Universität Potsdam
Institut für Informatik
Lehrstuhl Maschinelles Lernen

Graphical Models

Niels Landwehr

Agenda

- Graphical models: syntax and semantics.
- Inference in graphical models (exact, approximate)
- Graphical models in machine learning
  - Recap: Coin Tosses
  - Recap: Bayesian linear regression
  - Latent Dirichlet allocation
  - Hidden Markov models

Recap: Bayesian Linear Regression

- Solving regression problems

$$L = \langle (\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \rangle$$

$\mathbf{x}_i \in \mathbb{R}^m$ feature vector

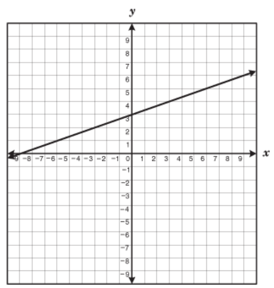
$y_i \in \mathbb{R}$ real-valued target

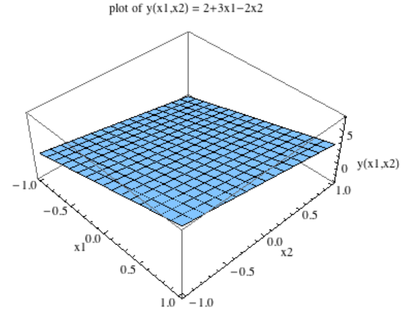
$$\mathbf{X} = \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{pmatrix} \qquad \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

- Linear regression

$$f_{\mathbf{w}}(\mathbf{x}) = \mathbf{w}^T \mathbf{x} = \sum_{i=1}^{m} w_i x_i$$

$\mathbf{w}$ „parameter vector“, „weight vector“





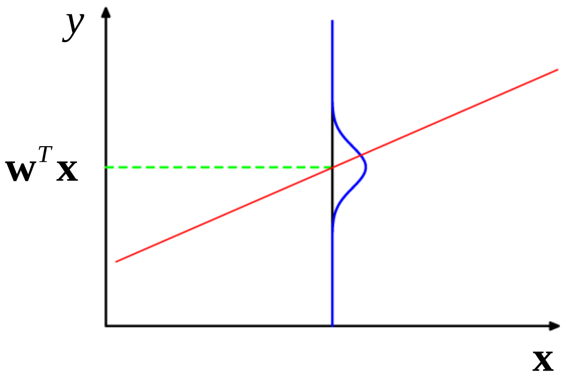
Recap: Bayesian Linear Regression

- Discriminative setting: $\mathbf{x}_i$ fixed input, y_i generated from $\mathbf{x}_i$ and $\mathbf{w}$ plus Gaussian noise.

$$p(y \mid \mathbf{x}, \mathbf{w}) = \mathbf{w}^T \mathbf{x} + N(y \mid 0, \sigma^2)$$
$$= N(y \mid \mathbf{w}^T \mathbf{x}, \sigma^2)$$

$$y_i \sim p(y \mid \mathbf{x}_i, \mathbf{w})$$

discriminative model: $p(\mathbf{x})$ not modeled



- Bayesian approach: posterior $\propto$ prior x likelihood

$$\underbrace{p(\mathbf{w} \mid L)}_{posterior} \propto \underbrace{p(L \mid \mathbf{w})}_{likelihood} \underbrace{p(\mathbf{w})}_{prior}$$

Recap: Bayesian Linear Regression

- Likelihood of data under model **w**:

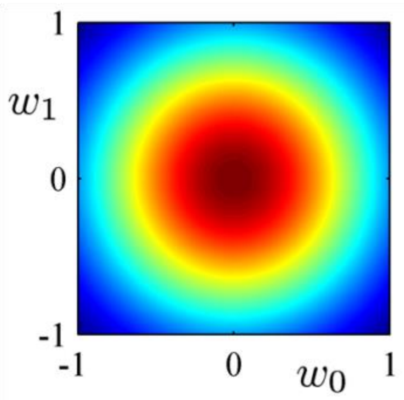
$$p(\mathbf{y} \mid \mathbf{X}, \mathbf{w}) = \prod_{i=1}^{N} p(y_i \mid \mathbf{x}_i, \mathbf{w}, \sigma^2) \qquad i.i.d.$$

$$= \prod_{i=1}^{N} N(y_i \mid \mathbf{w}^T \mathbf{x}_i, \sigma^2)$$

- Normally distributed prior over models:

$$p(\mathbf{w}) = N(\mathbf{w} \mid \mathbf{0}, \tau^2 I)$$

$$I = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix}$$



Isotropic multivariate normal distribution, mean **0,** variance τ^2

Bayesian Linear Regression as a Graphical Model

- What are the random variables?
  - Labels $y_1, \ldots, y_N$, model $\mathbf{w}$
  - Not: $\mathbf{x}_1, \ldots, \mathbf{x}_N$, hyperparameters σ^2, τ^2
  - Inputs $\mathbf{x}_i$ behave like hyperparameters (fixed quantities)

- Joint distribution over labels and parameter vector

$$p(y_1,\ldots,y_N,\mathbf{w} \mid \mathbf{x}_1,\ldots,\mathbf{x}_N,\sigma^2,\tau^2) = \overbrace{p(\mathbf{w} \mid \tau^2)}^{\text{Prior}} \overbrace{p(y_1,\ldots,y_N \mid \mathbf{w},\mathbf{x}_1,\ldots,\mathbf{x}_N,\sigma^2)}^{\text{Likelihood}}$$

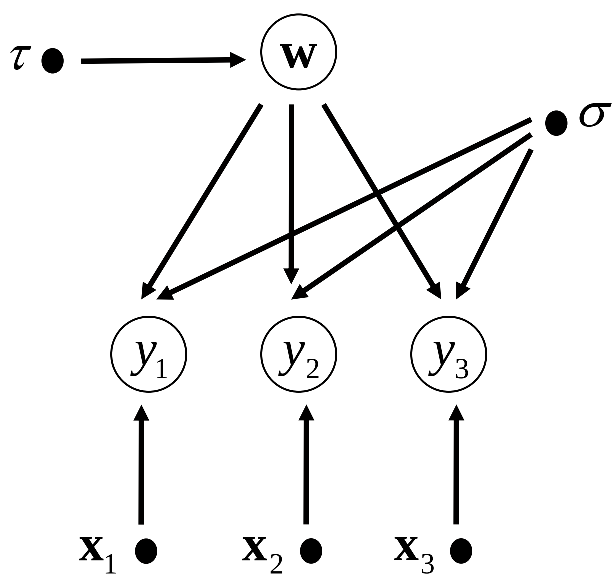
$$= p(\mathbf{w} \mid \tau^2) \prod_{i=1}^{N} p(y_i \mid \mathbf{w},\mathbf{x}_i,\sigma^2)$$

- Representation of Bayesian linear regression as a graphical model: structure can be seen from the form of the joint distribution.

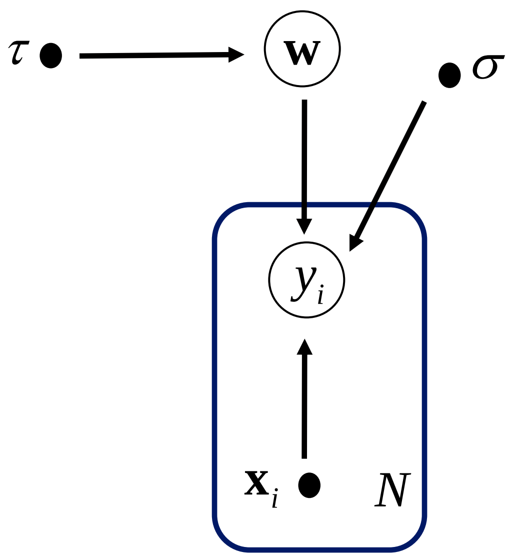
Bayesian Linear Regression as a Graphical Model

$$p(y_1,\ldots,y_N,\mathbf{w}\mid\mathbf{x}_1,\ldots,\mathbf{x}_N,\sigma^2,\tau^2)=p(\mathbf{w}\mid\tau^2)\prod_{i=1}^{N}p(y_i\mid\mathbf{x}_i,\mathbf{w},\sigma^2)$$

Graphical model, *N*=3



Graphical model, Plate notation



Bayes-optimal Prediction

- When applying model, need prediction for novel test instance $\mathbf{x}$:

$$\mathbf{x} \mapsto y$$

- Bayesian prediction

$$\begin{aligned} y_* &= \arg\max_y \, p(y \mid \mathbf{x}, \mathbf{y}, \mathbf{X}, \sigma^2, \tau^2) \\ &= \arg\max_y \int p(y \mid \mathbf{x}, \mathbf{w}, \sigma^2)\, p(\mathbf{w} \mid \mathbf{y}, \mathbf{X}, \sigma^2, \tau^2)\, d\mathbf{w} \end{aligned}$$

