Ranking-I

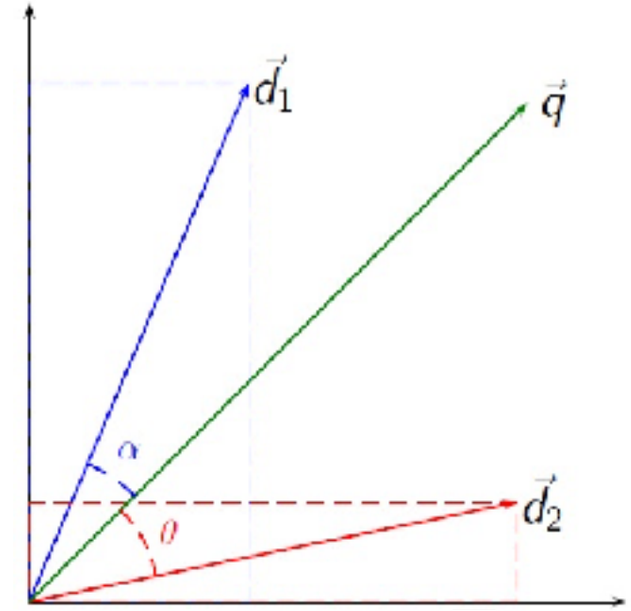
Probabilistic Interpretation and Language Models

Ranking in IR

- Ranking documents important for information overload, quickly finding documents which are **"relevant"** for the query
- Interpretations and Modelling of relevance
  - Geometric Interpretation — Vector Space Similarity, Okapi BM25
  - Probabilistic interpretation — **Topic for today**

Ranking in IR

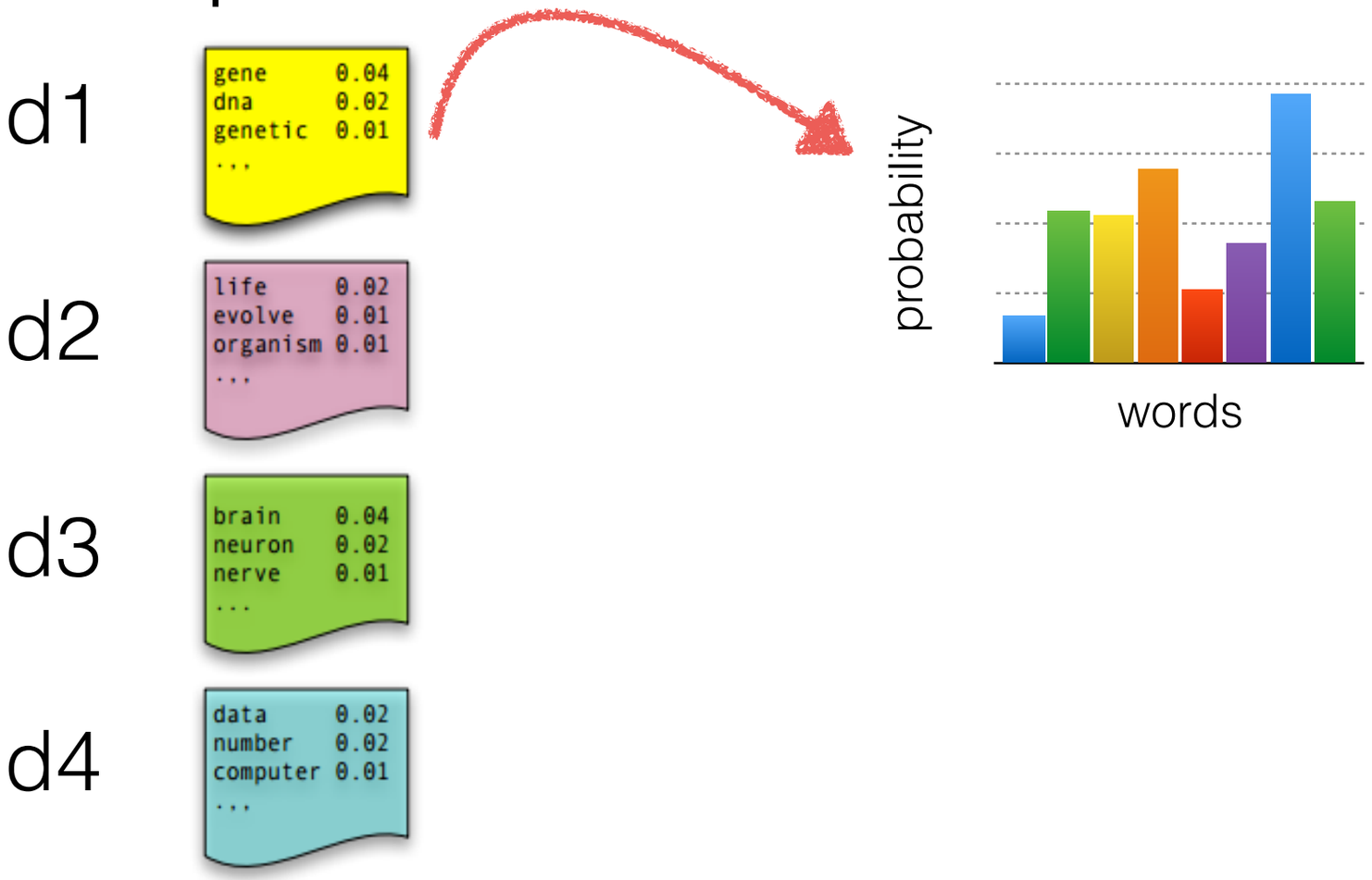
- Ranking documents important for information overload, quickly finding documents which are **"relevant"** for the query
- Interpretations and Modelling of relevance
  - Geometric Interpretation — Vector Space Similarity, Okapi BM25
  - Probabilistic interpretation — **Topic for today**



