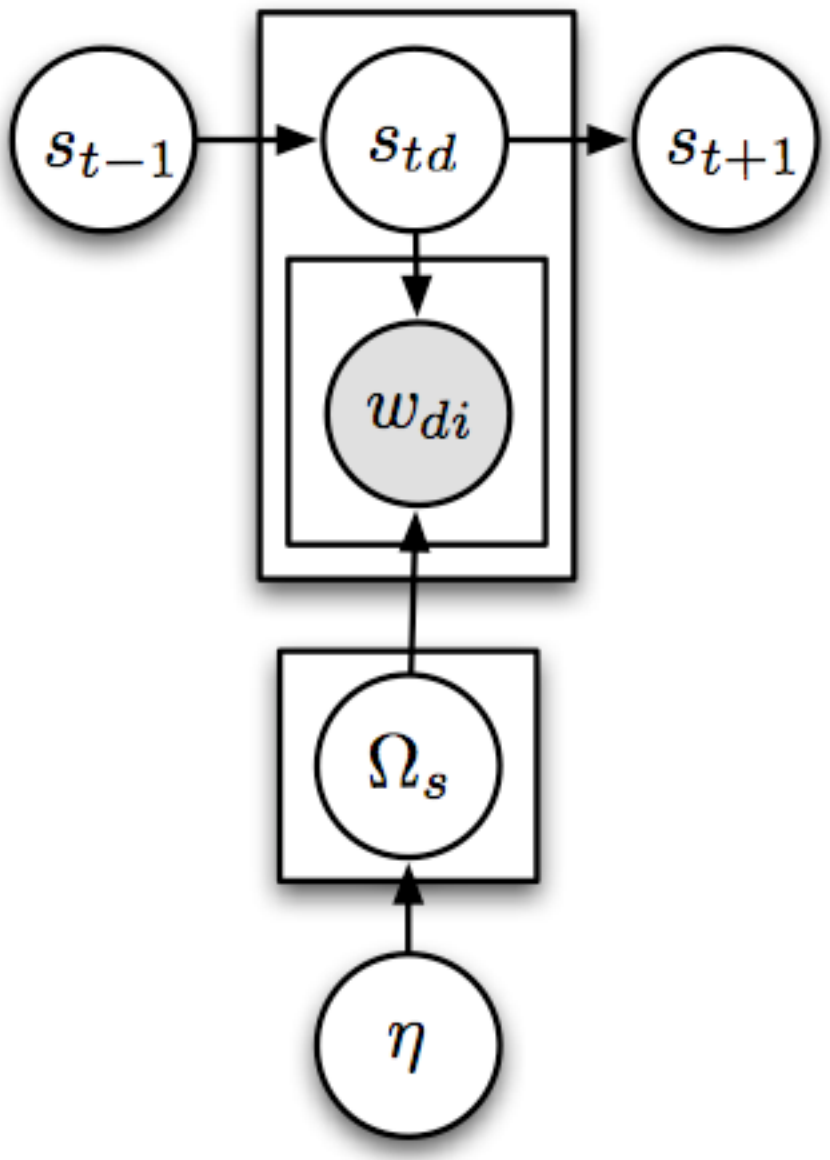
Post a Comment

News Stream

- Over 1 high quality news article per second
- Multiple sources (Reuters, AP, CNN, ...)
- Same story from multiple sources
- Stories are related

- Goals
  - Aggregate articles into a storyline
  - Analyze the storyline (topics, entities)

Clustering / RCRP



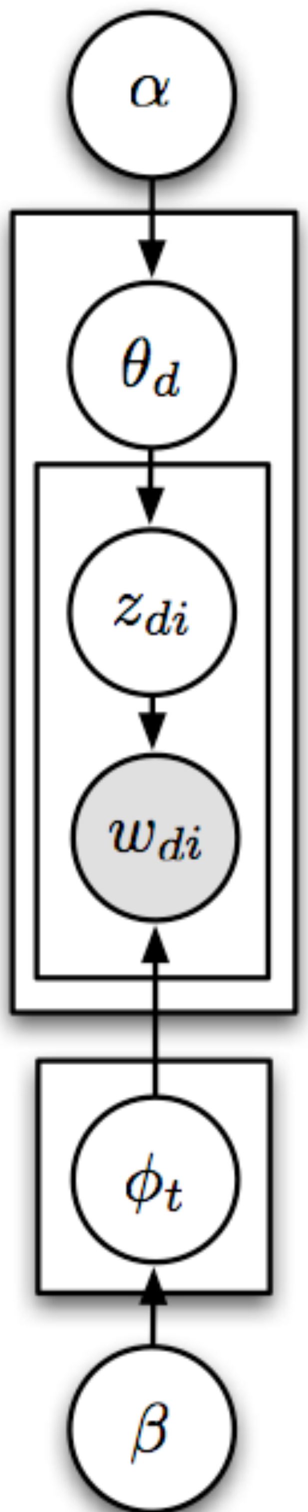
- Assume active story distribution at time t
- Draw story indicator
- Draw words from story distribution
- Down-weight story counts for next day

Ahmed & Xing, 2008

Clustering / RCRP

- Pro
  - Nonparametric model of story generation (no need to model frequency of stories)
  - No fixed number of stories
  - Efficient inference via collapsed sampler
- Con
  - We learn nothing!
  - No content analysis

Latent Dirichlet Allocation



- Generate topic distribution per article
- Draw topics per word from topic distribution
- Draw words from topic specific word distribution

Blei, Ng, Jordan, 2003

Latent Dirichlet Allocation

- Pro
  - Topical analysis of stories
  - Topical analysis of words (meaning, saliency)
  - More documents improve estimates
- Con
  - No clustering

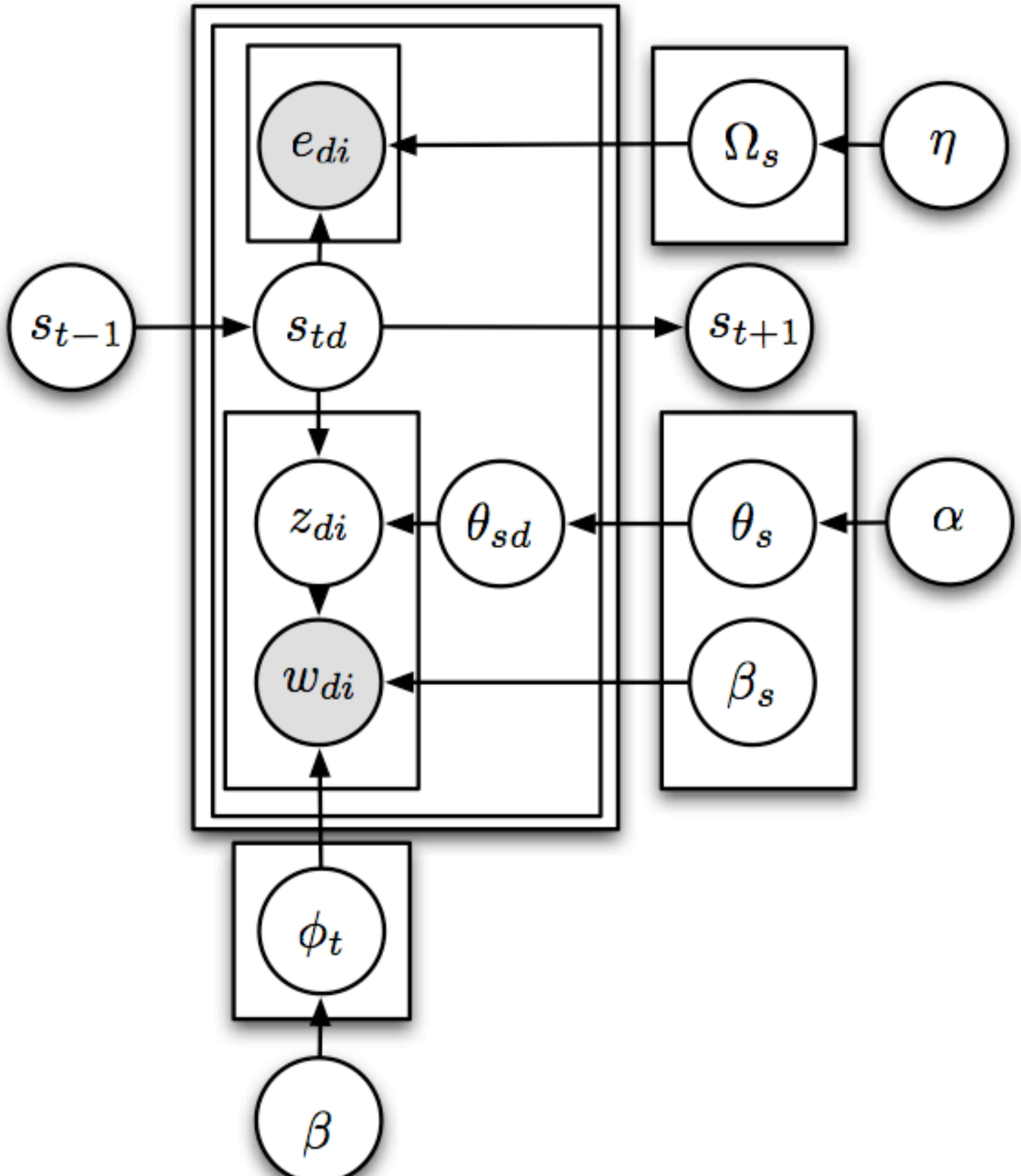
More Issues

More Issues

- Named entities are special, topics less
  (e.g. Tiger Woods and his mistresses)
- Some stories are strange
  (topical mixture is not enough - dirty models)
- Articles deviate from general story
  (Hierarchical DP)

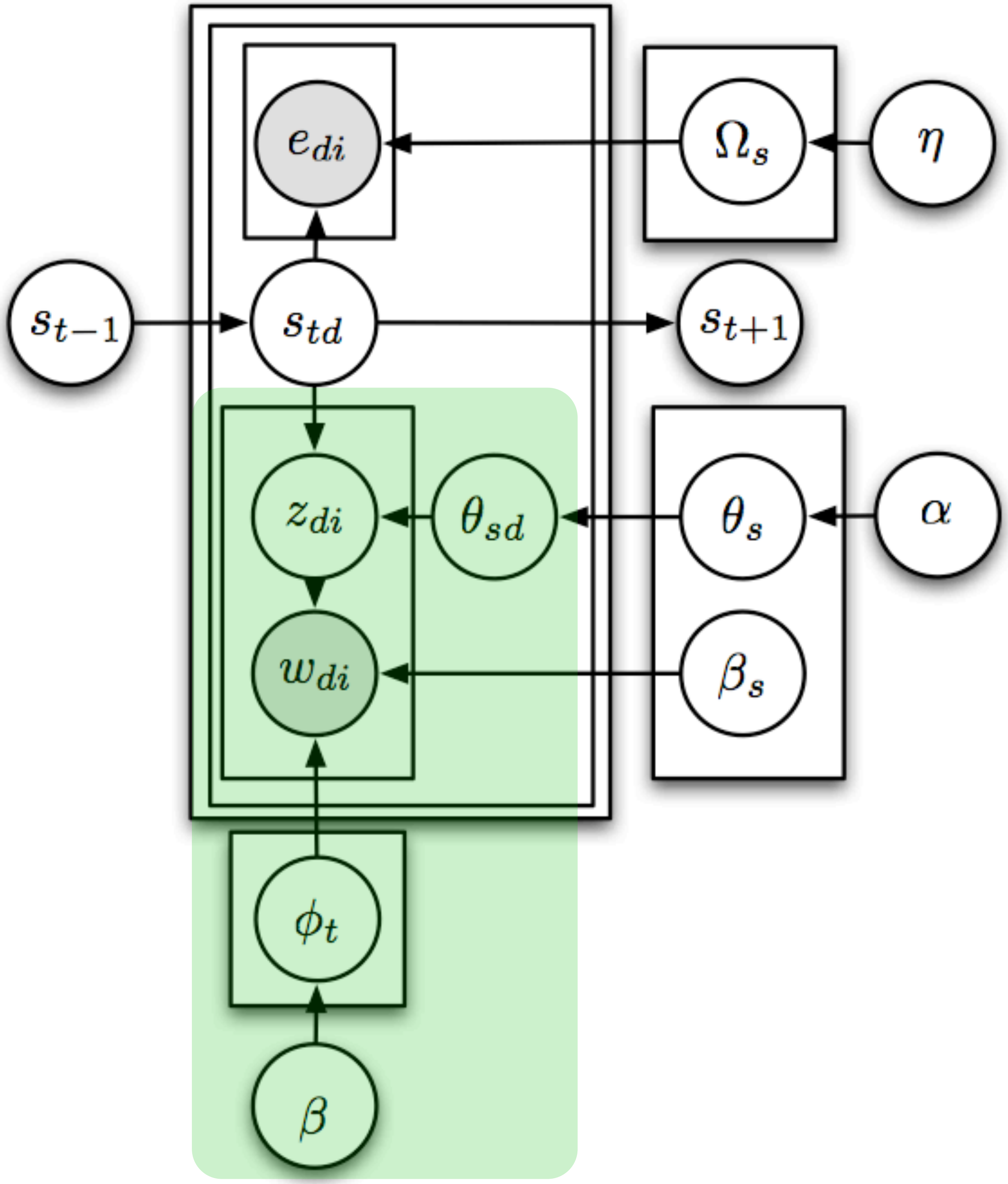
Storylines

Storylines Model



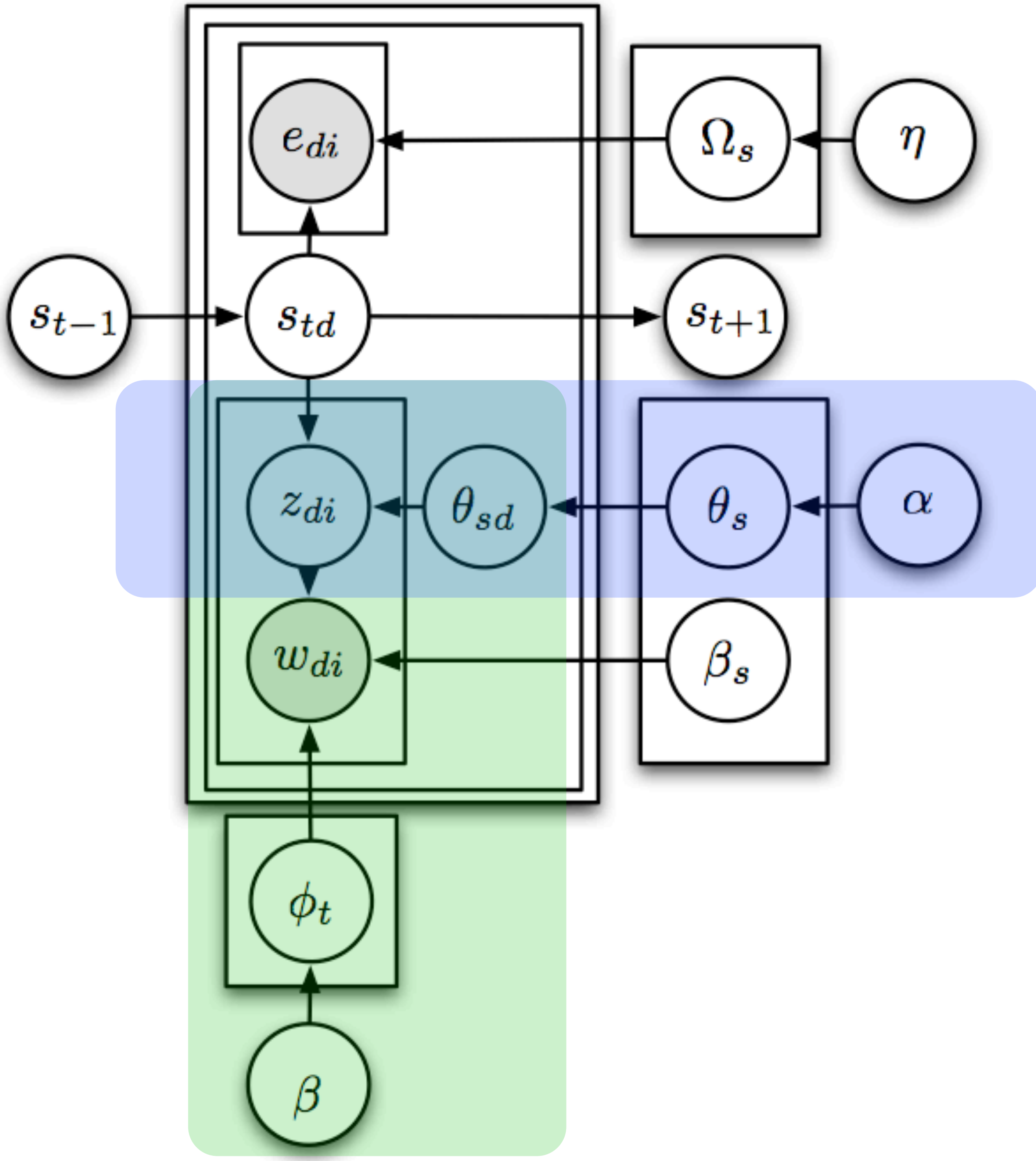
- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