„Bayesian model averaging“

Bayesian Linear Regression as a Graphical Model

- Graphical model representation of Bayesian linear regression, including a novel test instance.

$$p(y_1,\ldots,y_N,y,\mathbf{w}\,|\,\mathbf{x}_1,\ldots,\mathbf{x}_N,\mathbf{x},\sigma^2,\tau^2) = p(\mathbf{w}\,|\,\tau^2)\left(\prod_{i=1}^{N} p(y_i\,|\,\mathbf{w},\mathbf{x}_i,\sigma^2)\right)p(y\,|\,\mathbf{w},\mathbf{x},\sigma^2)$$

Graphical model, N=3

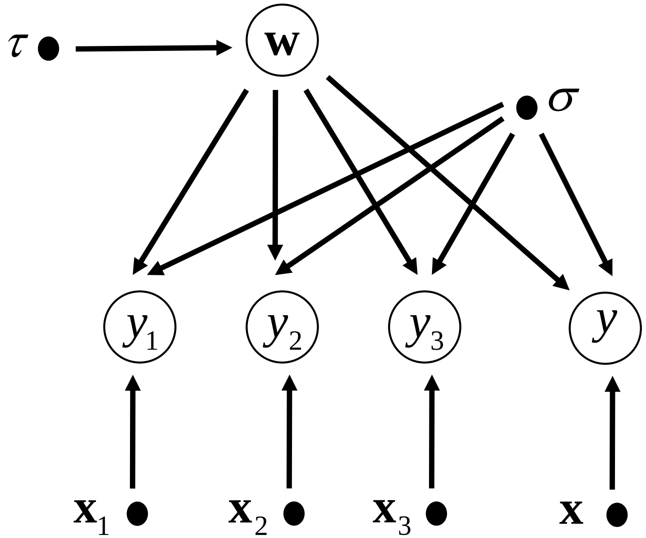
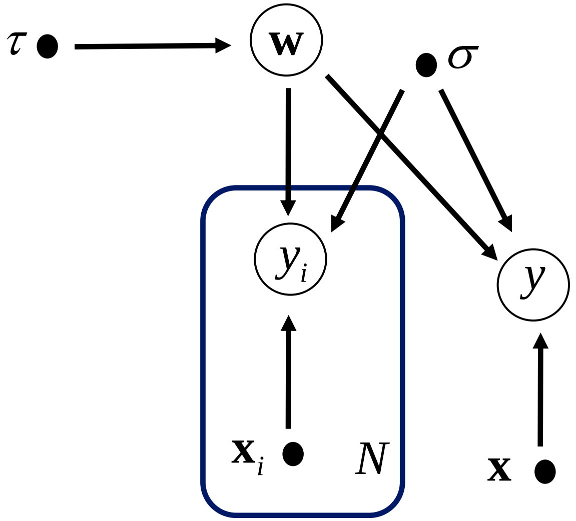
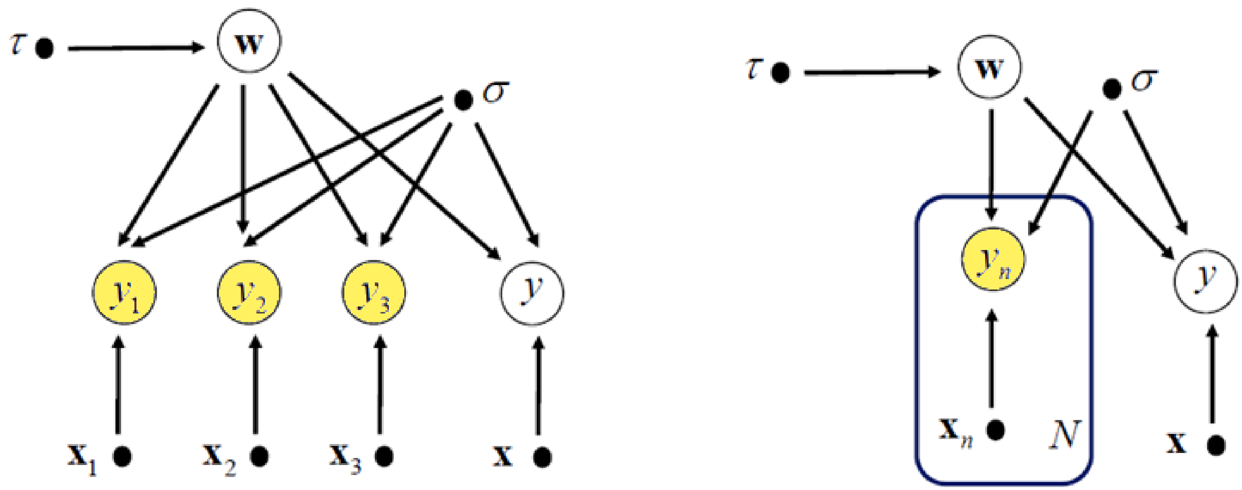


Plate notation



Bayesian Linear Regression as a Graphical Model



- Bayesian prediction
  - $y_* = \arg\max_y \, p(y \mid \mathbf{x}, \mathbf{y}, \mathbf{X}, \sigma^2, \tau^2)$
  - Inference problem: what is the most likely state of node y, given observed nodes $y_1, \ldots, y_N$?

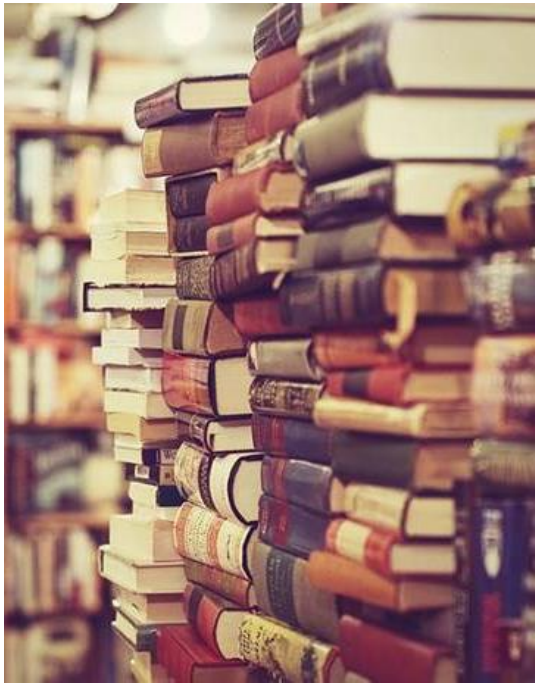
Agenda

- Graphical models: syntax and semantics.
- Inference in graphical models (exact, approximate)
- Graphical models in machine learning
  - Recap: Bayesian linear regression
  - Latent Dirichlet Allocation
  - Hidden Markov models

Topic Models

- „Topic models“: class of models for the (unsupervised) analysis of text corpora.

- Given a collection of text documents:
  - Discover the hidden themes (“topics”) that pervade the collection.
  - Describe topics by the words that most frequently appear.
  - Describe documents by the topics they are about.
  - Inferred annotations can be used to organize, summarize, and make predictions.



- Analysis is unsupervised, exploratory in nature.

Latent Dirichlet Allocation

- We will discuss *latent Dirichlet allocation* (LDA)
  - Well-funded probabilistic model.
  - Many practical applications.
  - Easily expressed as a graphical model.

Primer: Categorical and Dirichlet Distributions

- Let $X \in \{v_1, \dots, v_K\}$ denote a discrete random variable that takes on one of K values.

- The *categorical distribution* over X is given by

$$p(X = v_k) = \theta_k$$

where $\boldsymbol{\theta} = (\theta_1, \dots, \theta_K)^T \in R^K$ is a vector of probabilities, that is, $\sum_{k=1}^{K} \theta_k = 1$.

- The categorical distribution generalizes the Bernouilli distribution.

- We also write $X \sim Cat(X \mid \boldsymbol{\theta})$

- Example: rolling a fair dice, $\boldsymbol{\theta} = (1/6, \dots, 1/6)^T$.

Primer: Categorical and Dirichlet Distributions

- The Dirichlet distribution, given by

$$p(\boldsymbol{\theta}) = Dir(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) = \frac{\Gamma\left(\sum_{k=1}^{K} \alpha_k\right)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \theta^{\alpha_k - 1} \qquad \boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$$

  generalizes the Beta-distribution (identical to Beta for K=2).