$$sim(d,q) = \frac{\sum_{i \in V} w_d(i).w_q(i)}{\sqrt{\sum_{i \in V} w_d(q_i)^2}\sqrt{\sum_{i \in V} w_q(i)^2}}$$

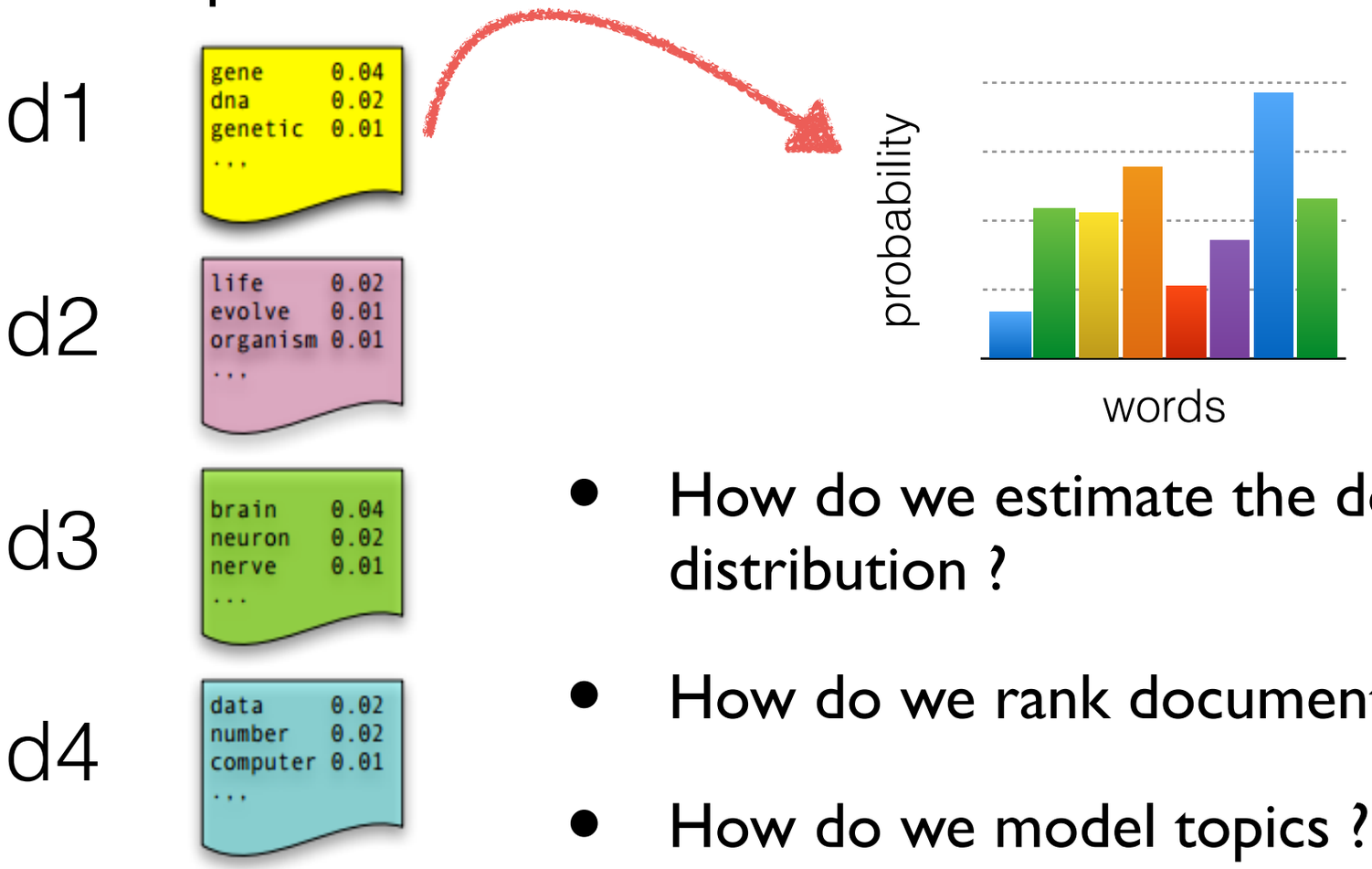
Probabilistic Interpretation

- Each document is a probability distribution over words
- Each document is just a collection of words or a "bag of words". Thus, the **order** of the words and the grammatical role of the words (subject, object, verbs, ...) are not considered in the model.
- Generative process