Storylines Model



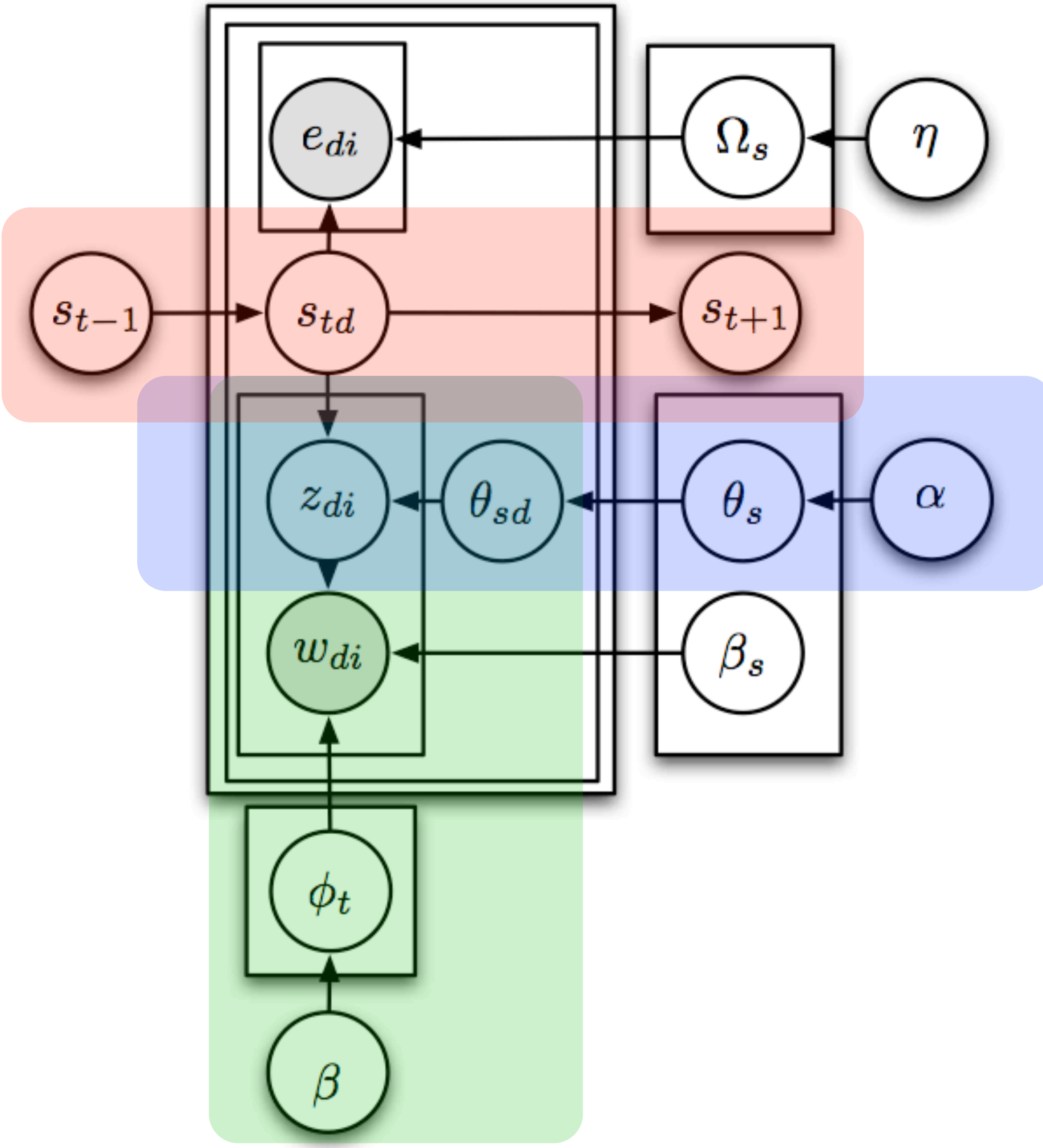
- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

Storylines Model



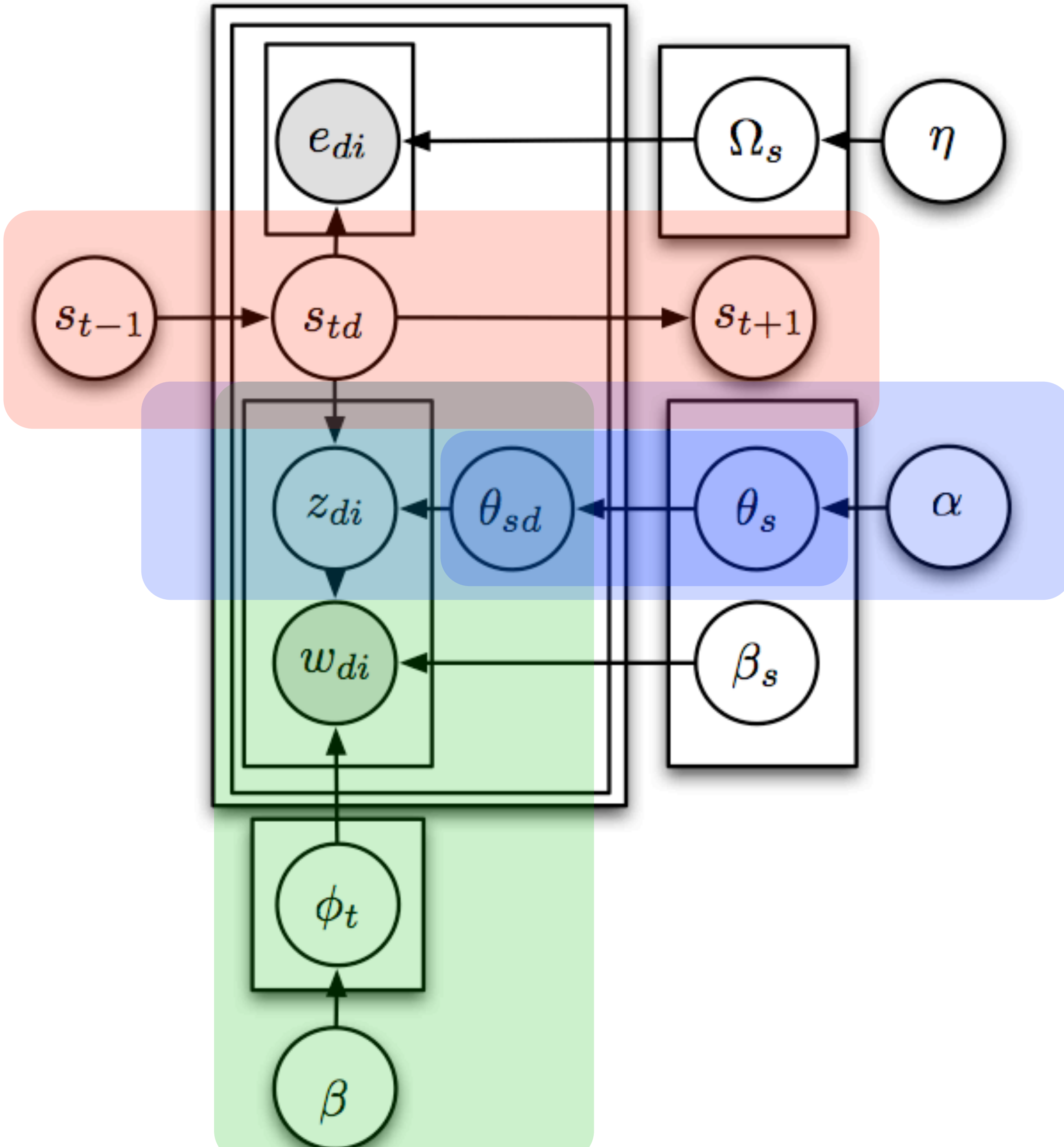
- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

Storylines Model



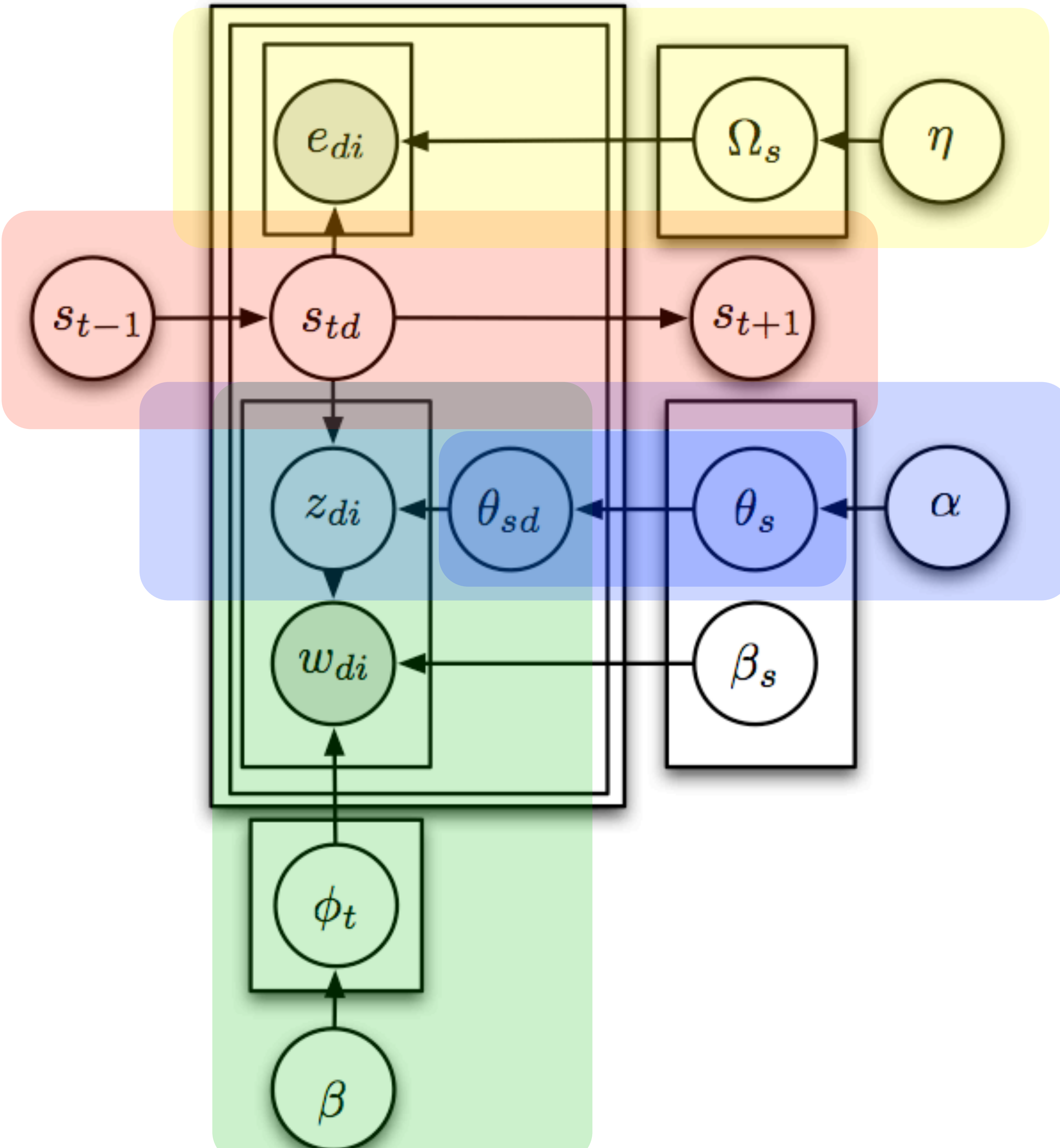
- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

Storylines Model



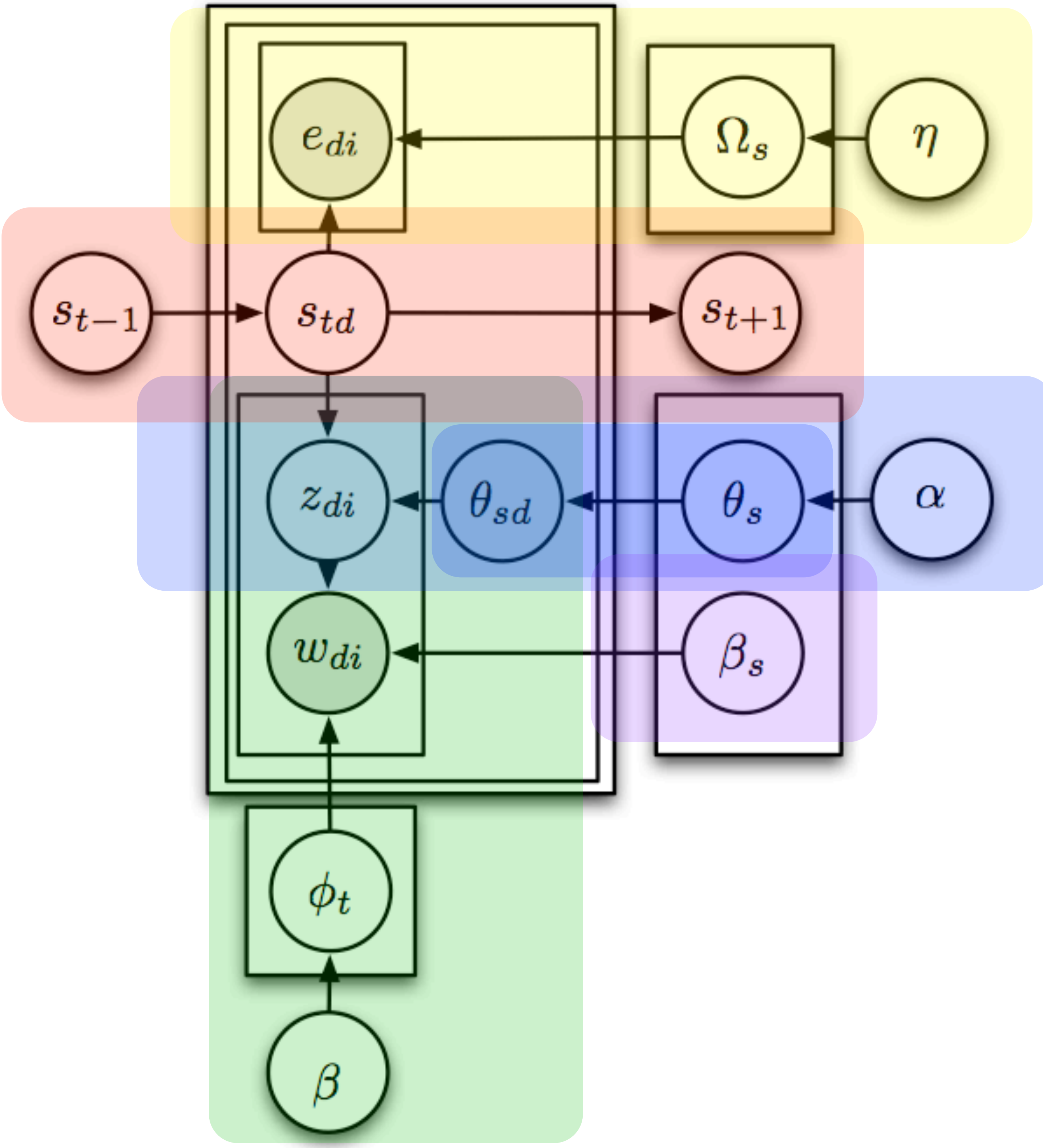
- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

Storylines Model



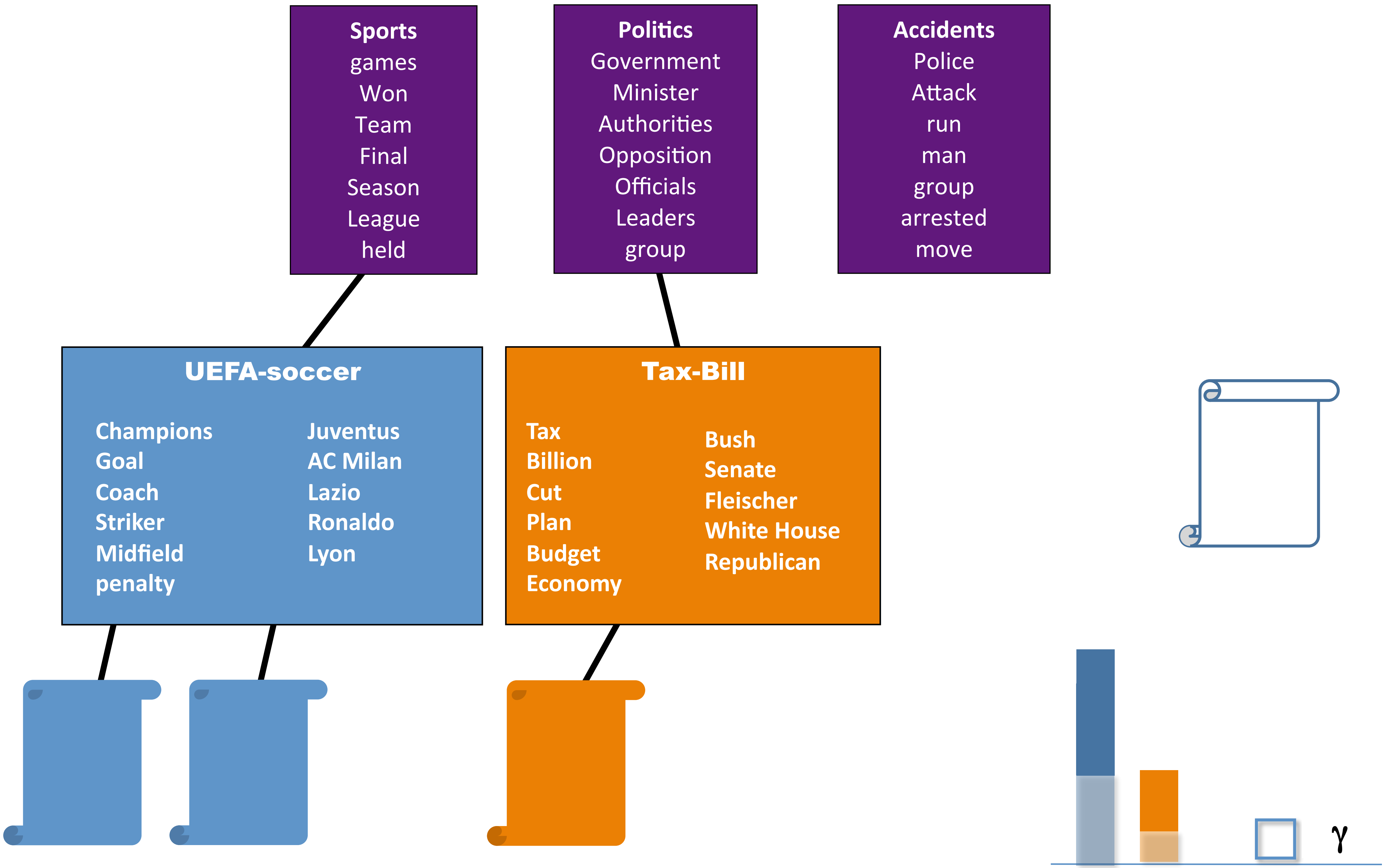
- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