- Reminder: $Beta(\theta \mid \alpha_1, \alpha_2) = \frac{\Gamma(\alpha_1 + \alpha_2)}{\Gamma(\alpha_1)\Gamma(\alpha_2)} \theta^{\alpha_1 - 1} (1-\theta)^{\alpha_2 - 1}$

- Conjugate prior: Dirichlet + categorical behaves like Beta + Bernoulli

Primer: Categorical and Dirichlet Distributions

- Recap: Beta is conjugate to Bernoulli
  - If
    $$\theta \sim Beta(\theta \mid \alpha_1, \alpha_2)$$
    $$X_i \sim \mathrm{Bern}(X \mid \theta)$$
  - Then $p(\theta \mid X_1, \ldots, X_N) = Beta(\theta \mid \alpha_1 + n_1, \alpha_2 + n_2)$

    Number of $X_i = 0$

    Number of $X_i = 1$

- Dirichlet is conjugate to categorical
  - If
    $$\boldsymbol{\theta} \sim Dir(\boldsymbol{\theta} \mid \boldsymbol{\alpha})$$
    $$X_i \sim Cat(X \mid \boldsymbol{\theta})$$
  - Then $p(\boldsymbol{\theta} \mid X_1, \ldots, X_N) = Dir(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + \mathbf{n})$

    $\mathbf{n} = (n_1, \ldots, n_K)^{\mathrm{T}}$

    $n_k =$ number of $X_i = v_k$

Primer: Symmetric Dirichlet Distribution

- Dirichlet distribution „smoothes" probability estimates towards the prior information in the vector $\boldsymbol{\alpha} \in R^K$.
- Example (*K=3)*:

$$\text{Prior: } \boldsymbol{\theta} \sim Dir(\boldsymbol{\theta} \mid (5,5,5))$$

$$\text{Observations: } \mathbf{n} = (3,0,2)$$

$$\text{MAP estimate: } \boldsymbol{\theta}^* = (0.41, 0.23, 0.36)$$

$$\text{Maximum likelihood estimate: } \boldsymbol{\theta}^* = (0.6, 0, 0.4)$$

- We often study *symmetric* Dirichlet distributions parameterized by a single $\alpha \in R$.

$$\boldsymbol{\theta} \sim Dir(\boldsymbol{\theta} \mid \alpha) = Dir(\boldsymbol{\theta} \mid (\alpha, \ldots, \alpha))$$

$$\alpha_1 = \alpha_2 = \ldots = \alpha_K = \alpha$$

Topic Models: Motivating Example

- Example: Analyze the following five sentences (=documents):

  (1) I like to eat broccoli and bananas.
  (2) I ate a banana and spinach smoothie for breakfast.
  (3) Chinchillas and kittens are cute.
  (4) My sister adopted a kitten yesterday.
  (5) Look at this cute hamster munching on a piece of broccoli.

- LDA: automatically discover topics that sentences contain.
- Possible answer:
  - Sentences (1) and (2): 100% Topic 1.
  - Sentences (3) and (4): 100% Topic 2.
  - Sentence 5: 60% Topic 1, 40% Topic 2.
  - Topic 1: 30% broccoli, 15% bananas, 10% breakfast, 10% munching, …
  - Topic 2: 20% chinchillas, 20% kittens, 20% cute, 15% hamster, …

Topic Models: Motivating Example

- Example: Analyze the following five sentences (=documents):

  (1) I like to eat broccoli and bananas.
  (2) I ate a banana and spinach smoothie for breakfast.
  (3) Chinchillas and kittens are cute.
  (4) My sister adopted a kitten yesterday.
  (5) Look at this cute hamster munching on a piece of broccoli.

- LDA: automatically discover topics that sentences contain.
- Possible answer:
  - Sentences (1) and (2): 100% Topic 1.
  - Sentences (3) and (4): 100% Topic 2.
  - Sentence 5: 60% Topic 1, 40% Topic 2.
  - Topic 1: 30% broccoli, 15% bananas, 10% breakfast, 10% munching, …
  - Topic 2: 20% chinchillas, 20% kittens, 20% cute, 15% hamster, …

Topic: food

Topic: cute animals

Latent Dirichlet Allocation

- Formalization: documents, words, and topics.
  - There are D documents, indexed by $d = 1, \ldots, D$.
  - There is a vocabulary of V different words.
  - Each document contains (up to) N words, denoted by $w_{d,1}, \ldots, w_{d,N}$.
  - There are K topics, indexed by $k = 1, \ldots, K$.

- Each topic k is described by a categorical distribution over words, represented by a parameter vector $\boldsymbol{\beta}_k \in R^V$.

$$\text{Vocabulary } = \{\text{bananas, broccoli, breakfast, chincillas, cute, hamster,} \ldots\}$$

$$\boldsymbol{\beta}_1 = (0.15 \quad 0.3 \quad 0.1 \quad 0.01 \quad 0.01 \quad 0.01 \quad \ldots)^{\mathrm{T}} \in R^V$$

$$\boldsymbol{\beta}_2 = (0.01 \quad 0.01 \quad 0.01 \quad 0.2 \quad 0.2 \quad 0.15 \quad \ldots)^{\mathrm{T}} \in R^V$$

Latent Dirichlet Allocation

- Formalization: documents, words, and topics.
  - There are D documents, indexed by $d = 1, \dots, D$.
  - There is a vocabulary of V different words.
  - Each document contains (up to) N words, denoted by $w_{d,1}, \dots, w_{d,N}$.
  - There are K topics, indexed by $k = 1, \dots, K$.

- Each document d is described by a categorical distribution over topics, represented by a parameter vector $\boldsymbol{\theta}_d \in R^K$.

$$\boldsymbol{\theta}_1 = (1.0 \quad 0.0) \in R^K$$

$$\boldsymbol{\theta}_2 = (1.0 \quad 0.0) \in R^K$$

$$\boldsymbol{\theta}_3 = (0.0 \quad 1.0) \in R^K$$

$$\boldsymbol{\theta}_4 = (0.0 \quad 1.0) \in R^K$$

$$\boldsymbol{\theta}_5 = (0.6 \quad 0.4) \in R^K$$

LDA: Generative Process

- Latent Dirichlet Allocation is a generative model, defining a generative process for the words appearing in the *D* documents.

- For topics $k = 1, \ldots, K$:
  - Draw the categorical distribution over words, $\boldsymbol{\beta}_k \sim Dir(\boldsymbol{\beta}|\eta)$.

Dirichlet prior

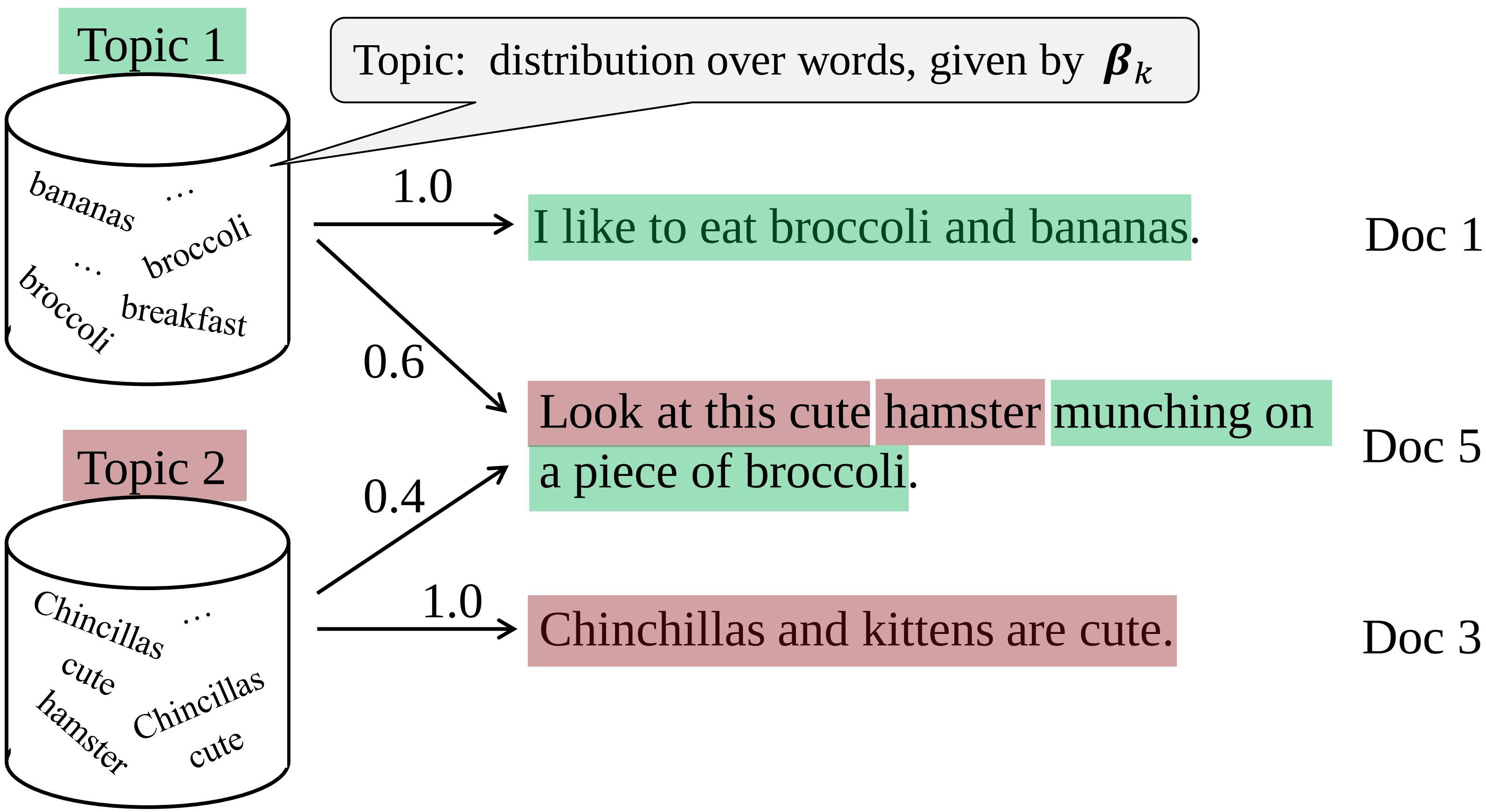
- For documents $d = 1, \ldots, D$:
  - Draw the categorial distribution over topics, $\boldsymbol{\theta}_d \sim Dir(\boldsymbol{\theta}|\alpha)$.
  - For each word $w_{d,1}, \ldots, w_{d,N}$:
    - Draw a topic $Z_{d,n} \sim Cat(Z|\boldsymbol{\theta}_d)$ for the word at position *n.*
    - Draw the word $w_{d,n} \sim Cat(w|\boldsymbol{\beta}_{Z_{d,n}})$