Probabilistic Interpretation

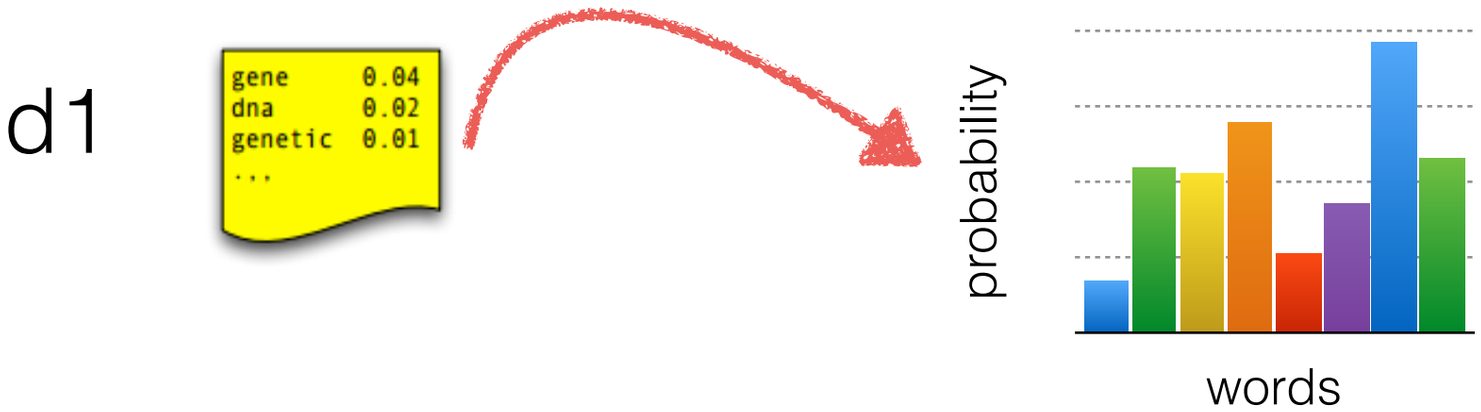
- Each document is a probability distribution over words
- Each document is just a collection of words or a "bag of words". Thus, the **order** of the words and the grammatical role of the words (subject, object, verbs, ...) are not considered in the model.
- Generative process



- How do we estimate the document distribution ?
- How do we rank documents given a query q ?
- How do we model topics ?