Storylines Model

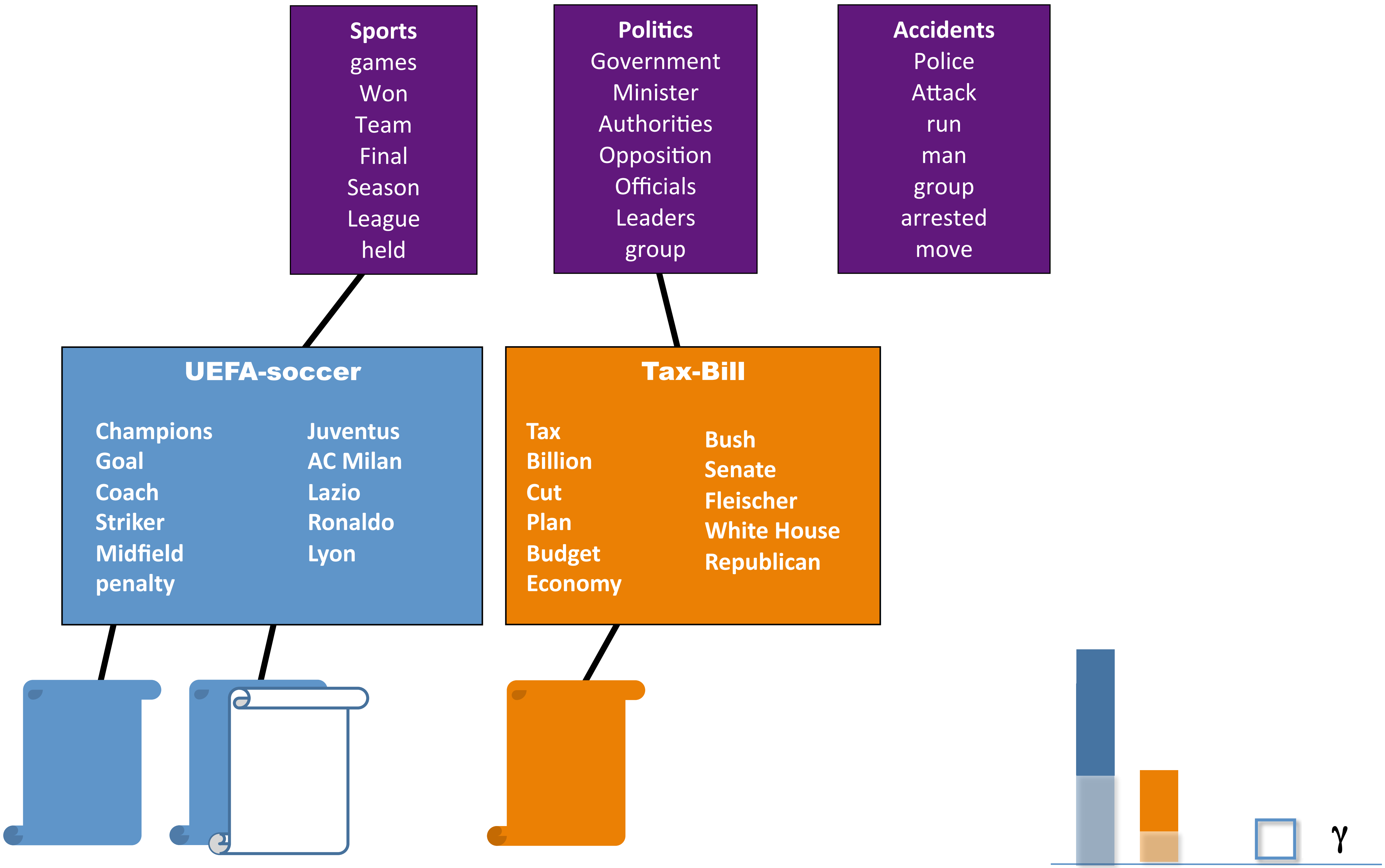


- Topic model
- Topics per cluster
- RCRP for cluster
- Hierarchical DP for article
- Separate model for named entities
- Story specific correction

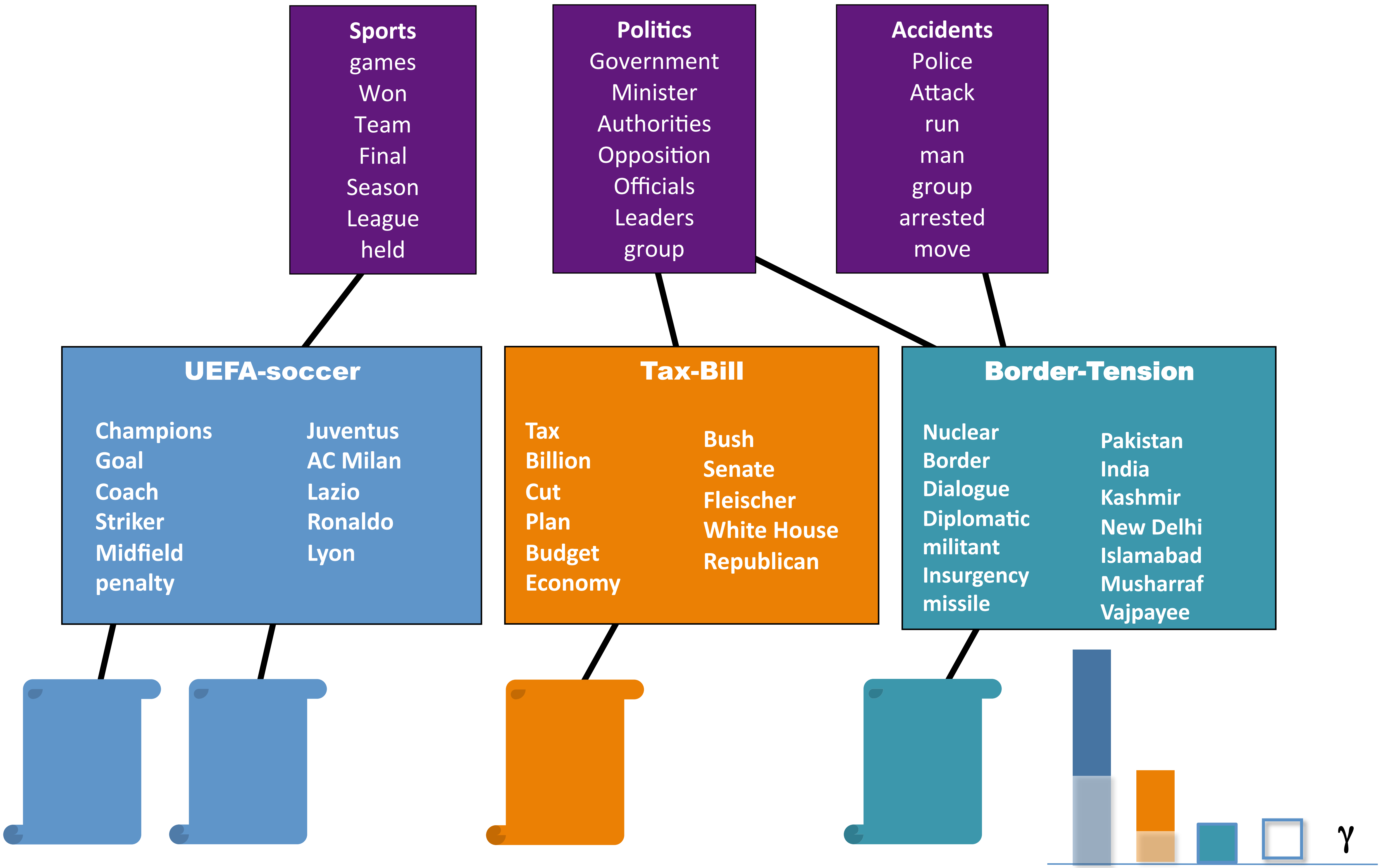
Dynamic Cluster-Topic Hybrid



Dynamic Cluster-Topic Hybrid



Dynamic Cluster-Topic Hybrid



Inference

- We receive articles as a stream
  Want topics & stories now
- Variational inference infeasible
  (RCRP, sparse to dense, vocabulary size)
- We have a ‘tracking problem’
  - Sequential Monte Carlo
  - Use sampled variables of surviving particle
  - Use ideas from Cannini et al. 2009

Particle Filter

- Proposal distribution - draw stories s, topics z

$$p(s_{t+1}, z_{t+1} | x_{1 \ldots t+1}, s_{1 \ldots t}, z_{1 \ldots t})$$

  using Gibbs Sampling for each particle
- Reweight particle via

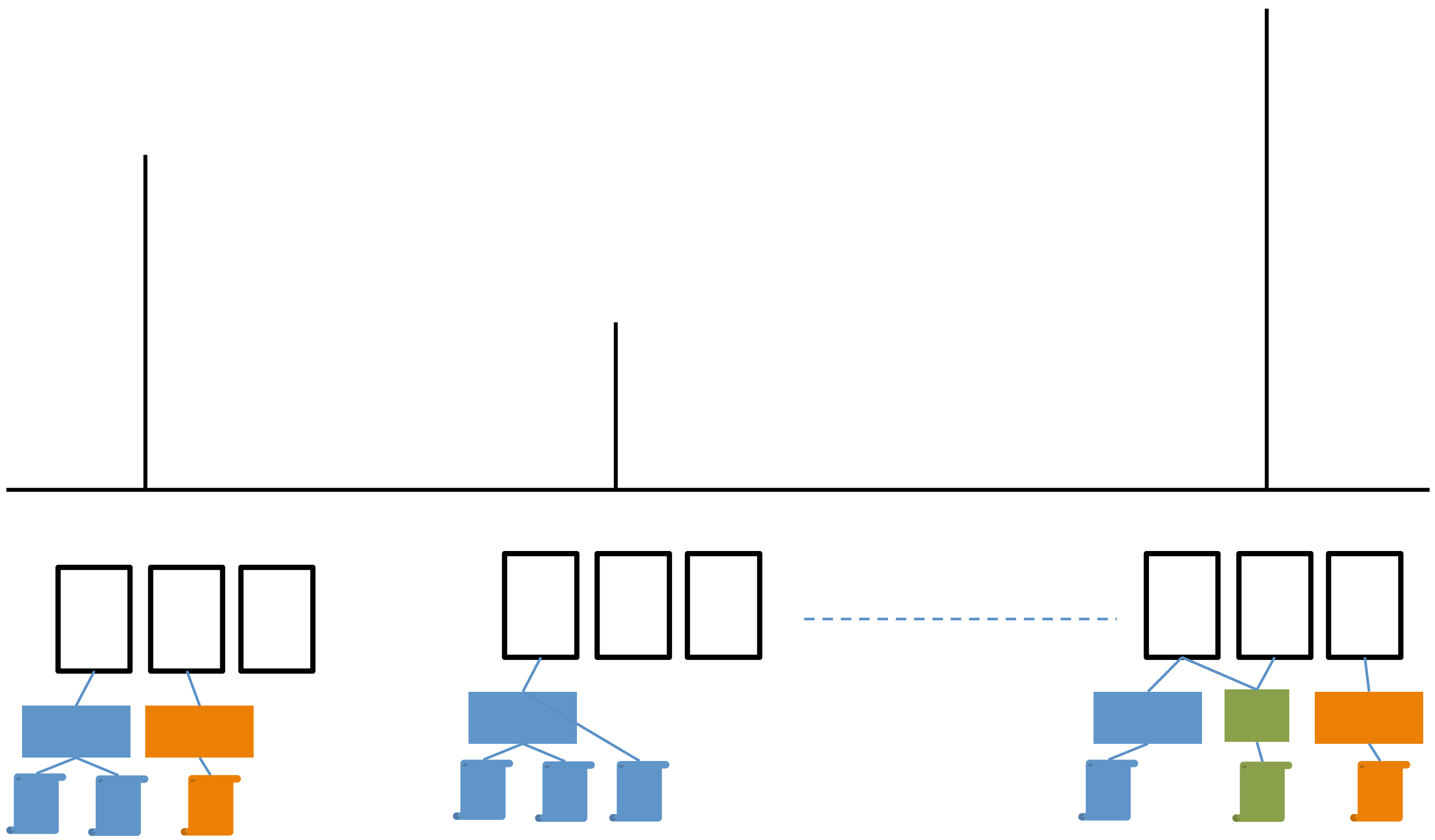
past state

$$p(x_{t+1} | x_{1 \ldots t}, s_{1 \ldots t}, z_{1 \ldots t})$$

new data

- Resample particles if l_2 norm too large (resample some assignments for diversity, too)
- Compare to multiplicative updates algorithm In our case predictive likelihood yields weights

Particle Filter

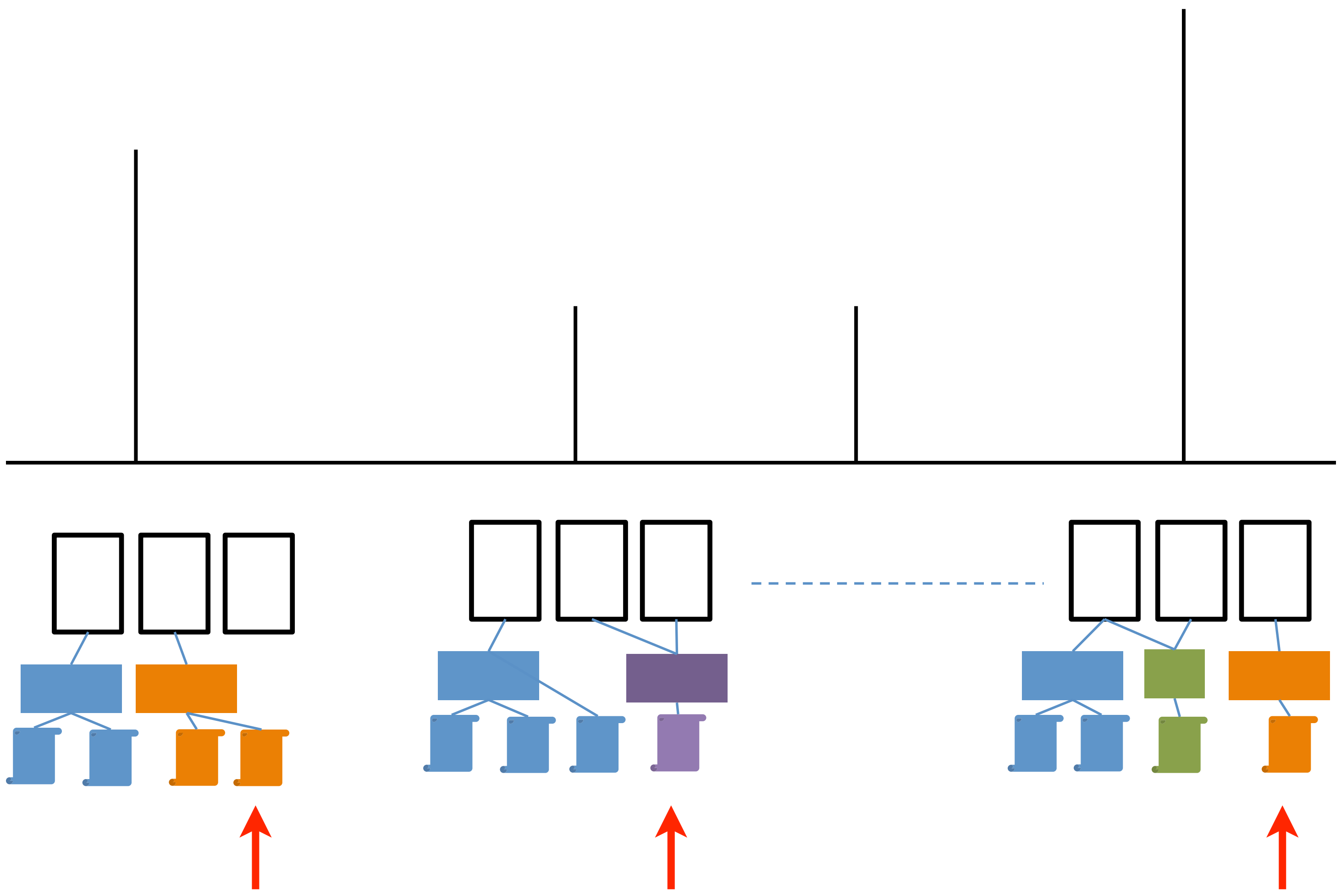


Algorithm 1 A Particle Filter Algorithm

Initialize ω_1^f to $\frac{1}{F}$ for all $f \in \{1, \dots F\}$
for each document d with time stamp t **do**
  for $f \in \{1, \dots F\}$ **do**
    Sample $s_{td}^f, \mathbf{z}_{td}^f$ using MCMC
    $\omega^f \leftarrow \omega^f P(\mathbf{x}_{td} | \mathbf{z}_{td}^f, \mathbf{s}_{td}^f, \mathbf{x}_{1:t,d-1})$
  end for
  Normalize particle weights
  if $\|\omega_t\|_2^{-2} <$ threshold **then**
    resample particles
    for $f \in \{1, \dots F\}$ **do**
      MCMC pass over 10 random past documents
    end for
  end if
end for

- **s** and **z** are tightly coupled
- Alternative to MCMC
  - Sample **s** then sample **z** (high variance)
  - Sample **z** then sample **s** (doesn't make sense)
- Idea (following a similar trick by Jain and Neal)
  - Run a few iterations of MCMC over s and z
  - Take last sample as the proposed value