Dirichlet prior

Distribution over words in the selected topic.

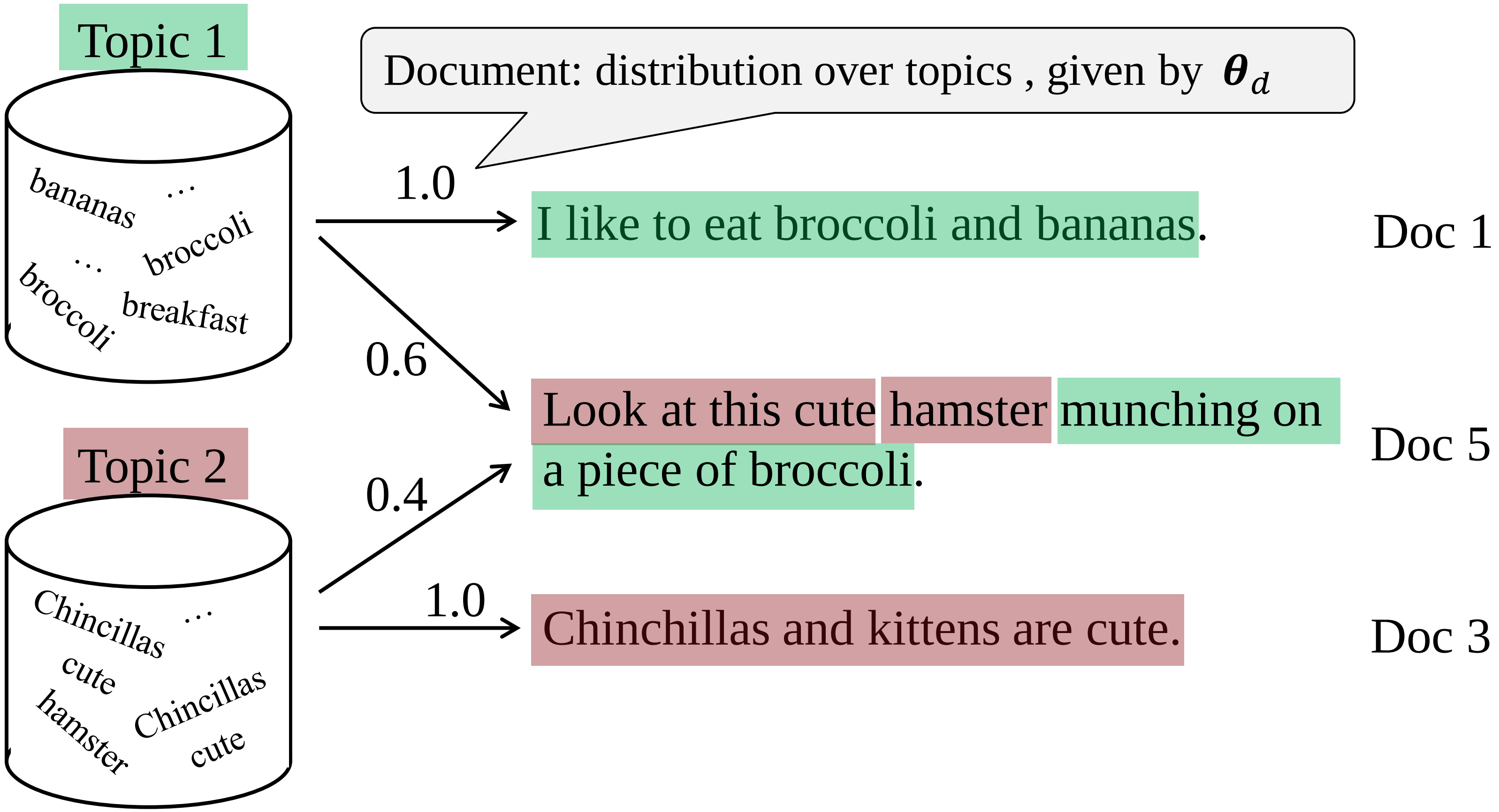
LDA: Generative Process

- Visualization of generative process for documents.



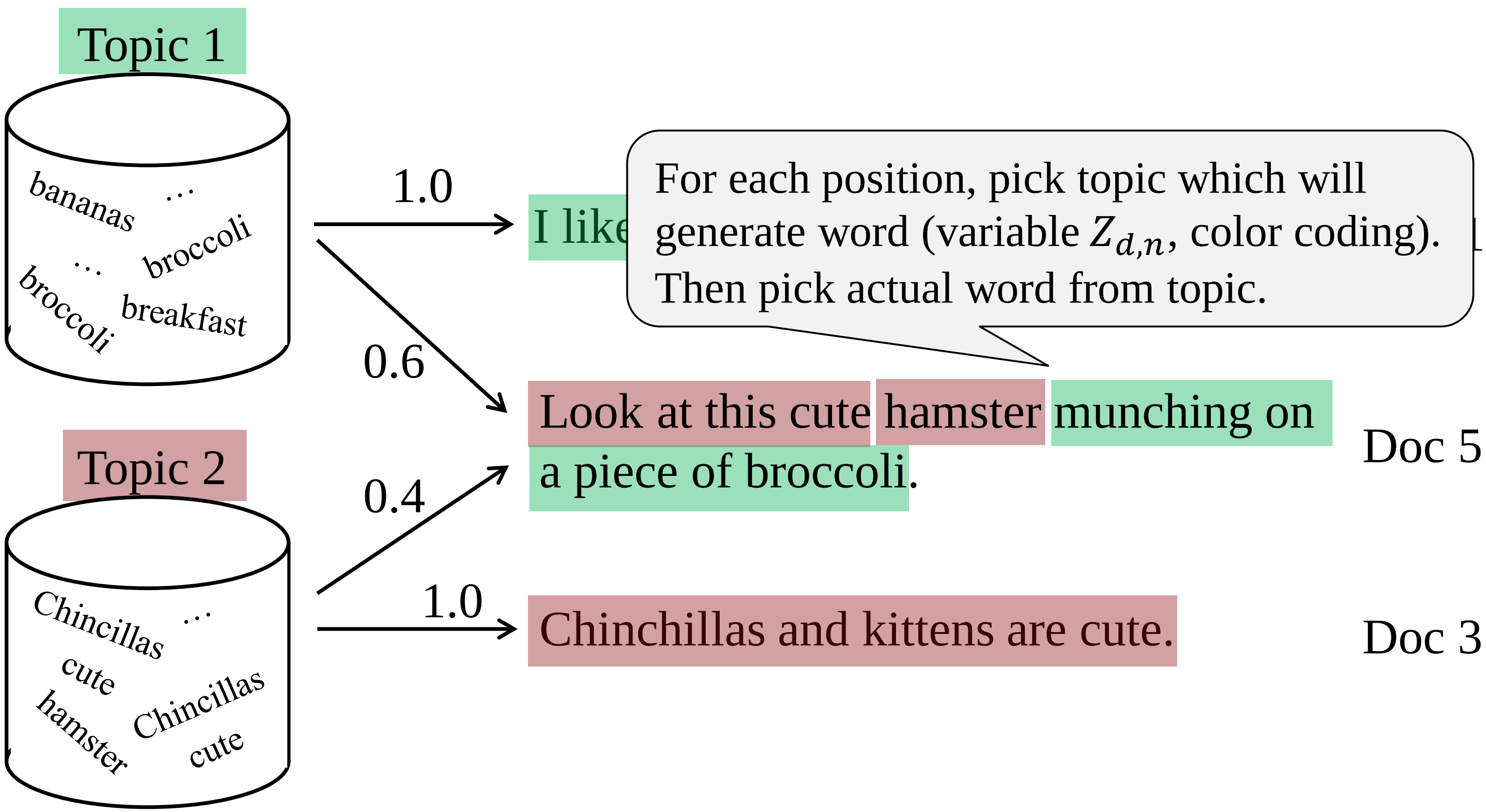
LDA: Generative Process

- Visualization of generative process for documents.



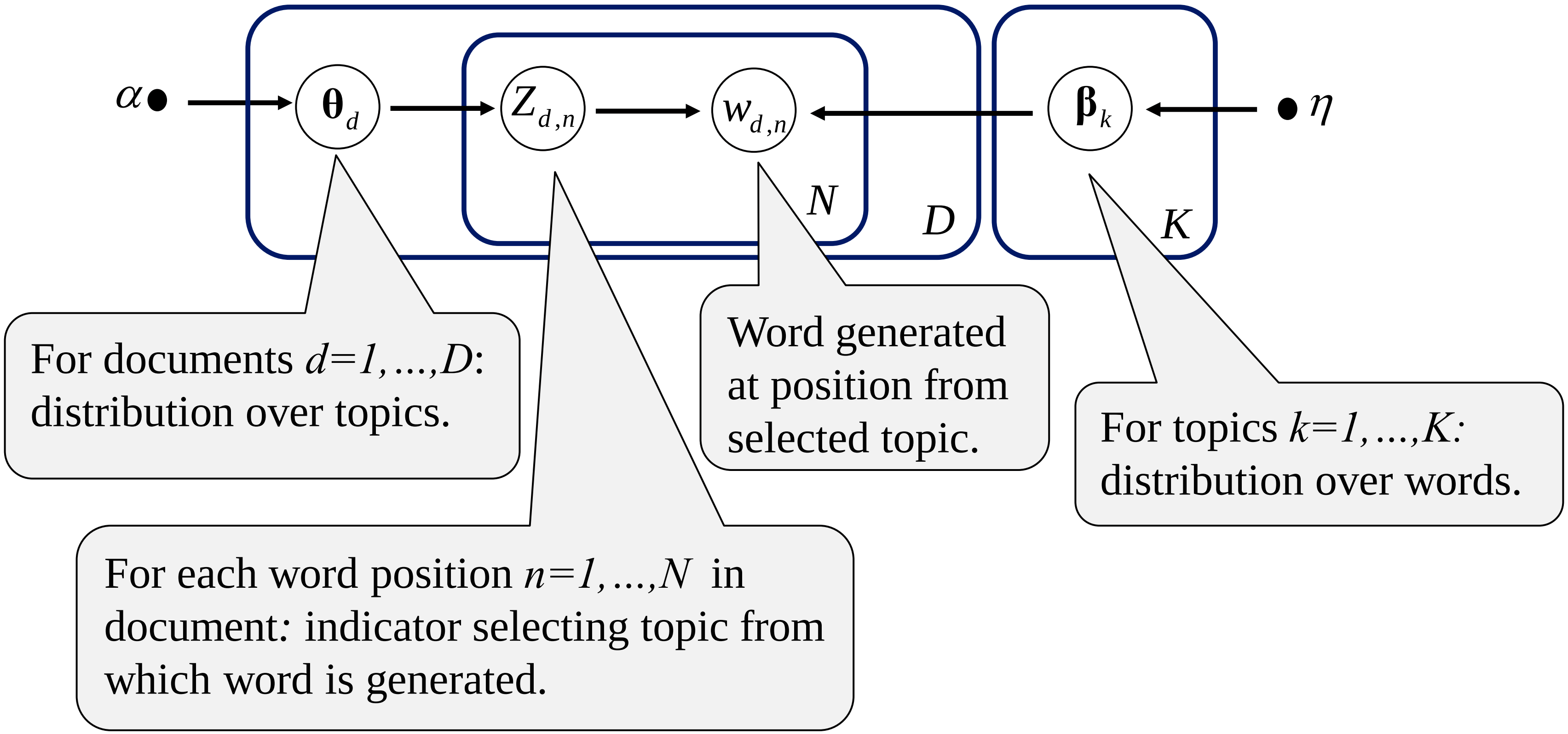
LDA: Generative Process

- Visualization of generative process for documents.



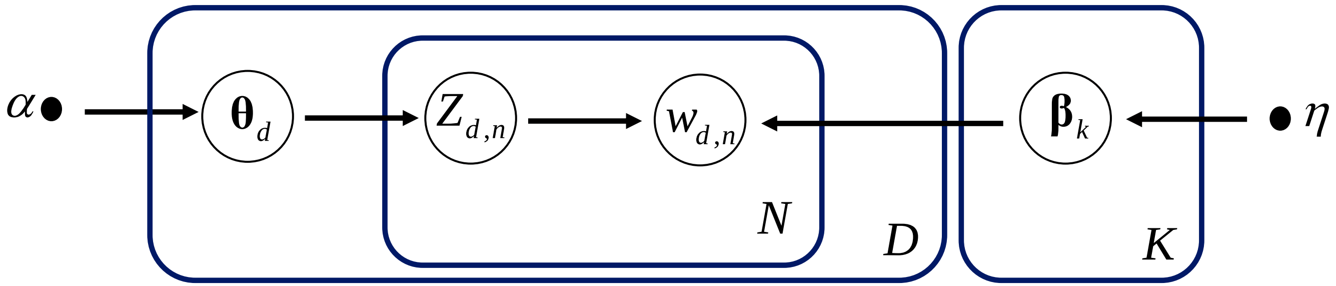
LDA: Graphical Model

- LDA as a graphical model (nested plate notation).

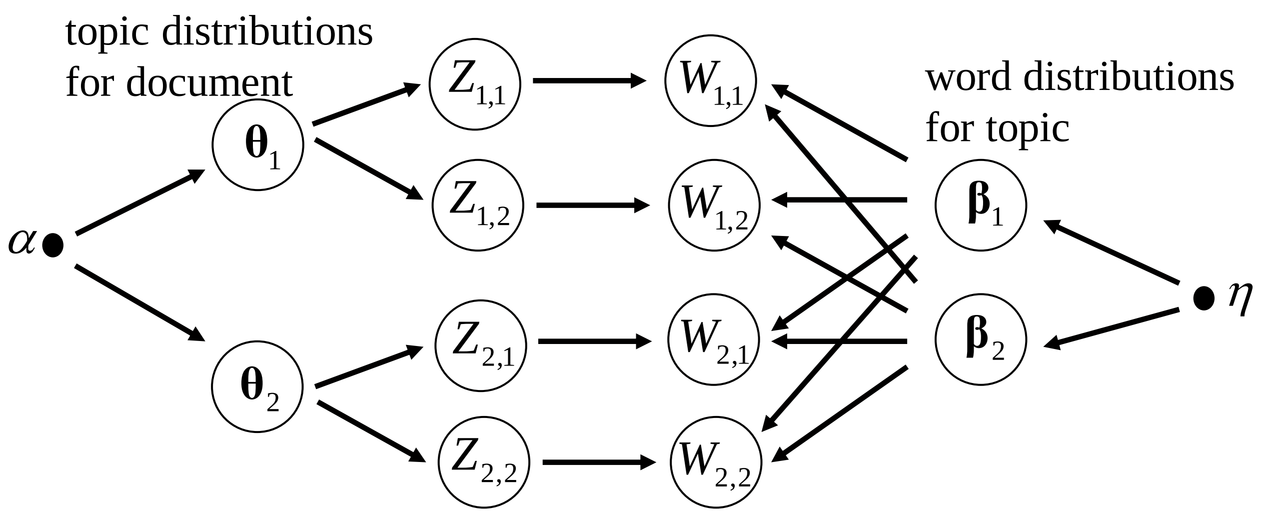


LDA: Graphical Model

- LDA as a graphical model (nested plate notation).

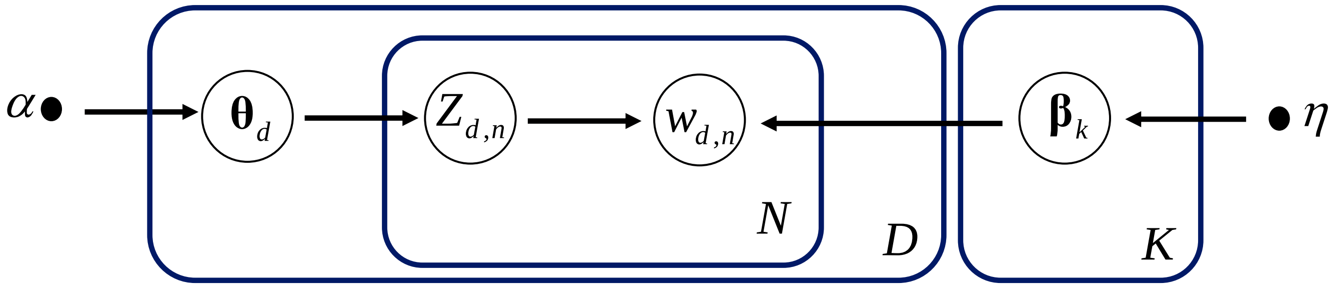


- Unrolled graphical model for small example (D=N=K=2):



LDA: Graphical Model

- LDA as a graphical model (nested plate notation).