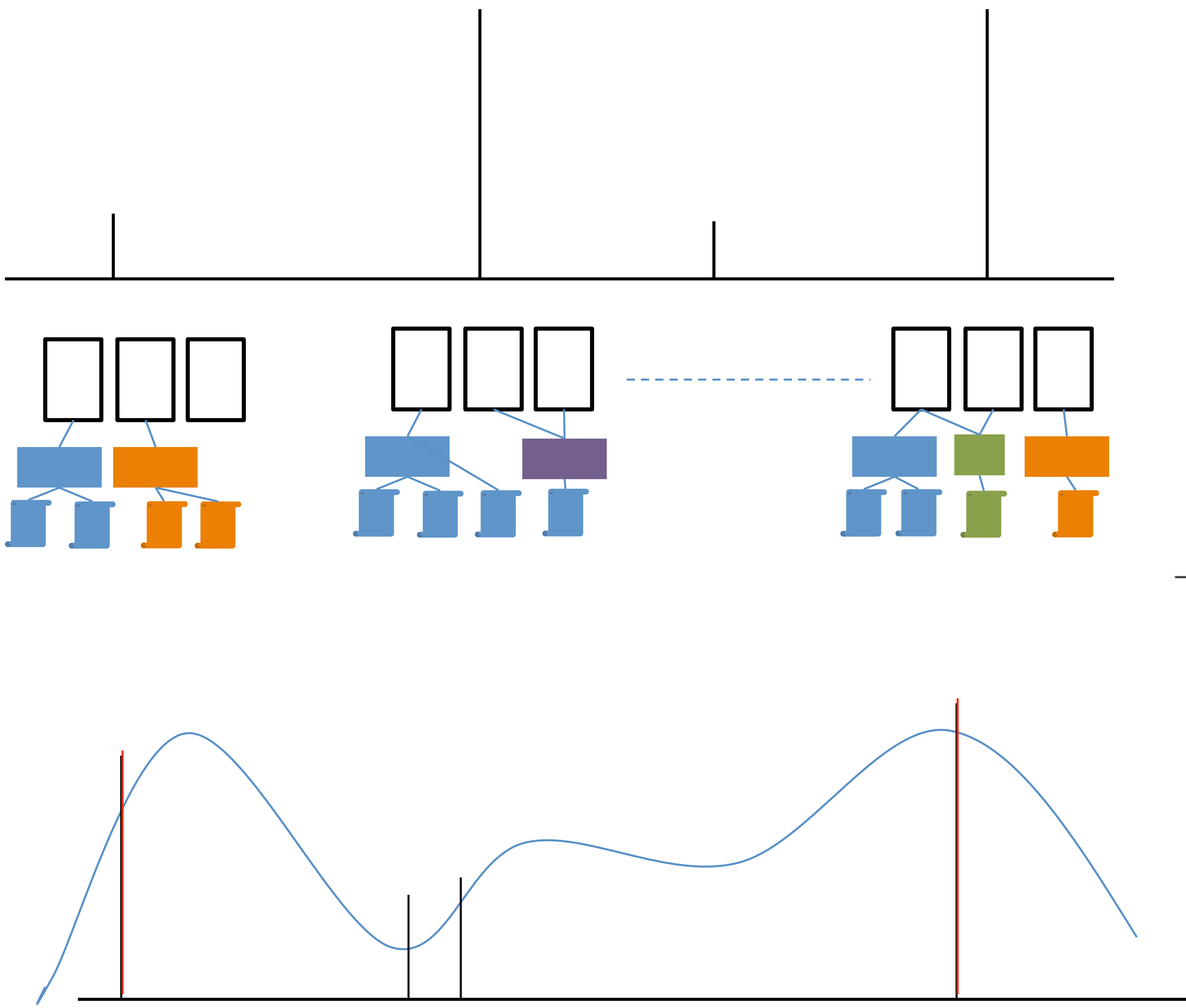
Particle Filter



Algorithm 1 A Particle Filter Algorithm

Initialize ω_1^f to $\frac{1}{F}$ for all $f \in \{1, \ldots F\}$
for each document d with time stamp t **do**
  for $f \in \{1, \ldots F\}$ **do**
    Sample $s_{td}^f, \mathbf{z}_{td}^f$ using MCMC
    $\omega^f \leftarrow \omega^f P(\mathbf{x}_{td} | \mathbf{z}_{td}^f, \mathbf{s}_{td}^f, \mathbf{x}_{1:t,d-1})$
  end for
  Normalize particle weights
  if $\|\omega_t\|_2^{-2} <$ threshold **then**
    resample particles
    for $f \in \{1, \ldots F\}$ **do**
      MCMC pass over 10 random past documents
    end for
  end if
end for

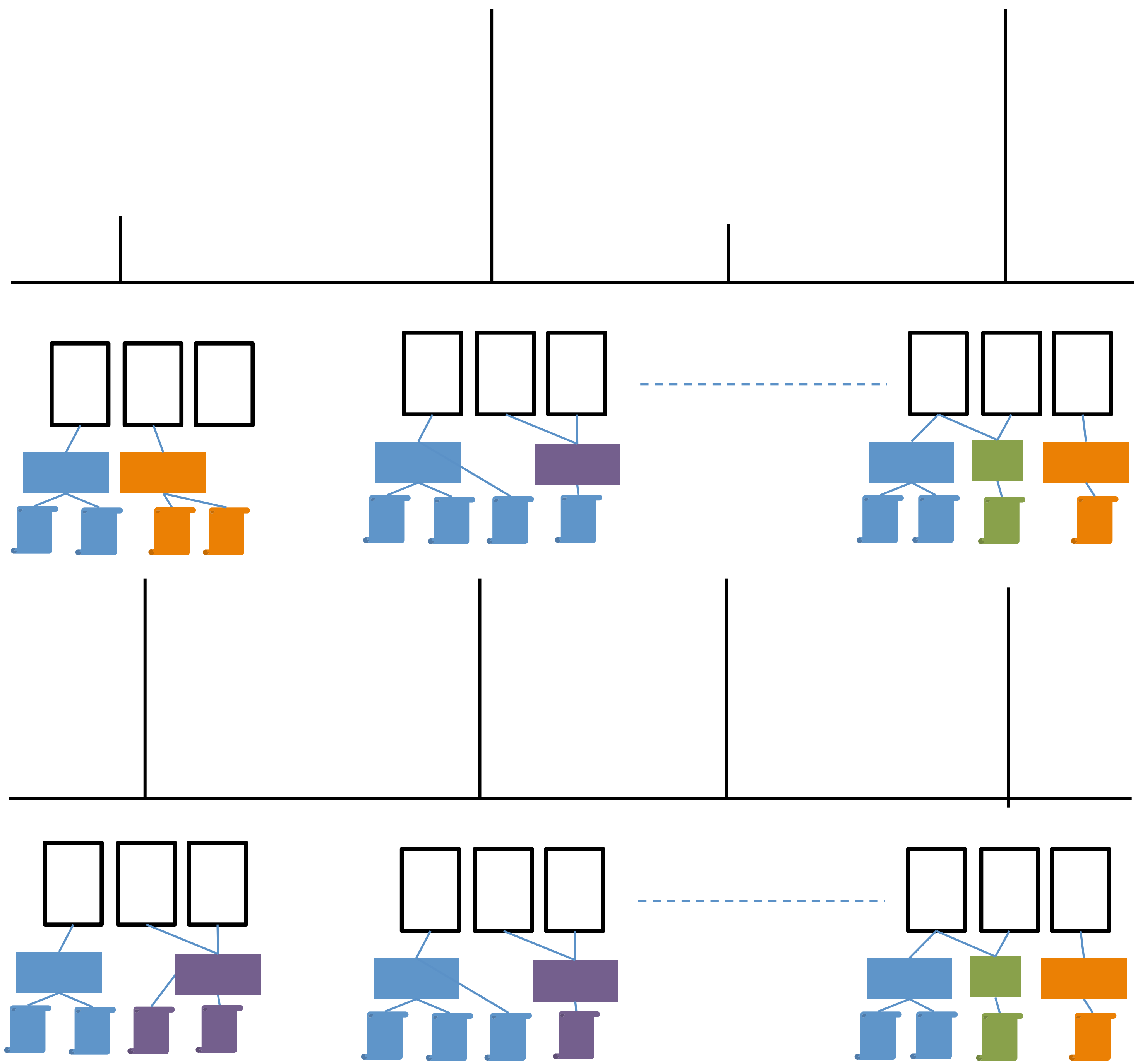
Particle Filter



Algorithm 1 A Particle Filter Algorithm

Initialize ω_1^f to $\frac{1}{F}$ for all $f \in \{1, \ldots F\}$
for each document d with time stamp t **do**
    for $f \in \{1, \ldots F\}$ **do**
        Sample $s_{td}^f, \mathbf{z}_{td}^f$ using MCMC
        $\omega^f \leftarrow \omega^f P(\mathbf{x}_{td} | \mathbf{z}_{td}^f, \mathbf{s}_{td}^f, \mathbf{x}_{1:t,d-1})$
    end for
    Normalize particle weights
    if $\|\omega_t\|_2^{-2} <$ threshold **then**
        resample particles
        for $f \in \{1, \ldots F\}$ **do**
            MCMC pass over 10 random past documents
        end for
    end if
end for

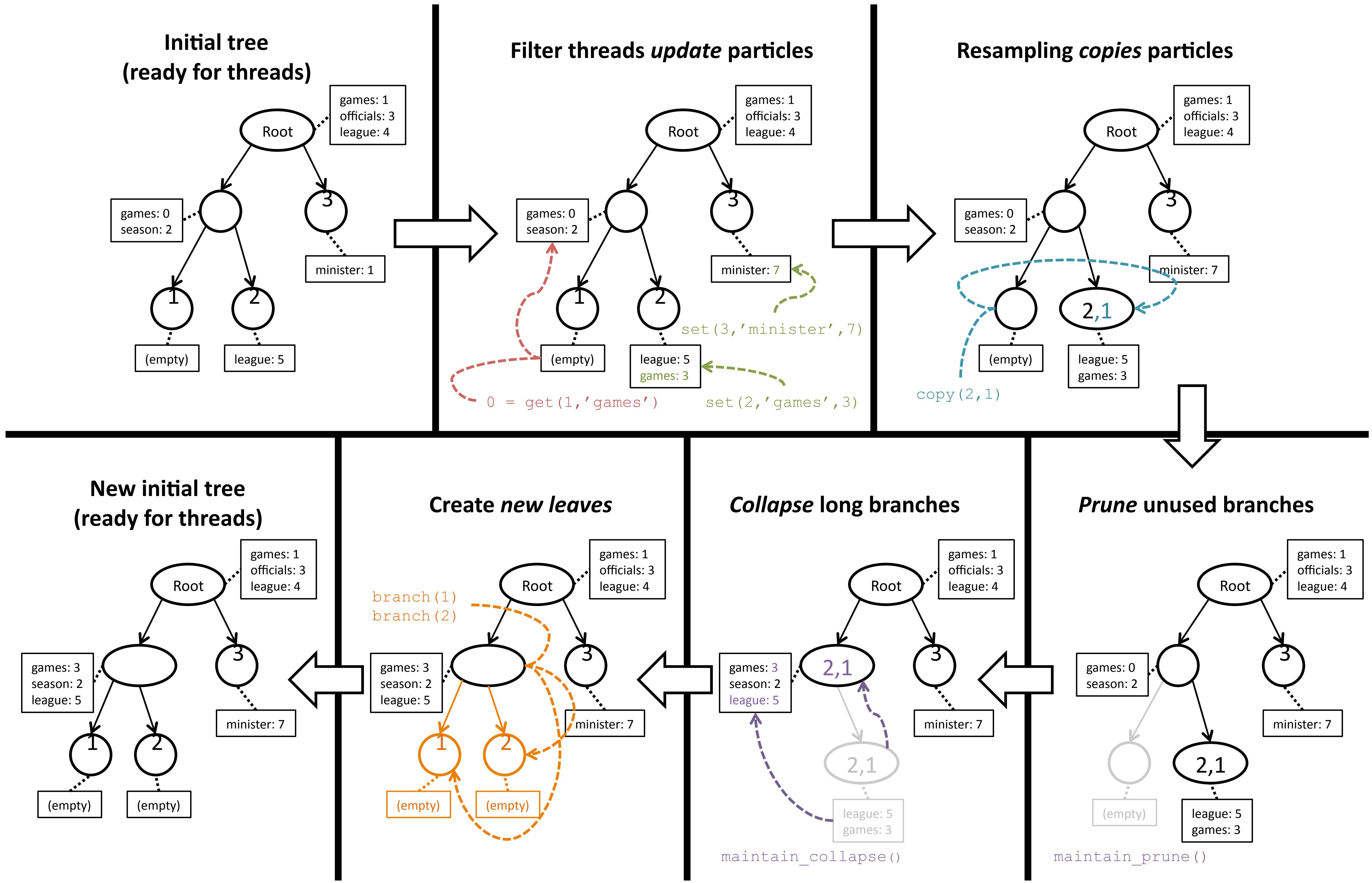
Particle Filter



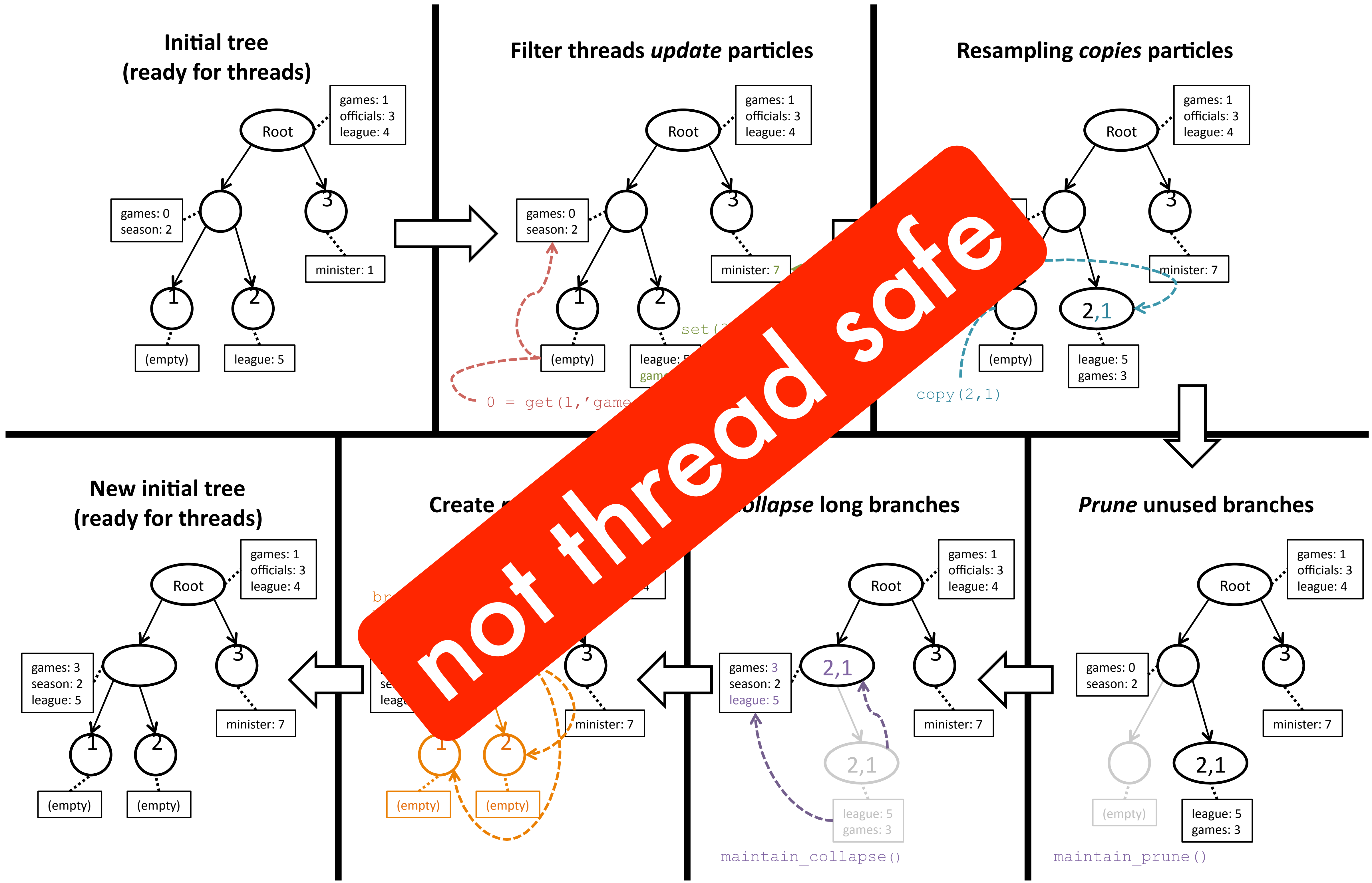
Algorithm 1 A Particle Filter Algorithm

Initialize ω_1^f to $\frac{1}{F}$ for all $f \in \{1, \ldots F\}$
for each document d with time stamp t **do**
  for $f \in \{1, \ldots F\}$ **do**
    Sample $s_{td}^f, \mathbf{z}_{td}^f$ using MCMC
    $\omega^f \leftarrow \omega^f P(\mathbf{x}_{td} | \mathbf{z}_{td}^f, \mathbf{s}_{td}^f, \mathbf{x}_{1:t,d-1})$
  end for
  Normalize particle weights
  if $\|\omega_t\|_2^{-2}$ < threshold **then**
    resample particles
    for $f \in \{1, \ldots F\}$ **do**
      MCMC pass over 10 random past documents
    end for
  end if
end for

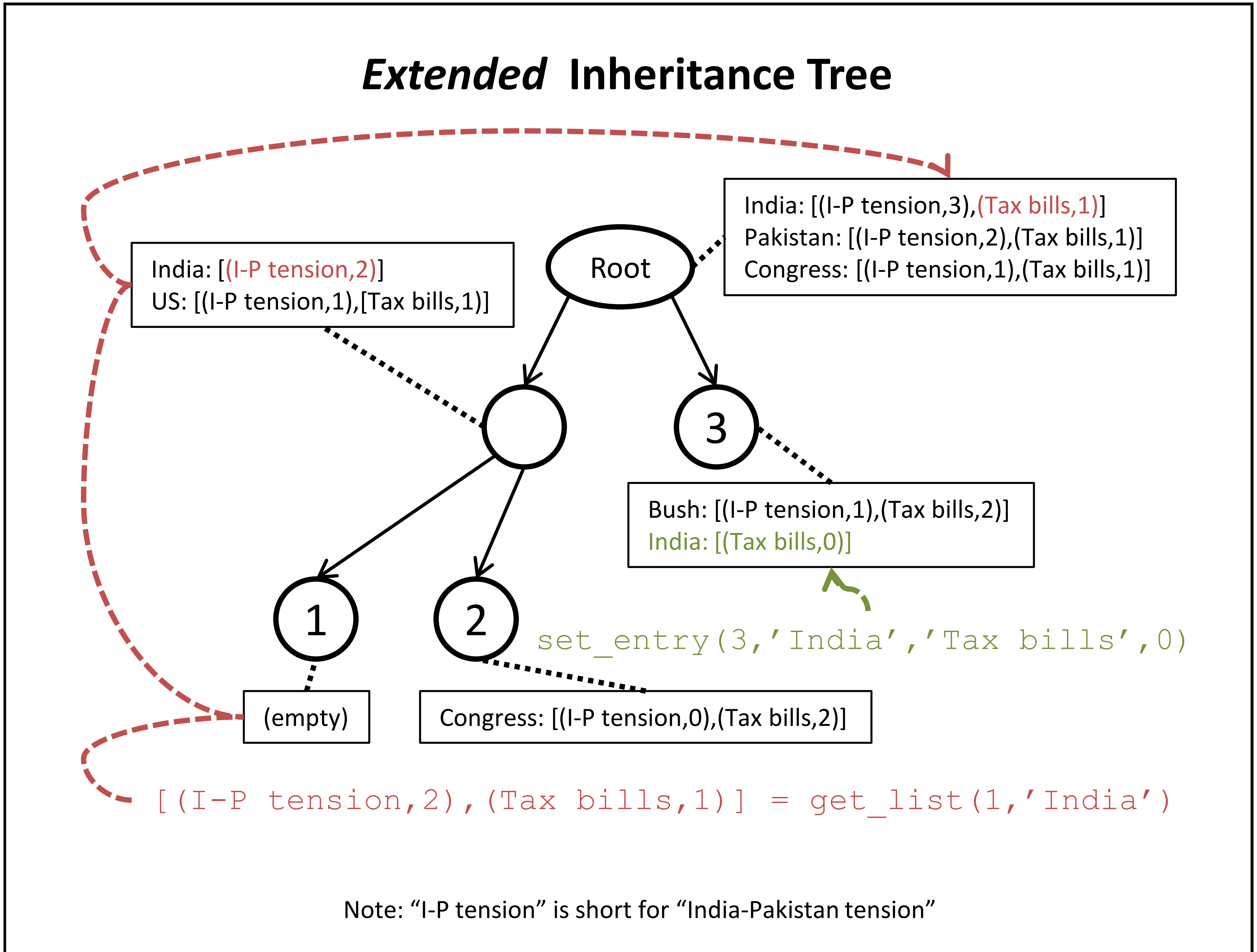
Inheritance Tree



Inheritance Tree



Extended Inheritance Tree



write only in the leaves (per thread)