- Collect all $\boldsymbol{\theta}_d$ in stacked vector $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_D)^{\mathrm{T}}$.
- Collect all $\boldsymbol{\beta}_k$ in stacked vector $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K)^{\mathrm{T}}$.
- Collect all $Z_{d,n}$ in matrix $\boldsymbol{Z} \in R^{D \times N}$.
- Collect all $w_{d,n}$ in matrix $\mathbf{W} \in R^{D \times N}$.
- Joint distribution:

$$p(\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{Z}, \mathbf{W} \mid \eta, \alpha) = \left(\prod_{k=1}^{K} p(\boldsymbol{\beta}_k \mid \eta) \right) \left(\prod_{d=1}^{D} p(\boldsymbol{\theta}_d \mid \alpha) \prod_{n=1}^{N} p(Z_{d,n} \mid \boldsymbol{\theta}_d) p(w_{d,n} \mid \boldsymbol{\beta}, Z_{d,n}) \right)$$

LDA For Text Analysis

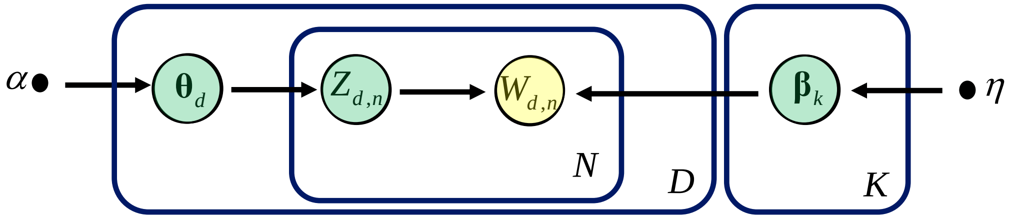
- Although LDA is a generative model, it is not actually good at generating natural language texts.

- LDA is a bag of words model:
  - To generate a word in a document, pick a random topic ($Z_{d,n}$ variable) and generate a word from that topic ($w_{d,n}$).
  - No sequential information: likely to generate texts like „breakfast broccoli bananas munching“ or „kitten cute hamster look“.

- **Practical application is in text analysis**:
  - Document collection is given. Discover hidden topics present in given document collection.
  - Annotate documents with topics.
  - This actually works well.

LDA For Text Analysis: Inference Problem

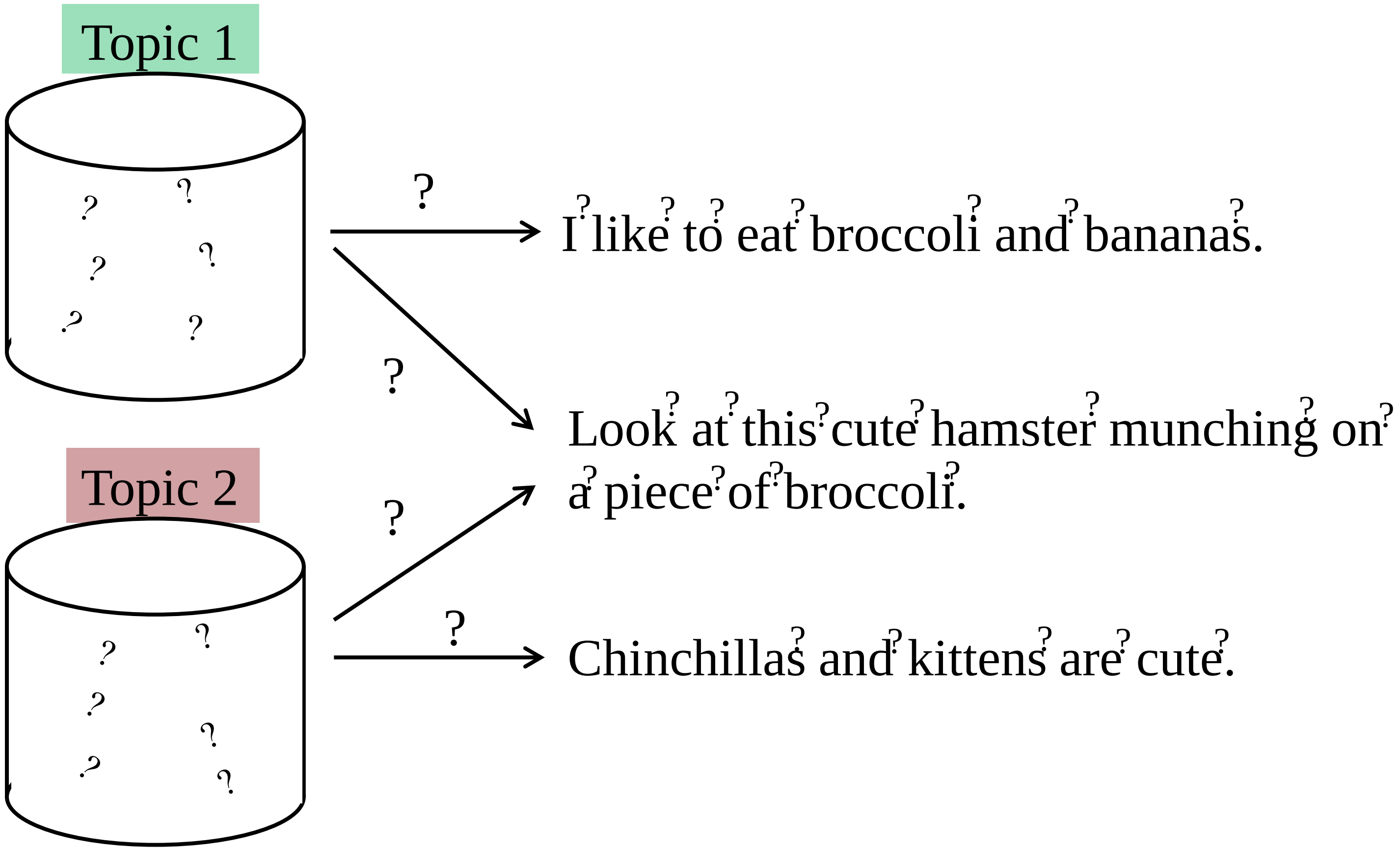
- Problem setting: given document collection, infer topic structure.
  - **Given**: collection of *D* documents with *N* words each, represented by variables $w_{d,n}$ $(\mathrm{d} = 1, \dots, \mathrm{D}, \mathrm{n} = 1, \dots \mathrm{N})$.
  - **Infer**:
    - topic distribution for each document (variables $\boldsymbol{\theta}_d$),
    - word distribution for each topic (variables $\boldsymbol{\beta}_k$)
    - which topic has generated a word ($Z_{d,n}$).
- Inference problem: compute $p(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{Z} | \boldsymbol{W})$.



- All variables except the word information $\boldsymbol{W}$ are *latent variables*, that is, they are never observed.

LDA For Text Analysis: Inference Problem

- Need to infer all other variables given word information.



LDA: Inference Algorithms

- Computing $p(\boldsymbol{\theta}, \boldsymbol{\beta}, \boldsymbol{Z} | \boldsymbol{W})$ means solving an inference problem in the graphical model representing LDA.

- Different approximate inference algorithms have been studied:
  - Gibbs sampler (Pritchard et al., 2000)
  - Collapsed Gibbs sampler (Griffiths & Steyvers, 2004)
  - Variational methods (Blei et al., 2003)
  - Expectation propagation (Minka and Lafferty, 2002)
  - …

- Sampling-based approaches very popular.

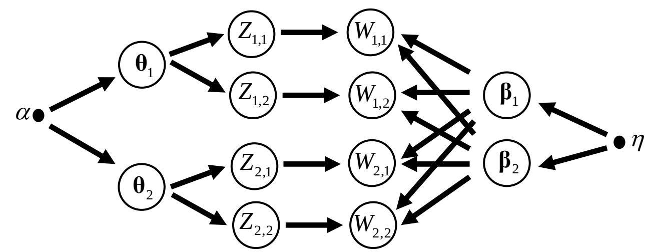
LDA: Gibbs Sampling

- Recap: Gibbs sampling resamples each variable given all other variables.
- Sample discrete variables $Z_{d,n}$, $w_{d,n}$: as discussed.
- Sample continuous variables $\boldsymbol{\theta}_d$ (topic distribution for document):

D-separation

$$p(\boldsymbol{\theta}_1 \mid \boldsymbol{\theta}_2, \boldsymbol{\beta}, \mathbf{W}, \mathbf{Z}, \eta, \alpha) = p(\boldsymbol{\theta}_1 \mid Z_{1,1}, Z_{1,2}, \alpha)$$

$$= Dir(\boldsymbol{\theta}_1 \mid (n_1 + \alpha, \ldots, n_K + \alpha))$$



Conjugate prior:
Posterior Dirichlet.

$$\text{with} \quad n_k = \sum_{n=1}^{N} [\![Z_{1,n} = k]\!]$$

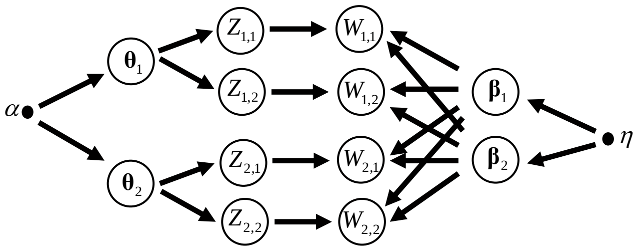
Count how often Topic *k* was choosen in Document 1.