Results

Ablation studies

- TDT5 (Topic Detection and Tracking) macro-averaged minimum detection cost: 0.714

- Removing features

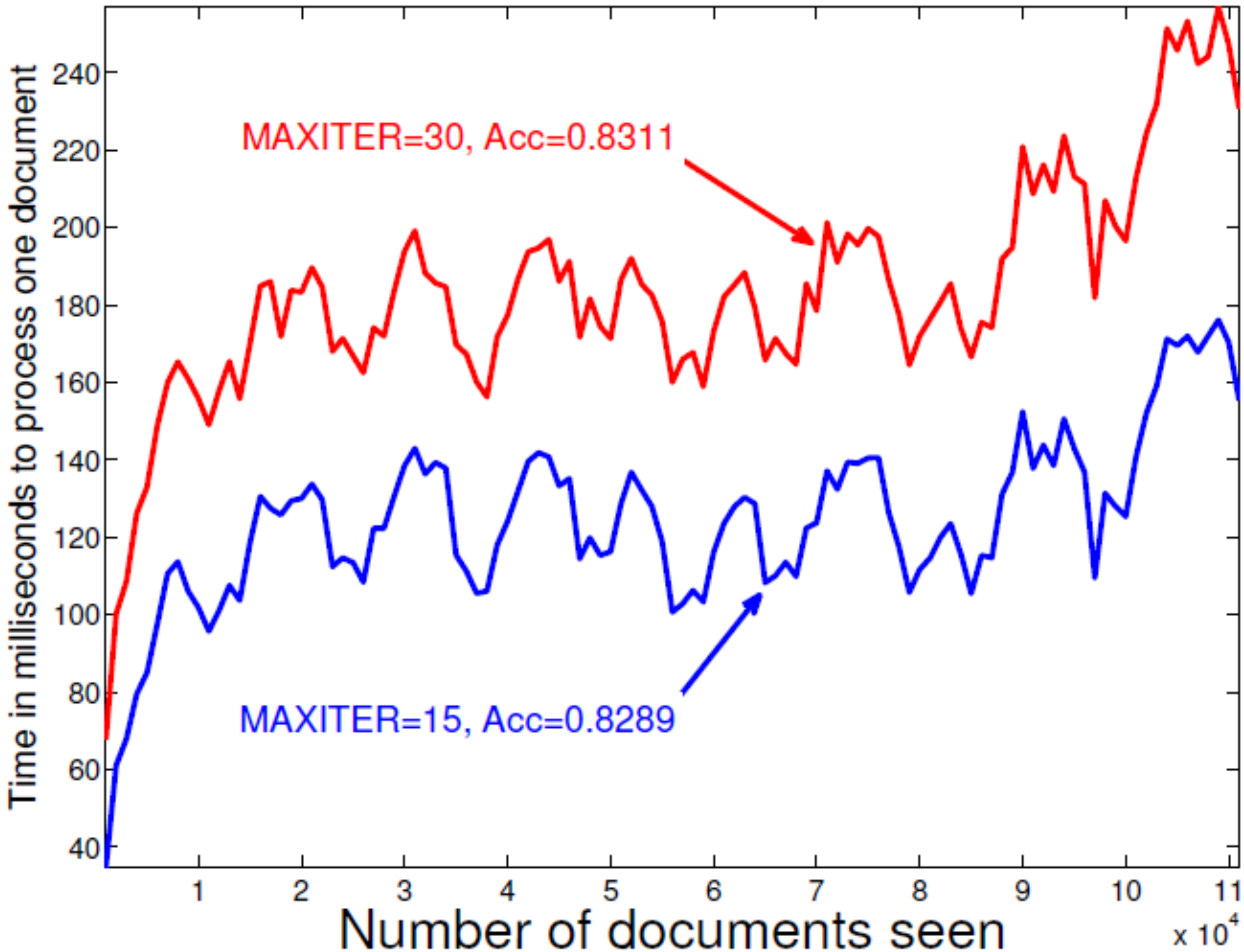
time	entities	topics	story words
0.84	0.90	0.86	0.75

Comparison

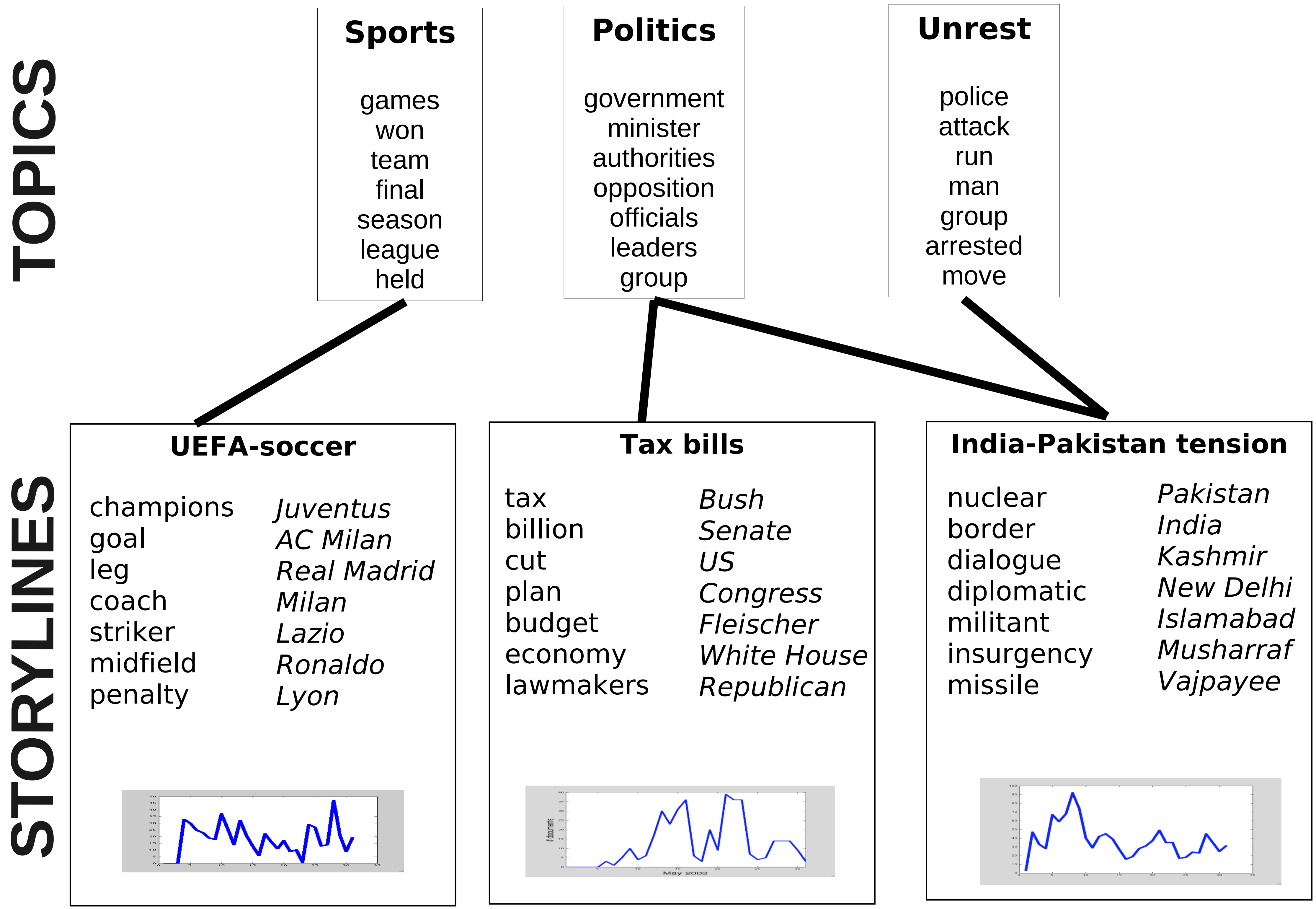
Sample No.	Sample size	Num Words	Num Entities	Story Acc.	LSHC Acc.
1	111,732	19,218	12,475	**0.8289**	0.738
2	274,969	29,604	21,797	**0.8388**	0.791
3	547,057	40,576	32,637	**0.8395**	0.800

Hashing & correlation clustering

Time-Accuracy trade off



Stories


TOPICS
Sports
games
won
team
final
season
league
held
Politics
government
minister
authorities
opposition
officials
leaders
group
Unrest
police
attack
run
man
group
arrested
move
STORYLINES
UEFA-soccer
champions
goal
leg
coach
striker
midfield
penalty
Juventus
AC Milan
Real Madrid
Milan
Lazio
Ronaldo
Lyon
Tax bills
tax
billion
cut
plan
budget
economy
lawmakers
Bush
Senate
US
Congress
Fleischer
White House
Republican
India-Pakistan tension
nuclear
border
dialogue
diplomatic
militant
insurgency
missile
Pakistan
India
Kashmir
New Delhi
Islamabad
Musharraf
Vajpayee
May 2003

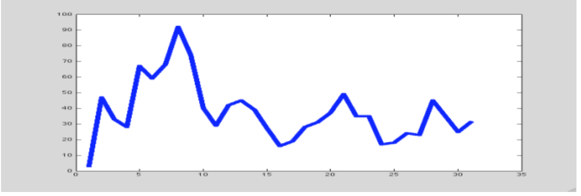
Related Stories

"Show similar stories by topic"

"Show similar stories, require the word nuclear*"*

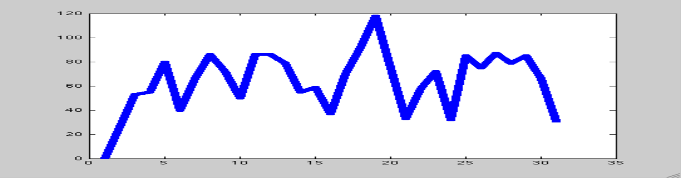
India-Pakistan tension

nuclear	*Pakistan*
border	*India*
dialogue	*Kashmir*
diplomatic	*New Delhi*
militant	*Islamabad*
insurgency	*Musharraf*
missile	*Vajpayee*



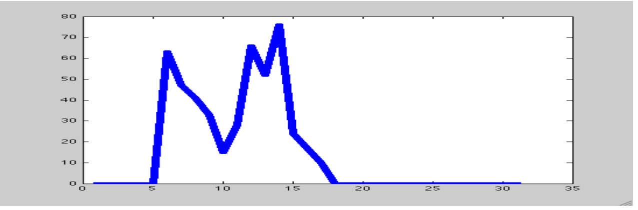
Middle-east conflict

Peace	*Israel*
Roadmap	*Palestinian*
Suicide	*West bank*
Violence	*Sharon*
Settlements	*Hamas*
bombing	*Arafat*



North Korea nuclear

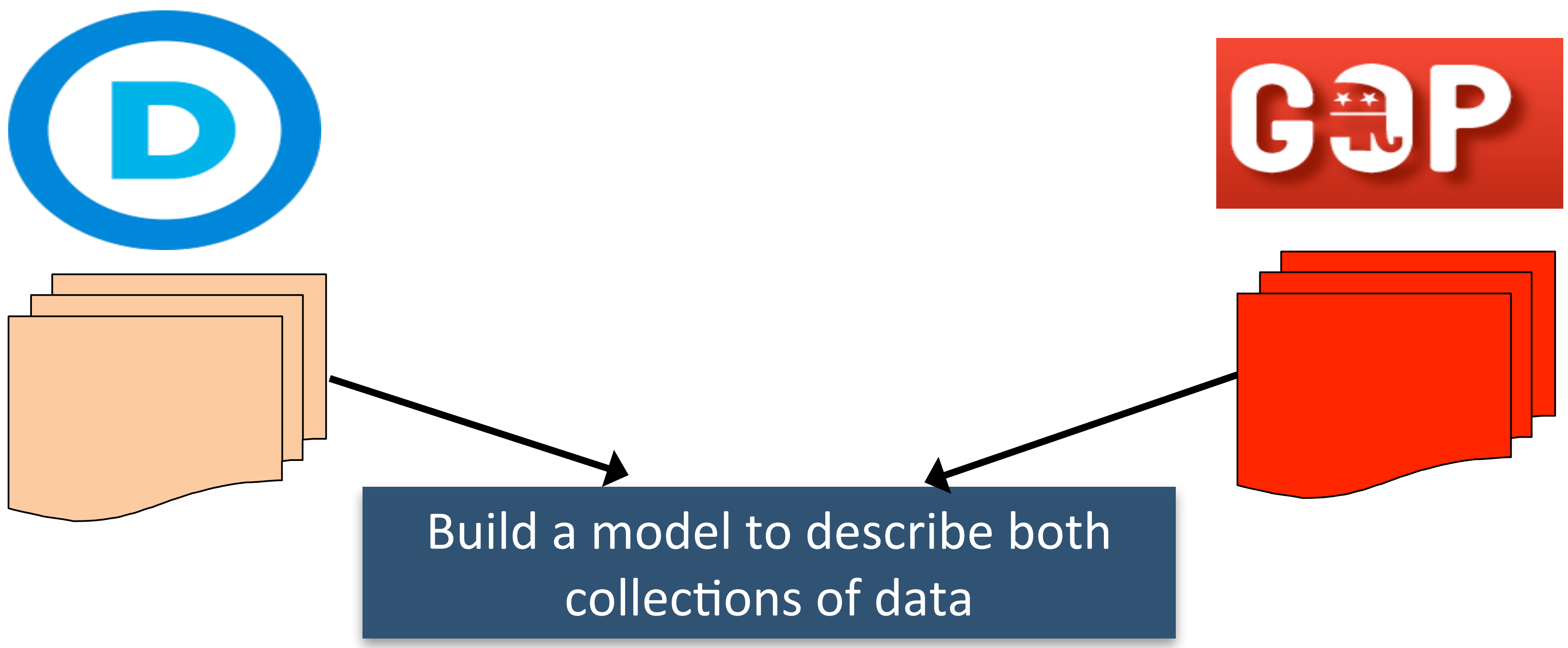
nuclear	*North Korea*
summit	*South Korea*
warning	*U.S*
policy	*Bush*
missile	*Pyongyang*
program	



Detecting Ideologies

Ahmed and Xing, 2010

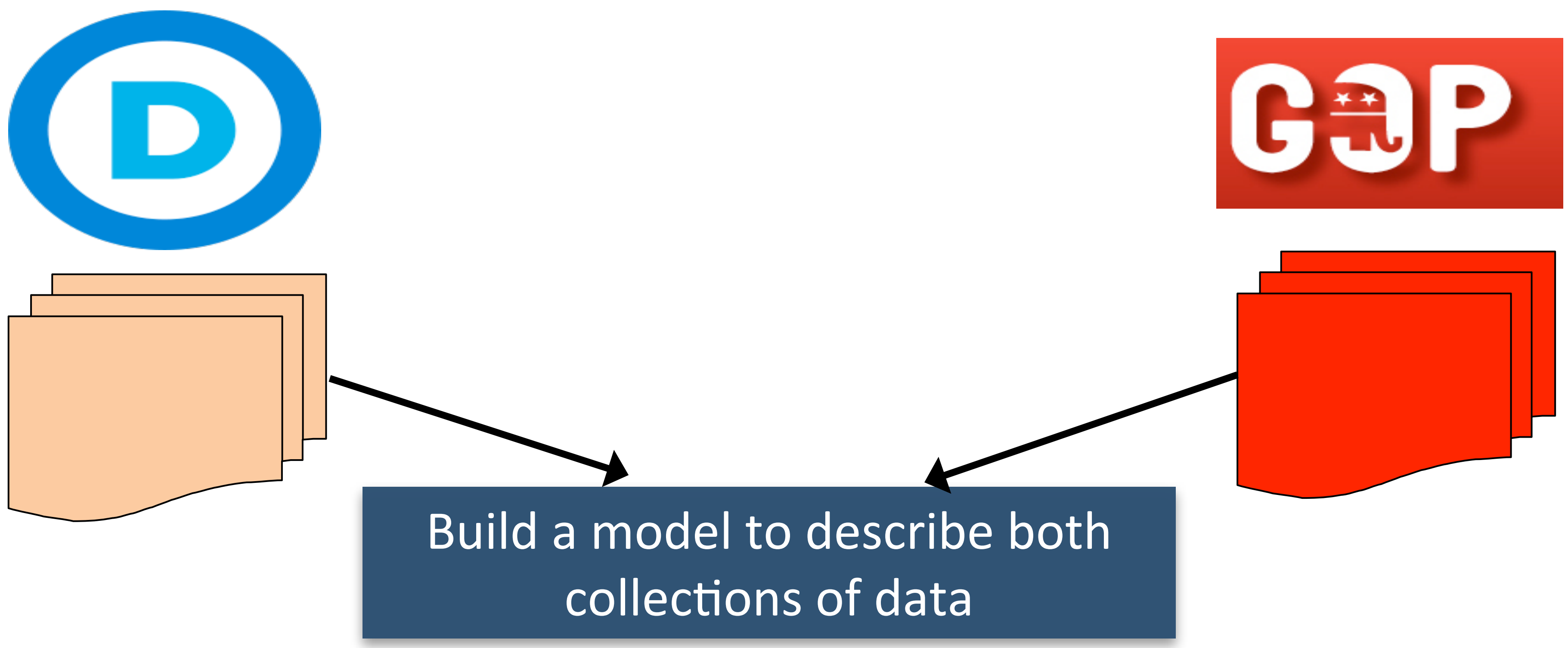
Ideologies



Visualization

- How does each ideology **view** mainstream events?
- On which topics do they **differ**?
- On which topics do they **agree**?