LDA: Gibbs Sampling

- Sample continuous variables $\boldsymbol{\beta}_k$ (word distribution for topic):

D-separation

$$p(\boldsymbol{\beta}_1 \mid \boldsymbol{\theta}, \boldsymbol{\beta}_2, \mathbf{W}, \mathbf{Z}, \eta, \alpha) = p(\boldsymbol{\beta}_1 \mid \mathbf{W}, \mathbf{Z}, \eta)$$
$$= Dir(\boldsymbol{\beta}_1 \mid (m_1 + \eta, \ldots, m_V + \eta))$$



Conjugate prior:
Posterior Dirichlet.

with $m_v = \sum_{d=1}^{D} \sum_{n=1}^{N} [\![w_{d,n} = v \wedge Z_{d,n} = 1]\!]$

Count how often word v was generated from
Topic 1 (in any document).

- We can easily sample $\boldsymbol{\theta}_d$ and $\boldsymbol{\beta}_k$ from their respective Dirichlet posterior.

LDA: Sampling Inference

- We obtain samples $(\boldsymbol{\theta}^t, \boldsymbol{\beta}^t, \boldsymbol{Z}^t)$ for $t=1,\ldots,T$ from Gibbs sampler.

- Values for variables of interest are usually derived by averaging over samples (= expected value under distribution):

$$\boldsymbol{\theta} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{\theta}^t \qquad \boldsymbol{\beta} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{\beta}^t \qquad \mathbf{Z} = \frac{1}{T}\sum_{t=1}^{T}\mathbf{Z}^t$$

- In practice, full Gibbs sampling rarely used.

- The popular „collapsed“ Gibbs sampler exploits conjugacy of the priors over $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$ to integrate out these variables and only run the Gibbs sampler over the topic assignments $\boldsymbol{Z}$.

LDA: Example

- 100-Topic model trained on a corpus of 16.000 *Associated Press* documents.

Top words for 4 topics.

"Arts"	"Budgets"	"Children"	"Education"
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER

Document about these topics

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.

LDA: Summary and Remarks

- LDA: probabilistic model for discovering hidden topics in a document collection, and describing documents by these topics.

- Generative model:
  - A topic is a distribution over words.
  - A document is a distribution over topics.
  - For each position in document, pick topic and generate word.

- All variables except words are latent, inferred during inference.

- For simplicity, we have defined the model assuming that the number N of words is the same in each document
  - Straightforward to generalize to N differing over documents.
  - Generate N from model (Poisson distribution).

Agenda

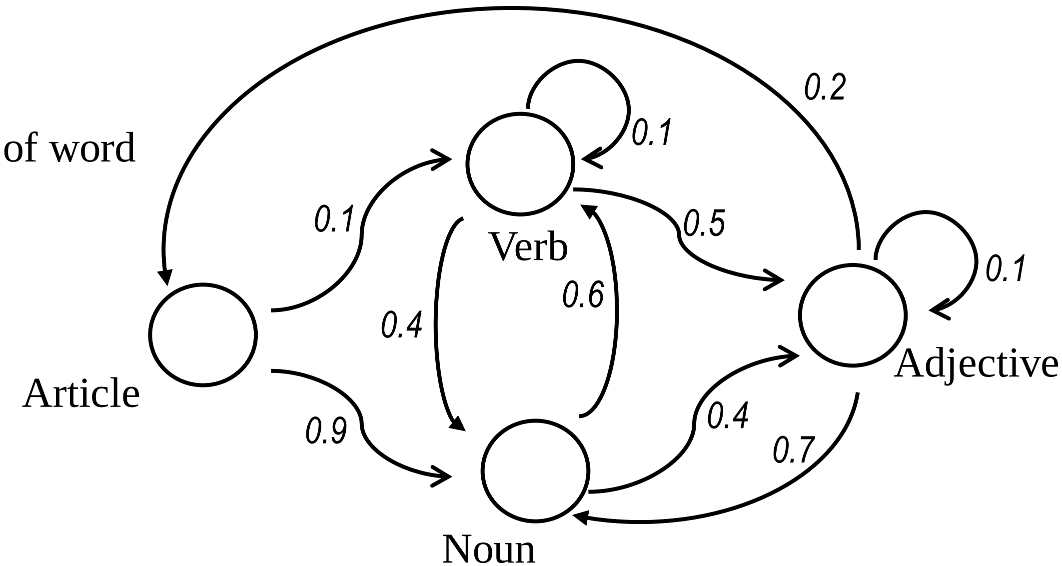
- Graphical models: syntax and semantics.
- Inference in graphical models (exact, approximate)
- Graphical models in machine learning
  - Recap: Bayesian linear regression
  - Latent Dirichlet allocation
  - Hidden Markov models

Hidden Markov Model: Probabilistic Automaton with Hidden States

- Hidden Markov model: probabilistic model for sequences.
- Temporal view: discrete time steps 1,….,T (= sequence elements).
- Models a probabilistic automaton that takes on a state from a finite set of states at each point in time.
- At each point in time, the automaton probabilistically changes into a novel state, based on the current state.

Example:
Model for natural language
Time steps = words in text
States = grammatical category of word

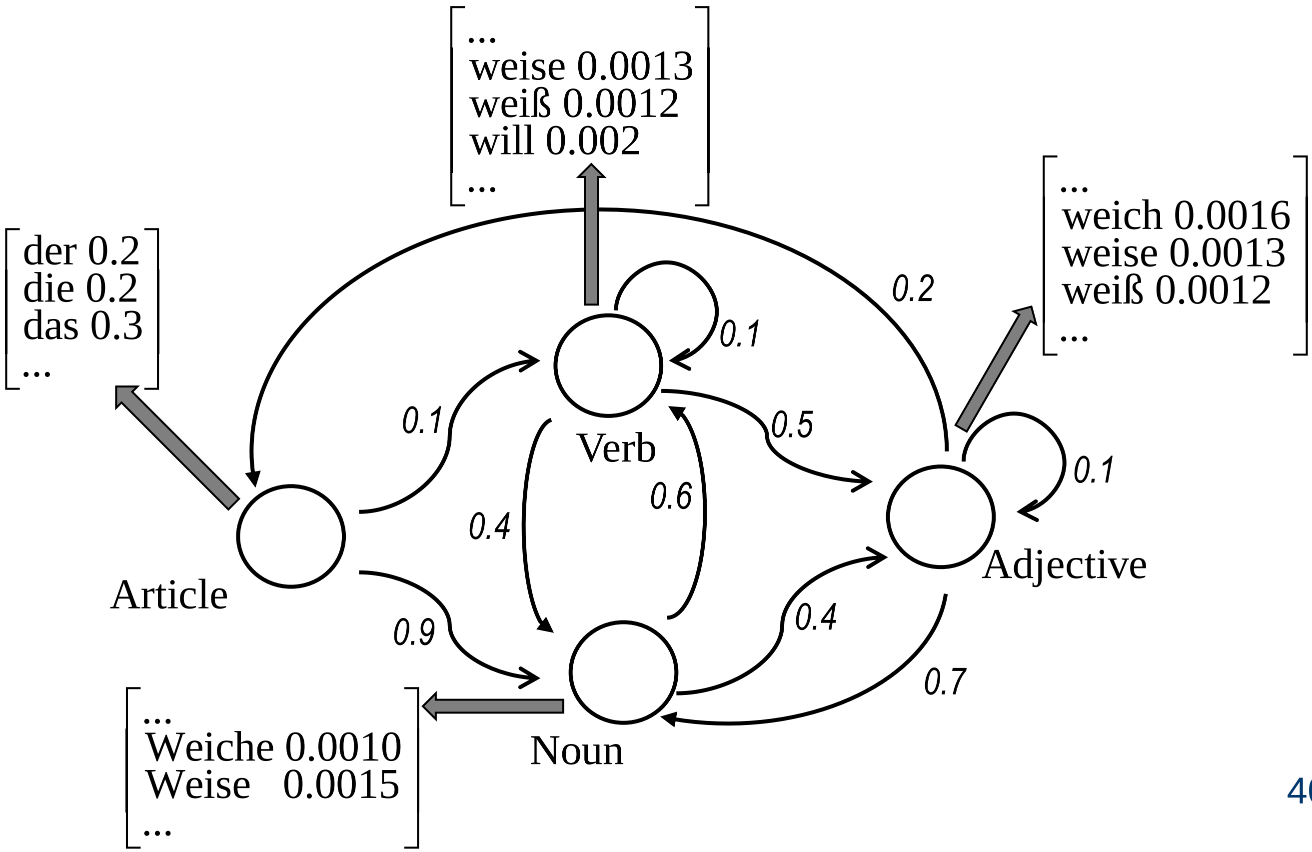
Probabilities for
state transitions



Hidden Markov Model: Probabilistic Automaton with Hidden States

- Sequence of states is not directly observable.
- Instead: states emit observations, these form the observable sequence.
- Distribution over possible observations depends on the current state.