Ideologies

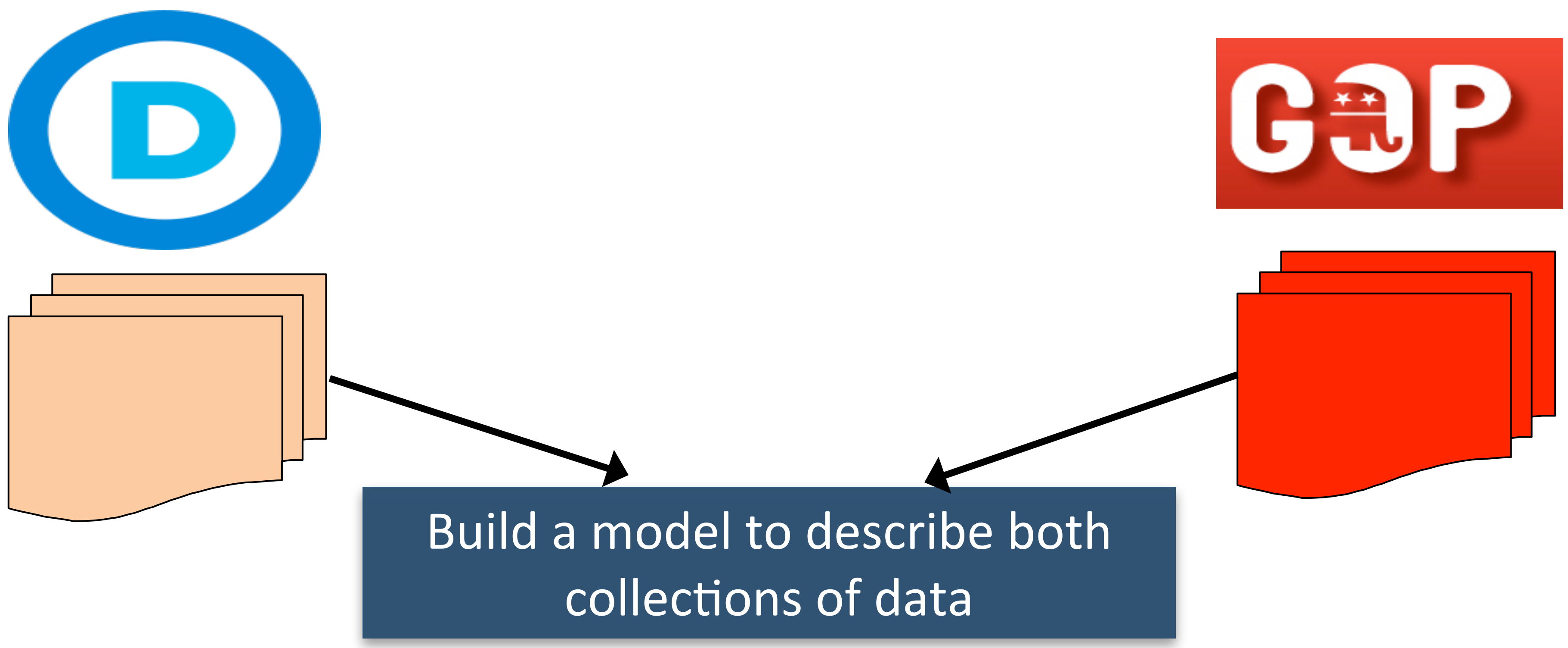


Visualization

Classification

- Given a new news article or a blog post, the system should infer
  - From which side it was written
  - Justify its answer on a topical level (view on abortion, taxes, health care)

Ideologies



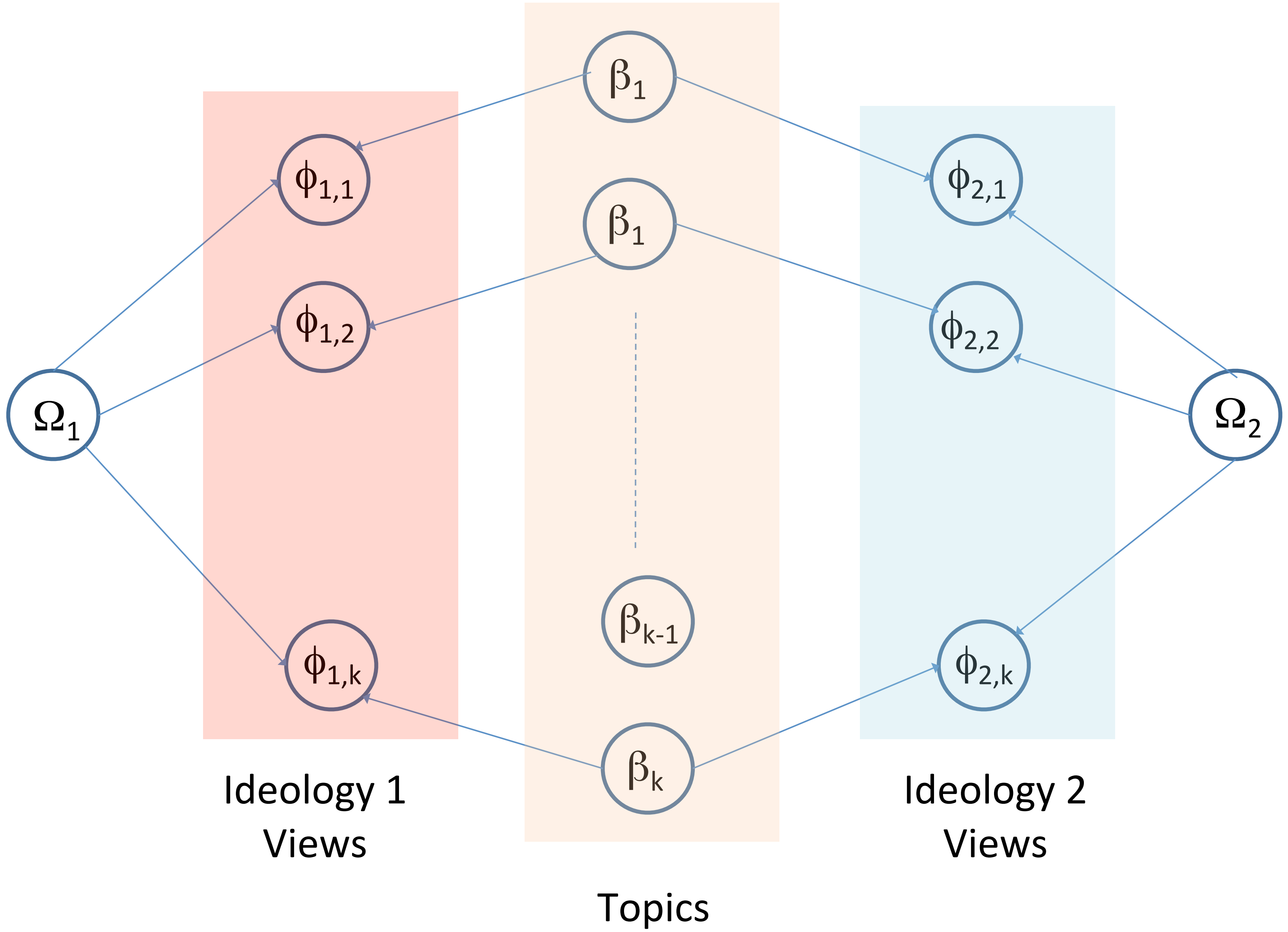
Visualization

Classification

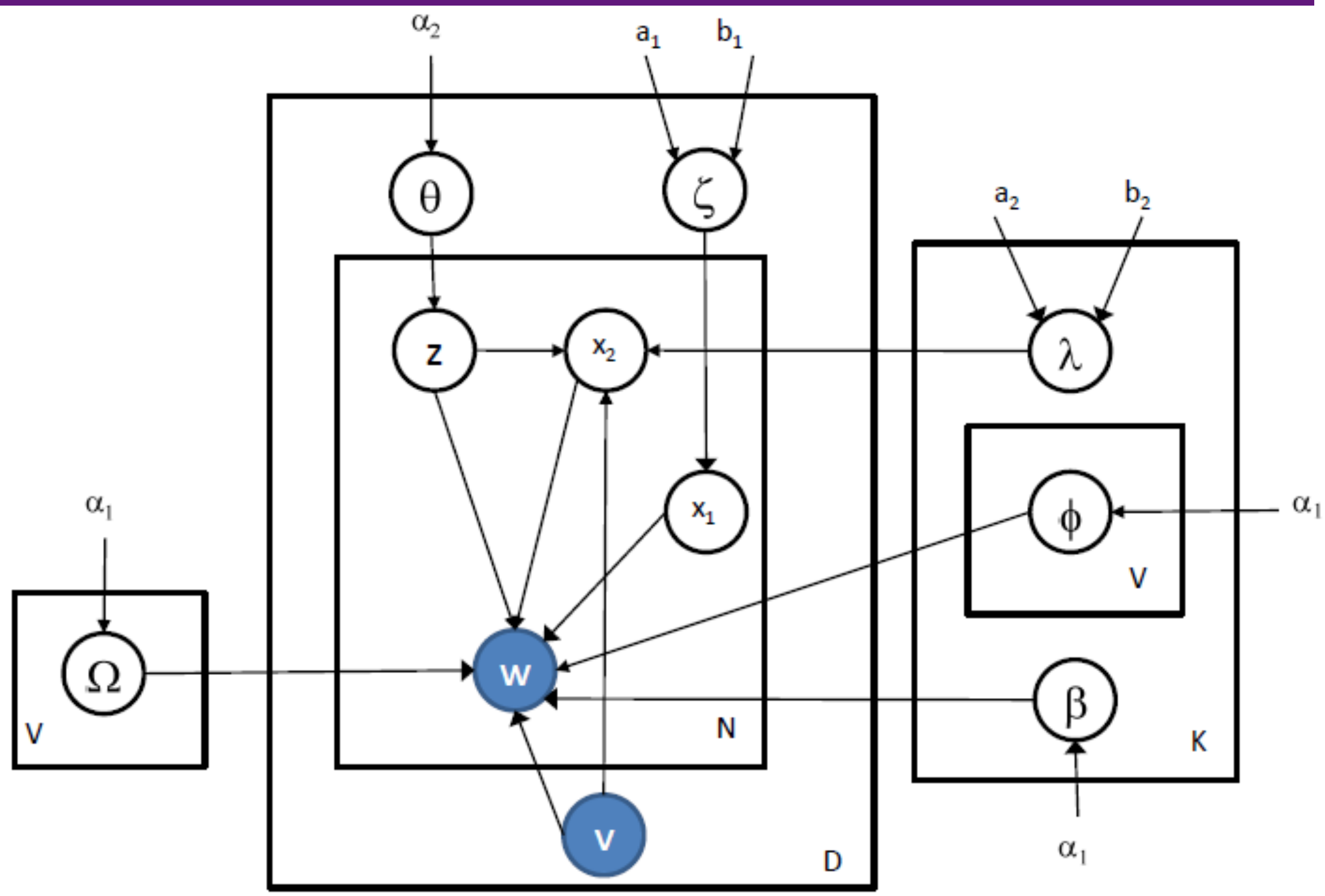
Structured browsing

- Given a **new** news article or a blog post, the user can ask for :
  - Examples of other articles from the same ideology about the same topic
  - Documents that could exemplify **alternative** views from **other ideologies**

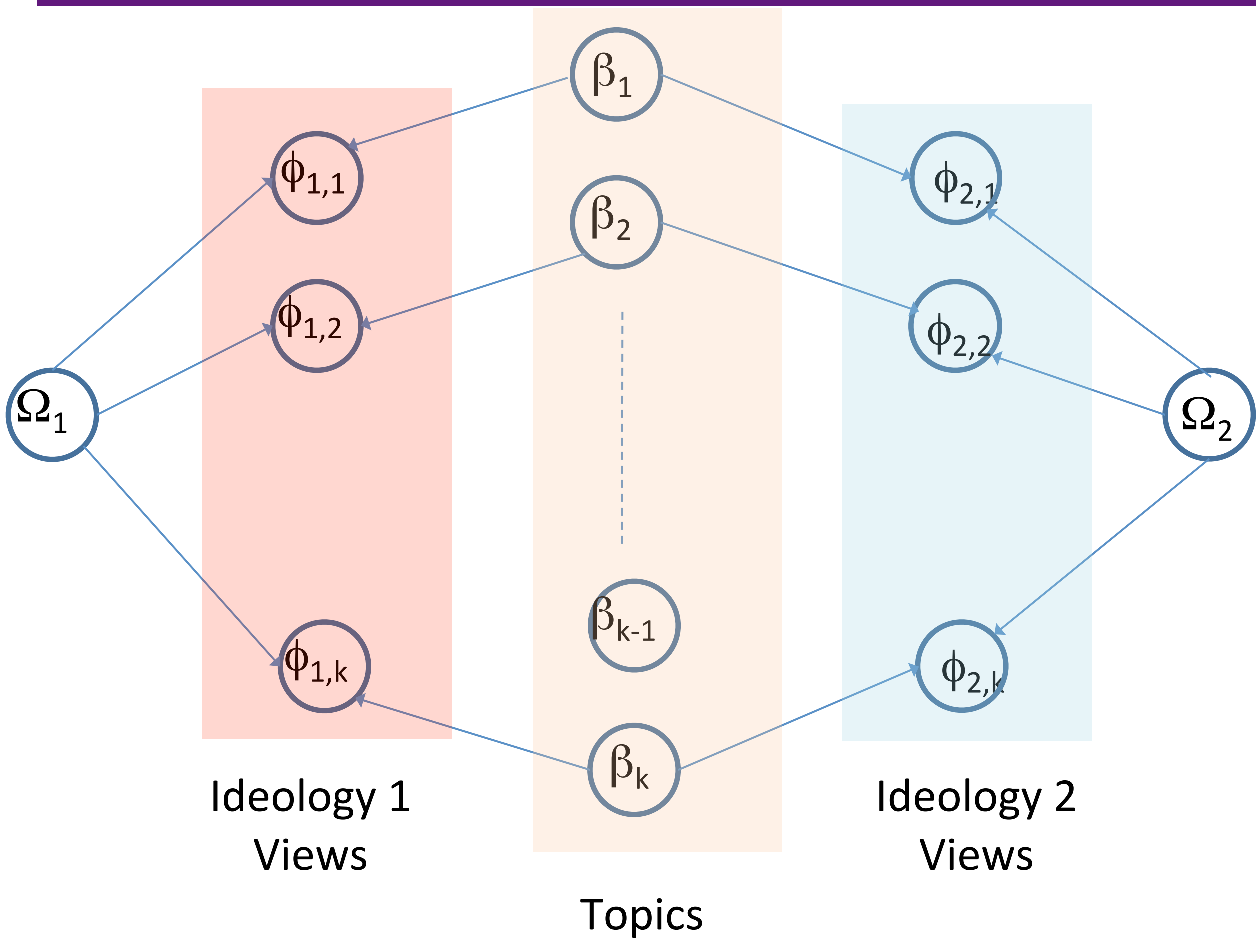
Building a factored model

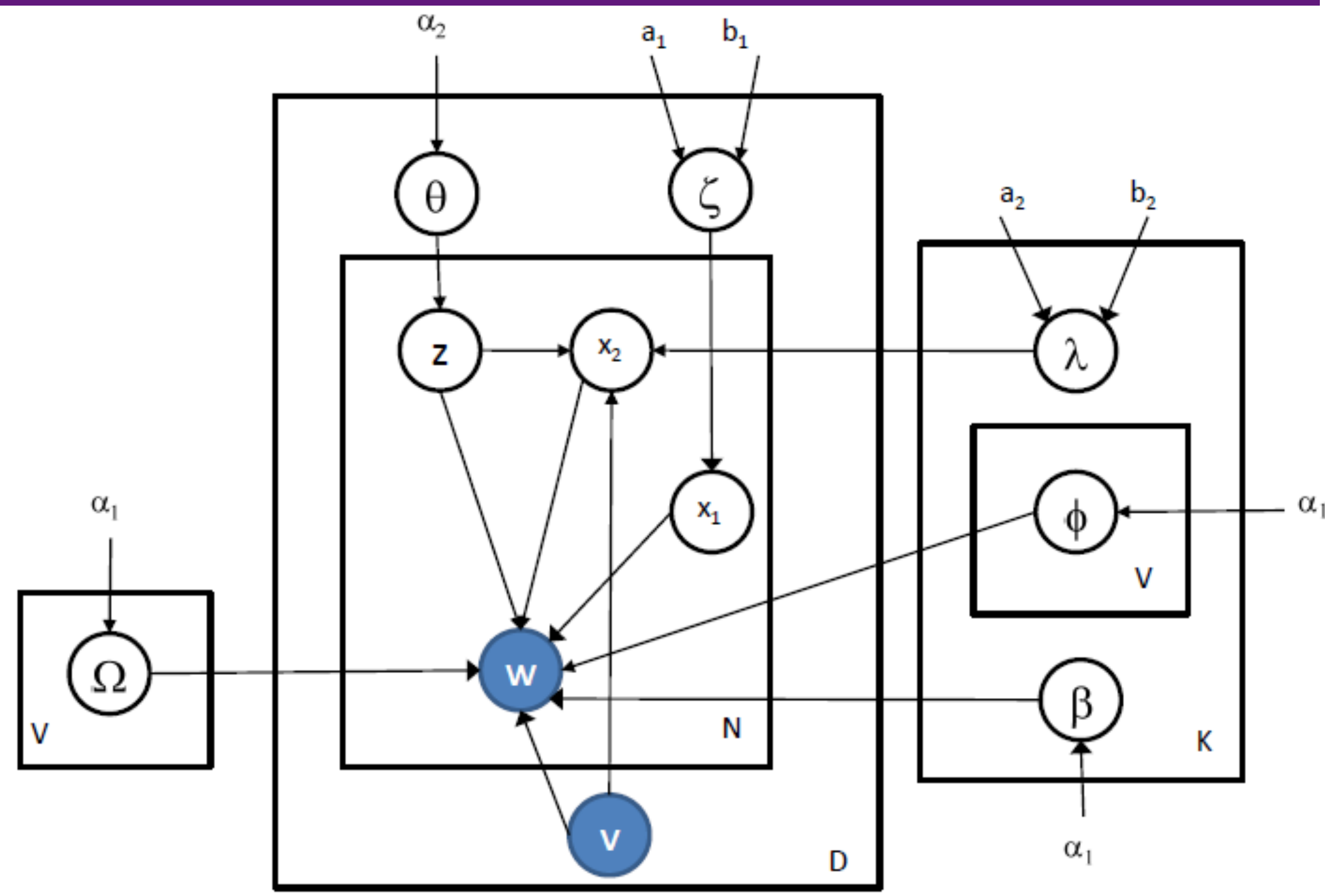


Building a factored model

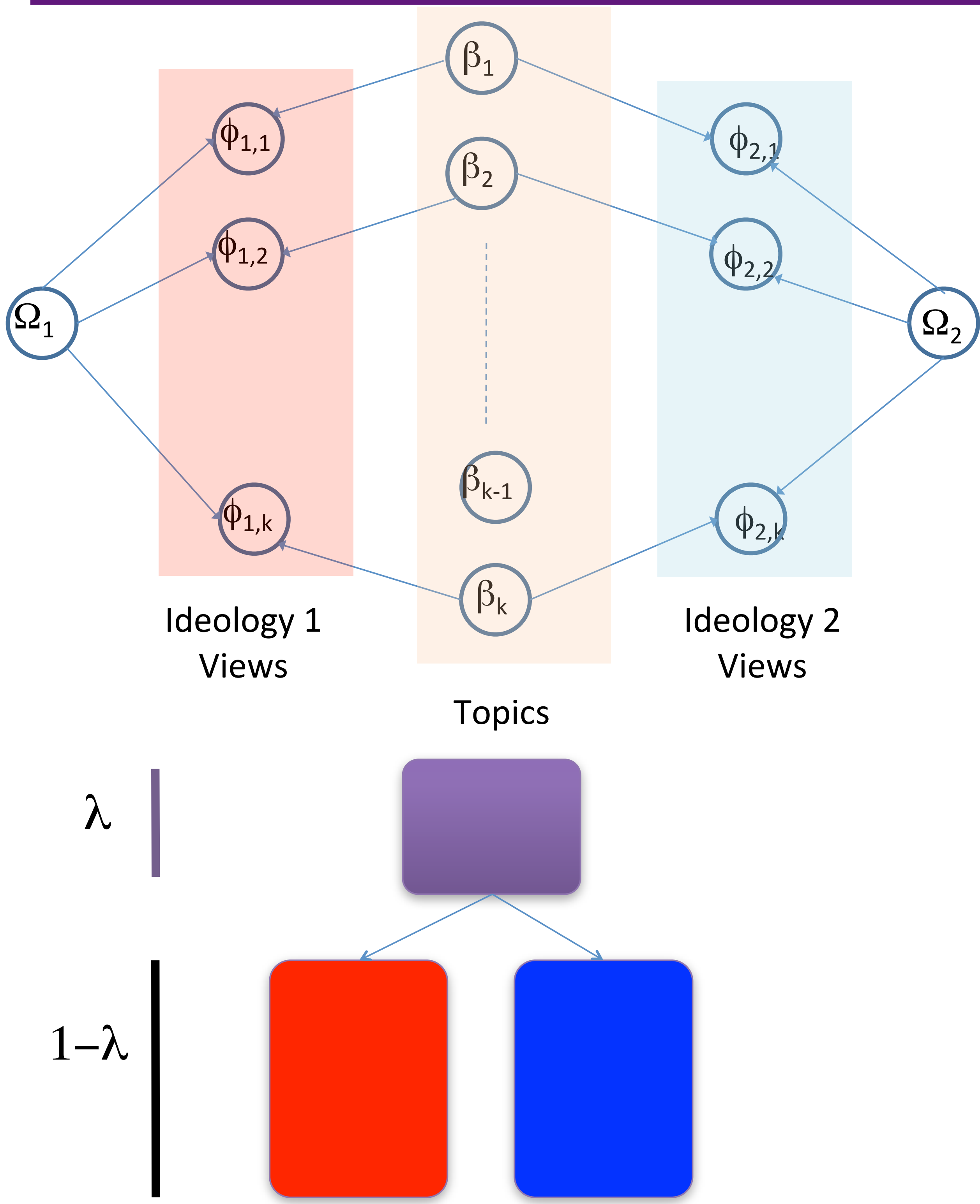


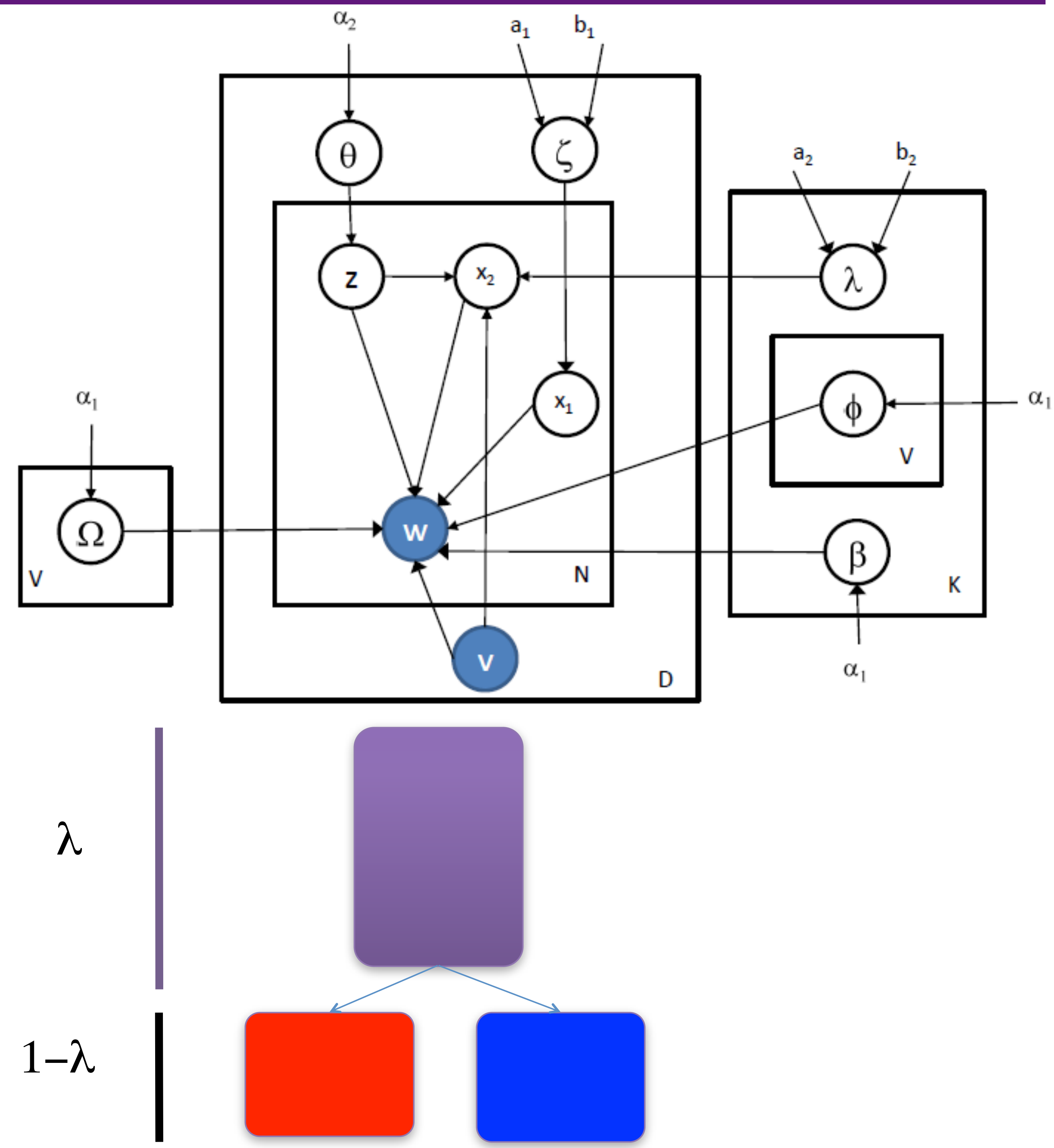
Building a factored model





Building a factored model

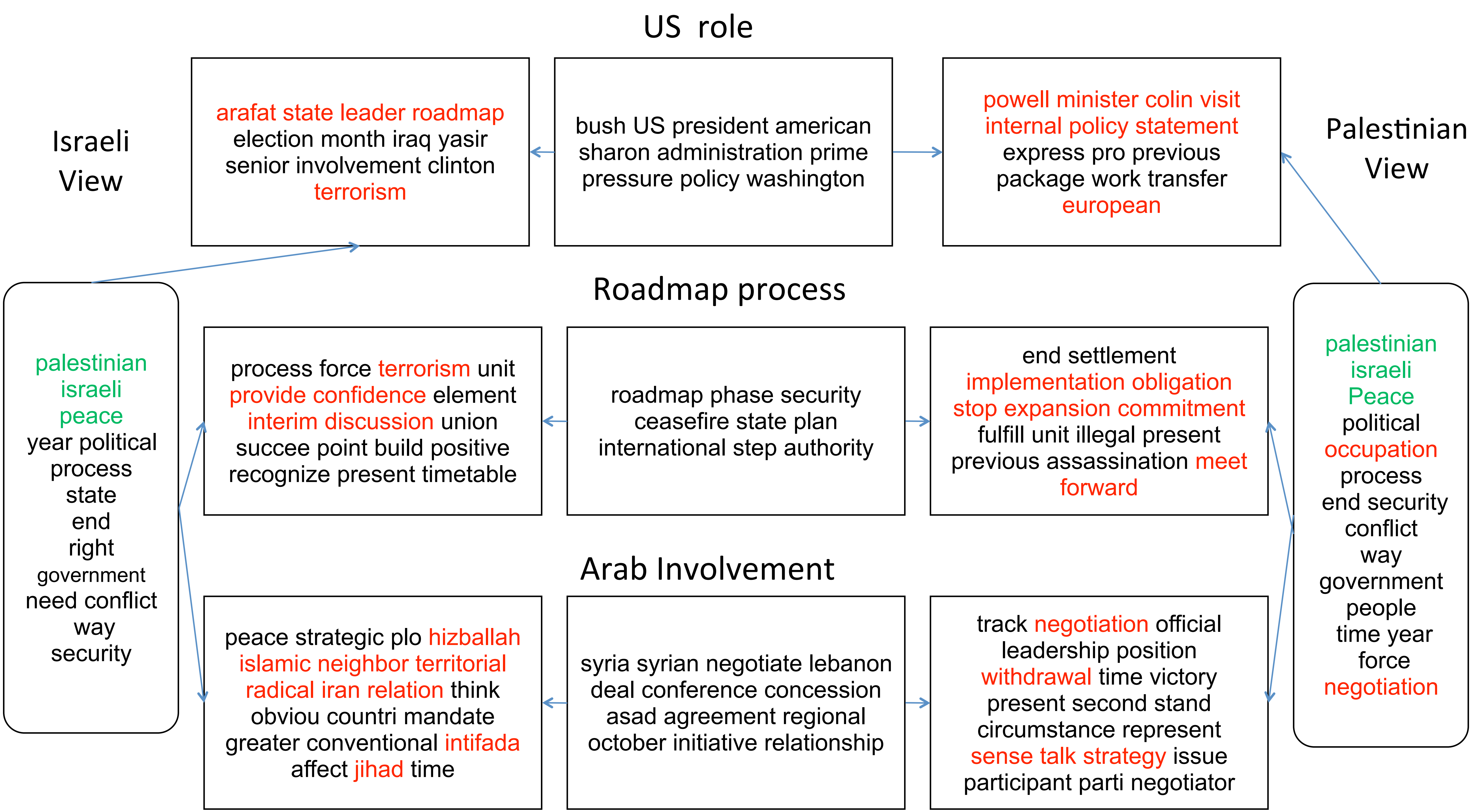




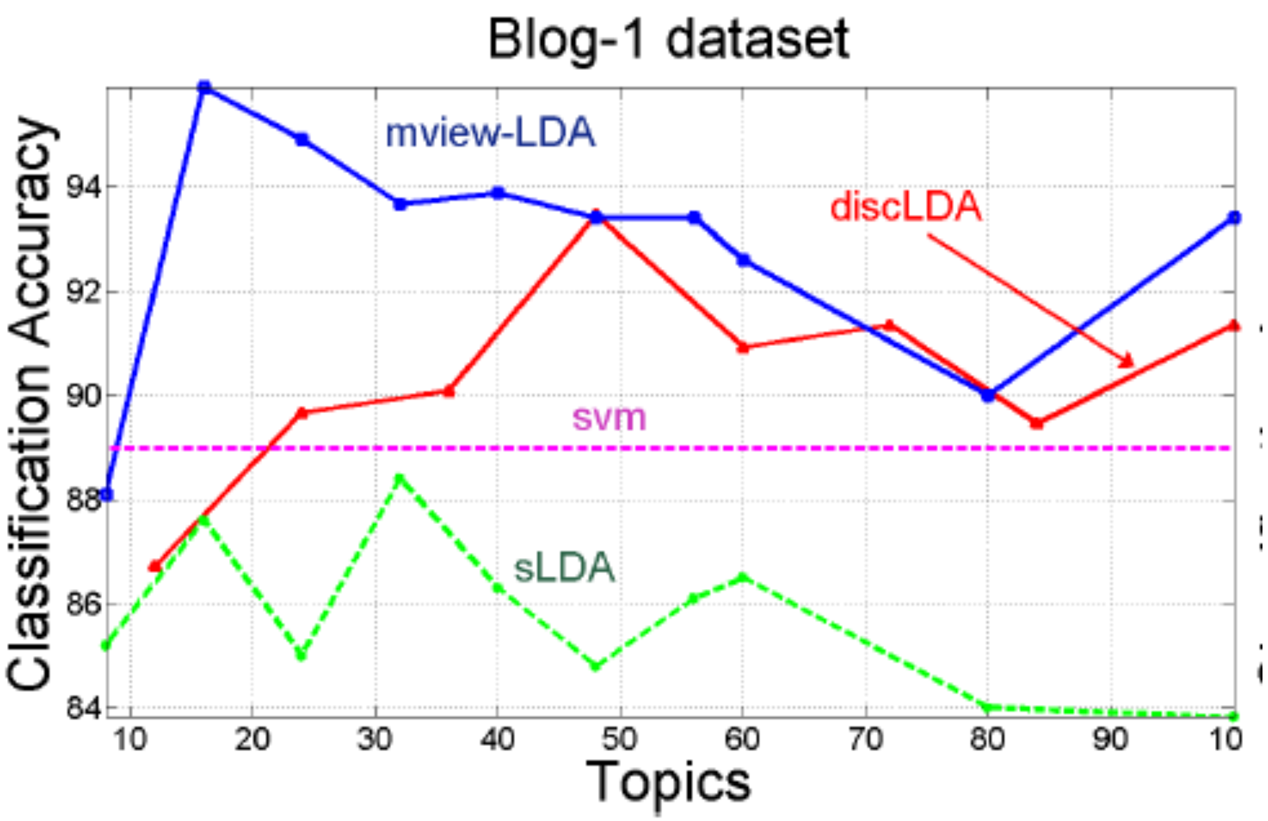
Data

- **Bitterlemons**:
  - Middle-east conflict, document written by Israeli and Palestinian authors.
  - ~300 documents form each view with average length 740
  - Multi author collection
  - 80-20 split for test and train
- **Political Blog-1:**
  - American political blogs (Democrat and Republican)
  - 2040 posts with average post length = 100 words
  - Follow test and train split as in (Yano et al., 2009)
- **Political Blog-**2 (test generalization to a new writing style)
  - Same as 1 but 6 blogs, 3 from each side
  - ~14k posts with ~200 words per post
  - 4 blogs for training and 2 blogs for test