Observations: actual word at position t



Formalization: States

- State at time t: random variable $q_t \in \{1,...,N\}$
- Automaton model:
  - Distribution over random initial state: $p(q_1)$
  - Distribution over next state given current state: $p(q_t \mid q_{t-1})$
- This results in joint distribution over all states:

$$p(q_1,....,q_T) = p(q_1)\prod_{t=2}^{T} p(q_t \mid q_{t-1})$$

- As a graphical model:

$$q_1 \longrightarrow q_2 \longrightarrow q_3 \longrightarrow q_4 \longrightarrow \cdots$$

- „Markov“-Assumption: $p(q_t \mid q_1,...,q_{t-1}) = p(q_t \mid q_{t-1})$

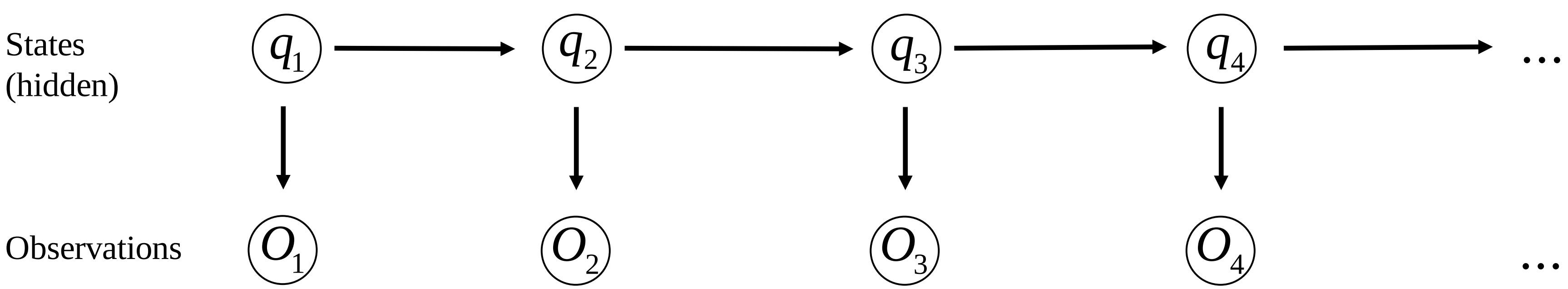
Formalization: Observations

- Observation at time t: random variable $O_t \in \{o_1, \ldots, o_M\}$.
- Automaton model: observation is generated depending on current state, from distribution $p(O_t \mid q_t)$.

- Joint distribution over all random variables:

$$p(q_1, \ldots, q_T, O_1, \ldots, O_T) = p(q_1)\, p(O_1 \mid q_1) \prod_{t=2}^{T} p(q_t \mid q_{t-1})\, p(O_t \mid q_t)$$

- As graphical model („Hidden Markov Model“ or „HMM“)



HMM Parameterization

- To define a hidden Markov model, we need to specify the following distributions:

$p(q_1)$ Distribution over initial states

$p(q_t \mid q_{t-1})$ Distribution over state transitions (independent of t)

$p(O_t \mid q_t)$ Distribution over observations (independent of t)

- Notation for the corresponding probability values:

Initial state probabilities: $p(q_1 = j) = \pi_j$ Vector $\pi \in \mathbb{R}^N$

Transition probabilities: $p(q_t = j \mid q_{t-1} = i) = a_{ij}$ Matrix $\mathrm{A} \in \mathbb{R}^{N \times N}$

Observation probabilities: $p(O_t = o_m \mid q_t = i) = b_i(o_m)$ Matrix $\mathrm{B} \in \mathbb{R}^{M \times N}$

- A hidden Markov mode is defined by the triple $\lambda = (A, B, \pi)$.

HMM Problem Settings

- Three basic problem settings for hidden Markov models:

- 1. Likelihood of an observation sequence: How likely is a sequence of observations $O_1, O_2, ..., O_T$ given a model λ ?

$$\text{Compute } p(O_1, ..., O_T \mid \lambda)$$

- 2. Most probable state, given observations:

$$\text{a)} \quad \text{Compute } \arg\max_{q_t} p(q_t \mid O_1, ..., O_T, \lambda)$$

$$\text{b)} \quad \text{Compute } \arg\max_{q_1, ..., q_T} p(q_1, ..., q_T \mid O_1, ..., O_T, \lambda)$$

- 3. Given several observation sequences, find a model that best explains the data:

$$\text{Compute } \arg\max_{\lambda} p(\{(O_1, ..., O_T), ...\} \mid \lambda)$$

1. Likelihood of an Observation Sequence

- Likelihood of an observation sequence: How likely is an observation sequence $O_1, O_2, \ldots, O_T$ given a model λ ?
- Sum rule:

$$p(O_1,\ldots,O_T \mid \lambda) = \sum_{q_1} \ldots \sum_{q_T} p(O_1,\ldots,O_T, q_1,\ldots,q_T \mid \lambda)$$

Exponential time

- Goal: polynomial-time algorithm.
- Solution:
  - Forward-Backward algorithm: dynamic programming.
  - Forward-Backward is special case of (general) message passing.
  - Also solves Problem 2.a)

Forward-Backward Algorithm

- Define auxiliary variables

$$\alpha_t(i) = p(O_1,\ldots,O_t, q_t = i \mid \lambda)$$ $N \cdot T$ variables overall

$$\beta_t(i) = p(O_{t+1},\ldots,O_T \mid q_t = i, \lambda)$$ $N \cdot T$ variables overall

- Theorem:

$$\alpha_1(i) = \pi_i b_i(O_1)$$

$$\alpha_{t+1}(i) = \left(\sum_{j=1}^{N} \alpha_t(j) a_{ji} \right) b_i(O_{t+1})$$

Allows recursive computation of all $\alpha_t(i)$ for $t = 1,\ldots,T$ and $\mathrm{j} = 1,\ldots,N$ in time $O(N^2 T)$

$$\beta_T(i) = 1$$

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$

Allows recursive computation of all $\beta_t(i)$ for $t = T,\ldots,1$ and $i = 1,\ldots,N$ in time $O(N^2 T)$

Forward-Backward Algorithm

- If we have computed all $\alpha_t(i)$ and $\beta_t(i)$:

Solution problem 1:

$$p(O_1,\dots,O_T \mid \lambda) = \sum_{i=1}^{N} p(O_1,\dots,O_T, q_T = i \mid \lambda)$$ sum rule

$$= \sum_{i=1}^{N} \alpha_T(i)$$

Total time: $O(N^2 T)$

Forward-Backward Algorithm

- If we have computed all $\alpha_t(i)$ and $\beta_t(i)$:

Solution Problem 2.a):

$$p(q_t = i \mid O_1,\ldots,O_T,\lambda) = \ldots$$

$$= \frac{\alpha_t(i)\beta_t(i)}{p(O_1,\ldots,O_T \mid \lambda)}$$

Total time: $O(N^2 T)$

Outlook

- Problem 2.b): Solution with Viterbi algorithm (same idea as in Forward-Backward)
- Problem 3: Solution with Baum-Welch algorithm
  - Instance of EM-Algorithm (see also Gaussian Mixture Models)
  - Makes use of Forward-Backward in E-step.

- Alternative approaches for modeling sequential data: discriminative models
  - HM-SVM
  - Conditional Random Fields

- More details in the lecture „speech technology“.

Applications HMMs

- Part-of-speech tagging in natural language texts
  - Hidden states correspond to grammatical categories (article, verb, noun, etc).
  - Observations are words in text.
  - Goal: assign grammatical categories to words (= find most likely hidden state sequence).

- Speech recognition: acoustic model
  - Hidden states are spoken words.
  - Observation is the acoustic signal.
  - Goal: reconstruct spoken words from acoustic signal.
  - Partially superseeded by deep learning.

- Bioinformatics
  - Localization or annotation of genes.
  - Usually extensions of the basic hidden Markov model.