Bitterlemons dataset

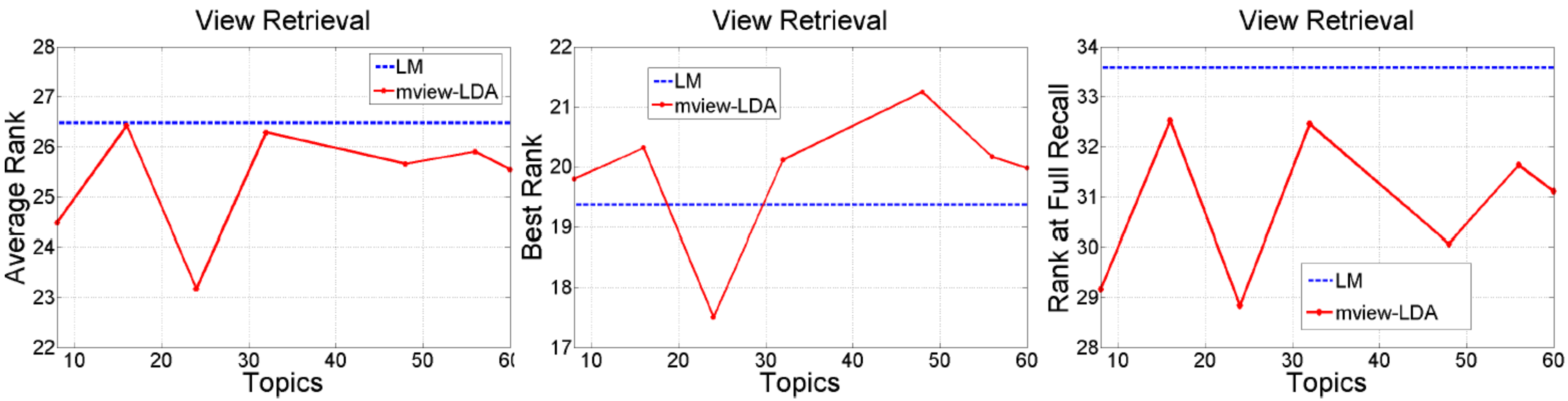


Classification accuracy



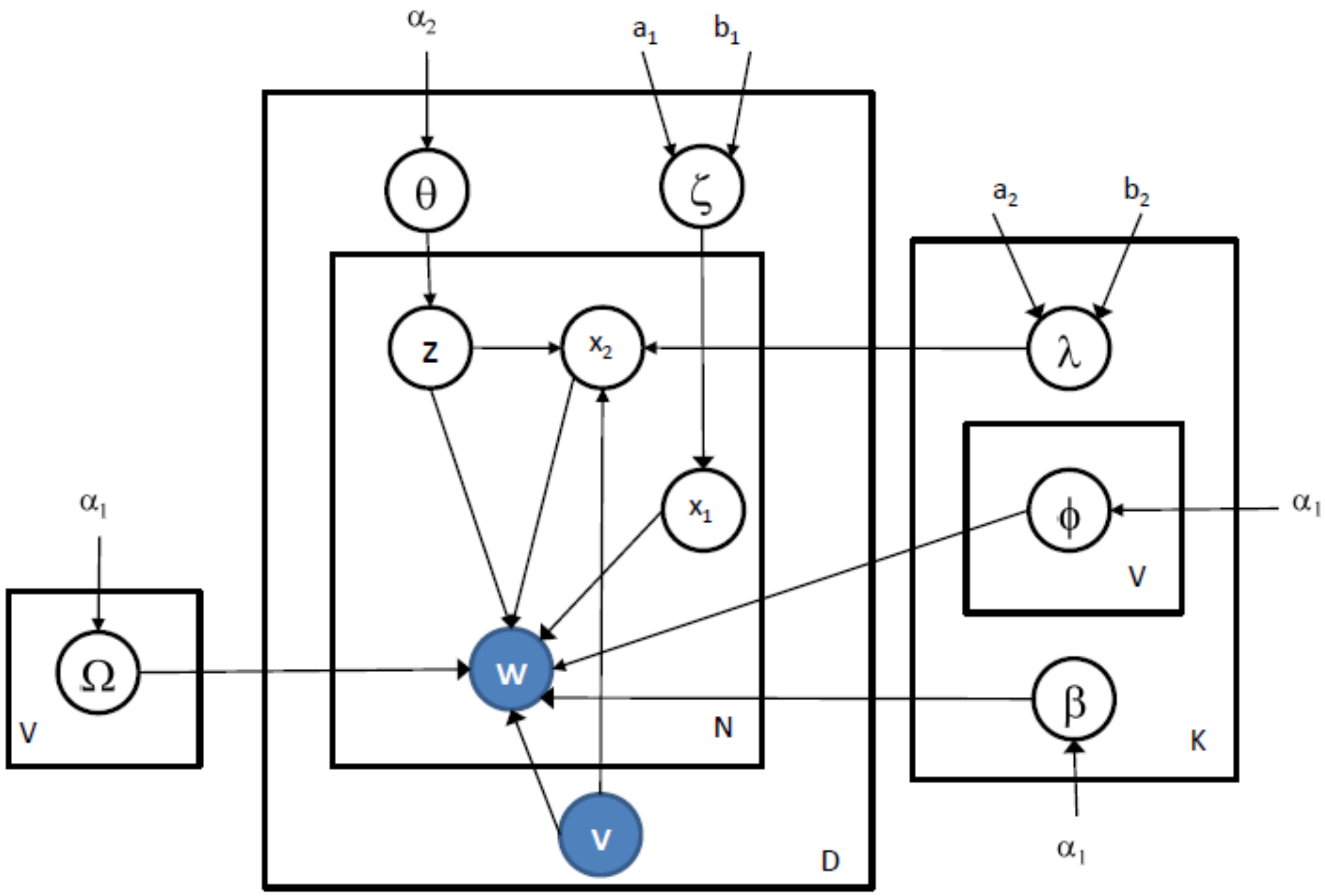
Generalization to new blogs

Finding alternate views

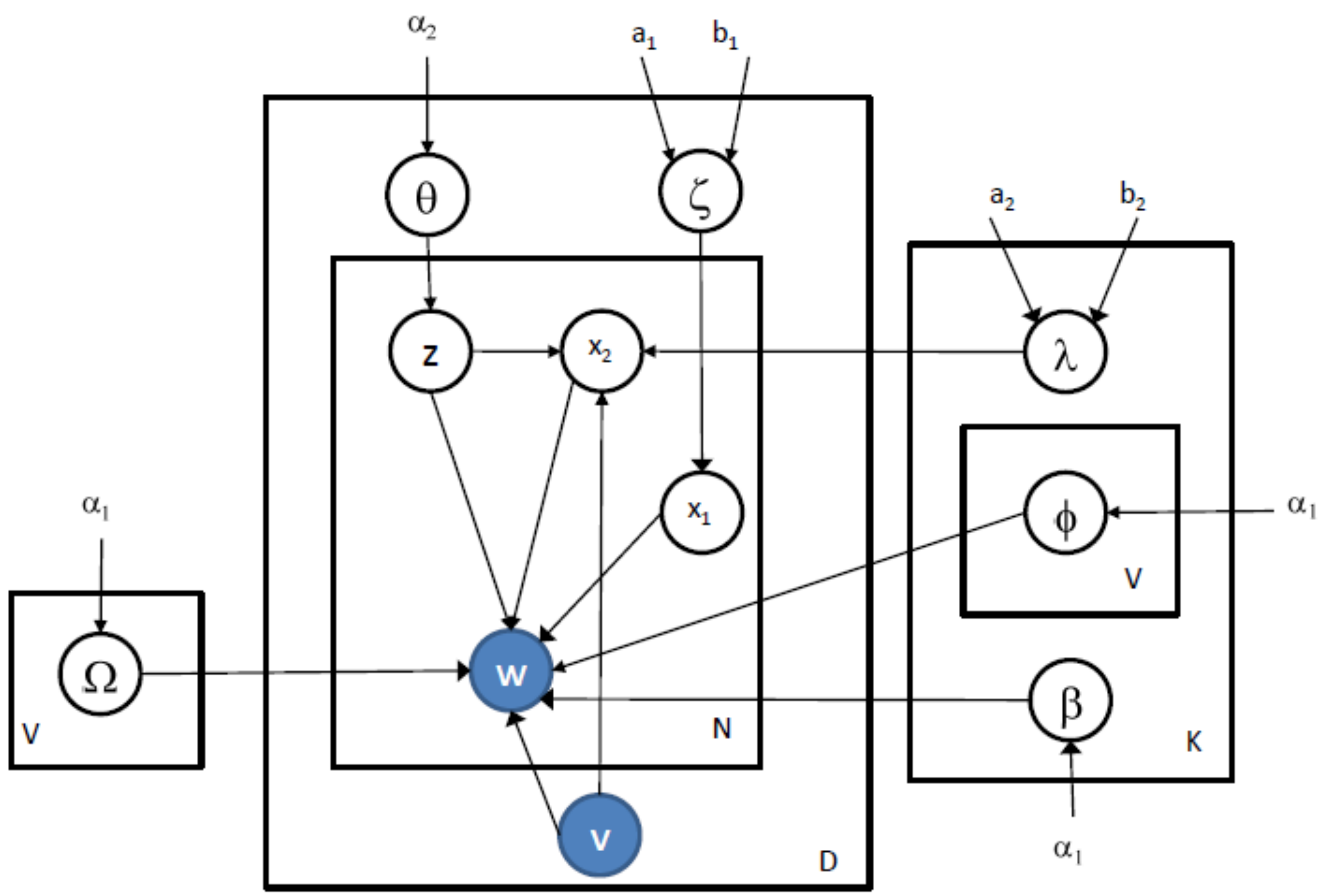


- Given a document written in one ideology, retrieve the equivalent
- Baseline: SVM + cosine similarity

Unlabeled data

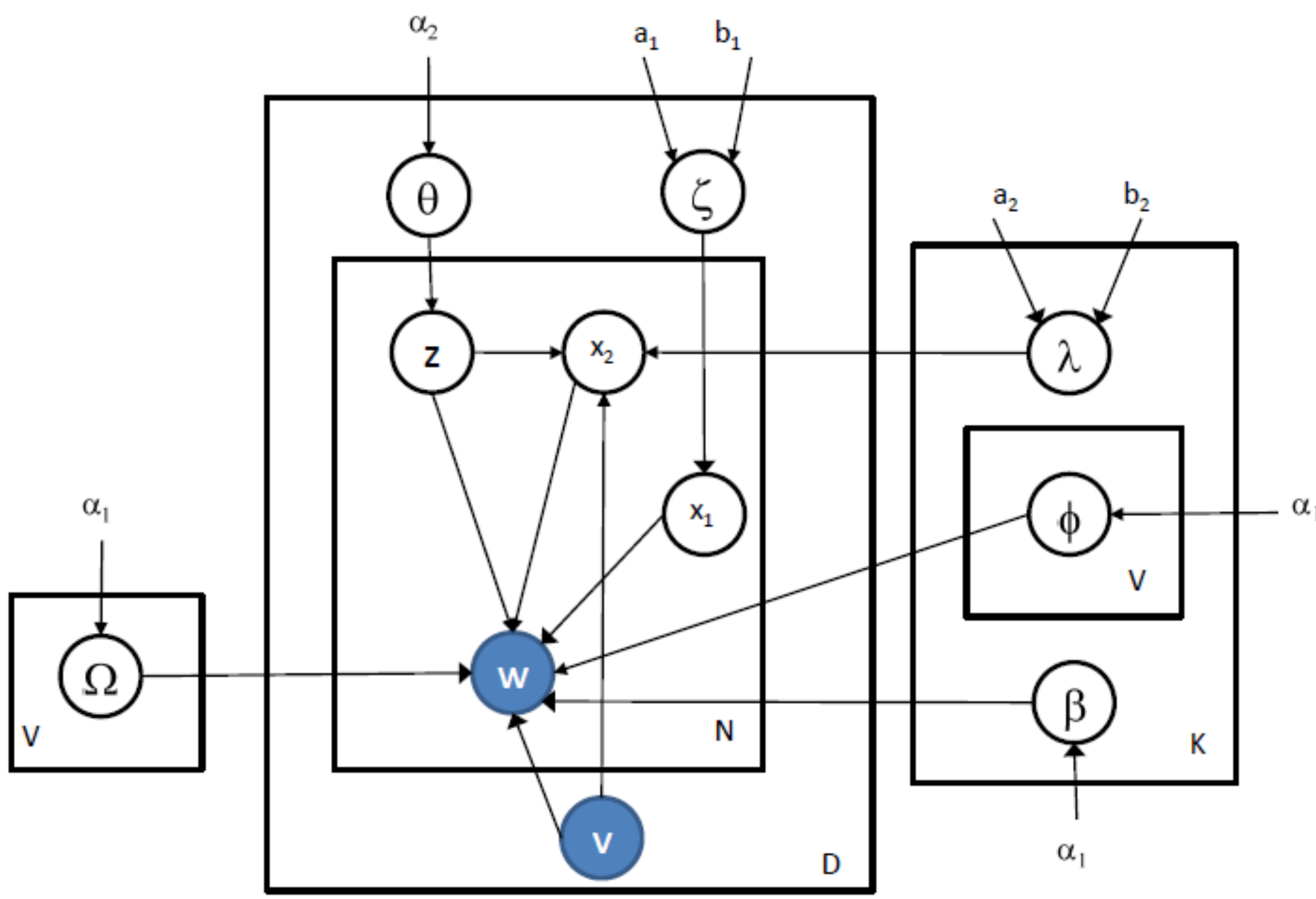


Unlabeled data



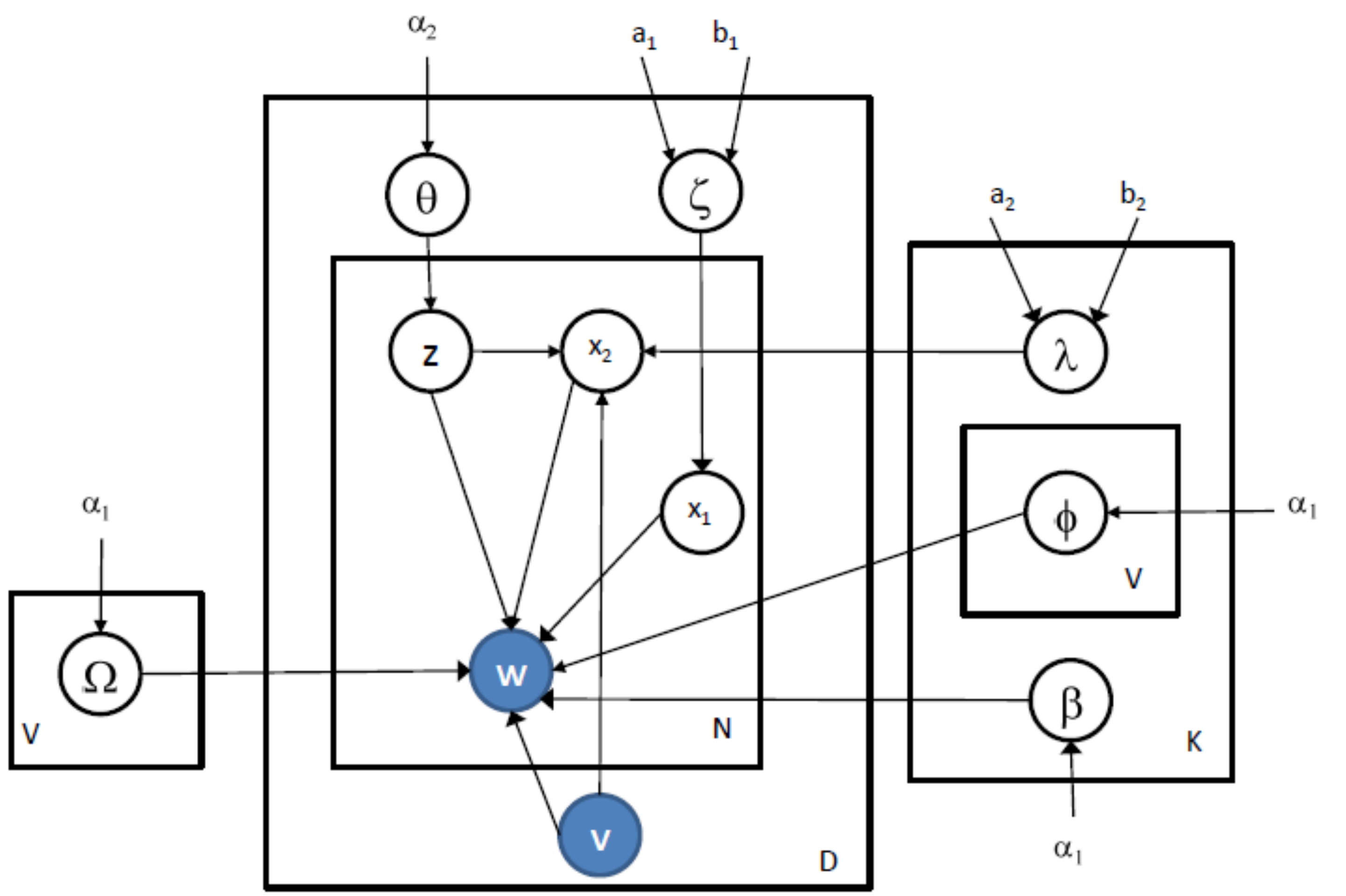
- In theory this is simple
  - Add a step that samples the document view (v)
  - **Doesn't mix** in practice because tight coupling between v and ($\mathbf{x}_1$,$\mathbf{x}_2$,$\mathbf{z}$)

Unlabeled data



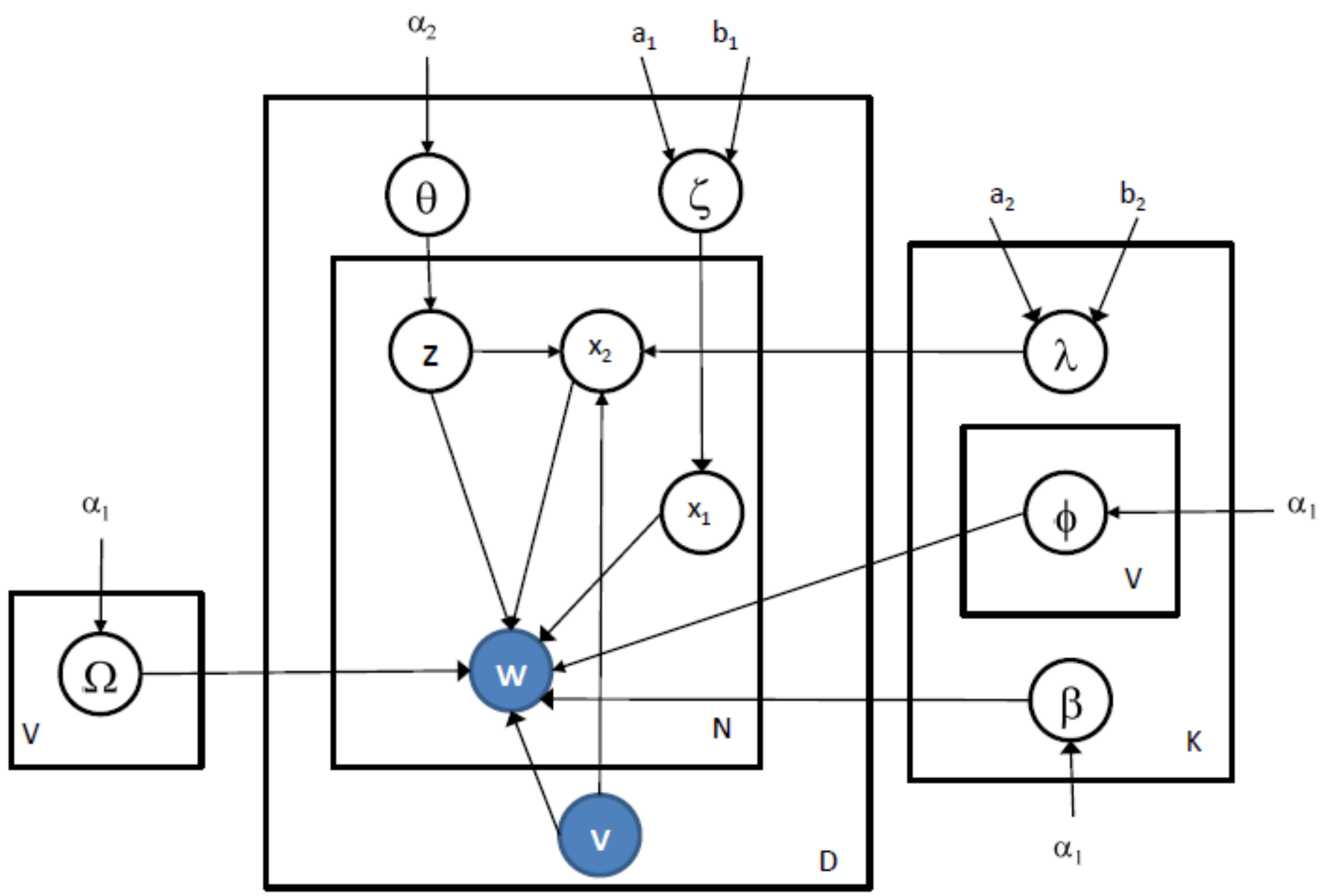
- In theory this is simple
    - Add a step that samples the document view (v)
    - **Doesn't mix** in practice because tight coupling between v and ($\mathbf{x}_1$,$\mathbf{x}_2$,$\mathbf{z}$)
- Solution

Unlabeled data



- In theory this is simple
  - Add a step that samples the document view (v)
  - **Doesn't mix** in practice because tight coupling between v and ($\mathbf{x}_1$,$\mathbf{x}_2$,$\mathbf{z}$)
- Solution
  - Sample v and ($\mathbf{x}_1$,$\mathbf{x}_2$,$\mathbf{z}$) as a block using a Metropolis-Hasting step

Unlabeled data



- In theory this is simple
  - Add a step that samples the document view (v)
  - **Doesn't mix** in practice because tight coupling between v and ($\mathbf{x}_1$,$\mathbf{x}_2$,$\mathbf{z}$)
- Solution
  - Sample v and ($\mathbf{x}_1$,$\mathbf{x}_2$,$\mathbf{z}$) as a block using a Metropolis-Hasting step
  - This is a **huge proposal**!