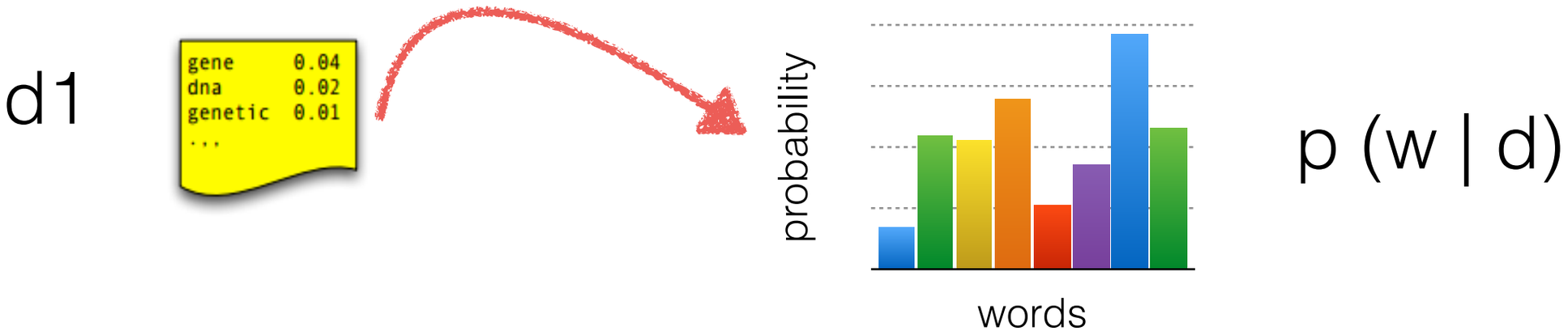
Probabilistic Interpretation

- What is the probability of the word "dna" is generated from document d1 ?



Probabilistic Interpretation

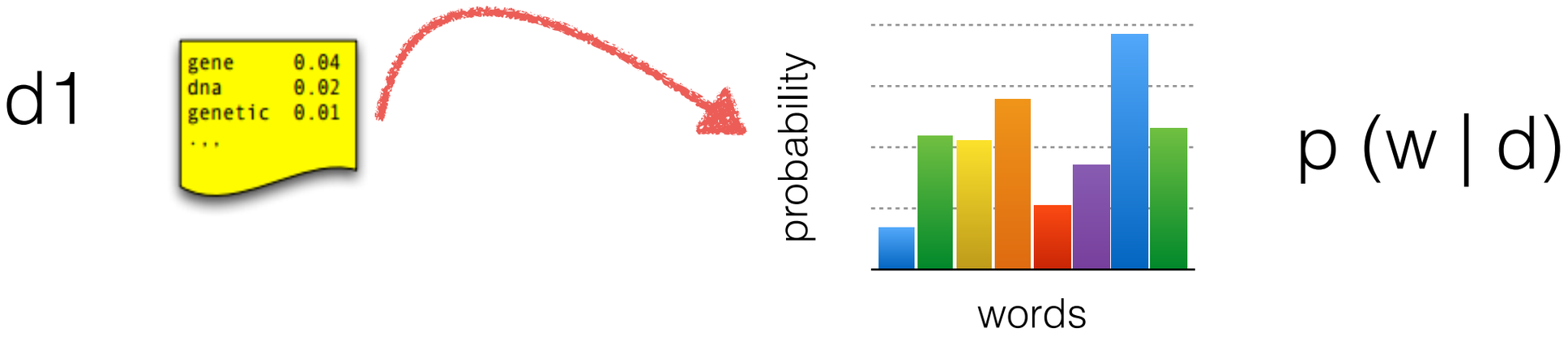
- What is the probability of the word "dna" is generated from document d1 ?



- What is the probability of the words "dna" and "gene" being generated from document d1 ? (assume independence between words)

Probabilistic Interpretation

- What is the probability of the word "dna" is generated from document d1 ?



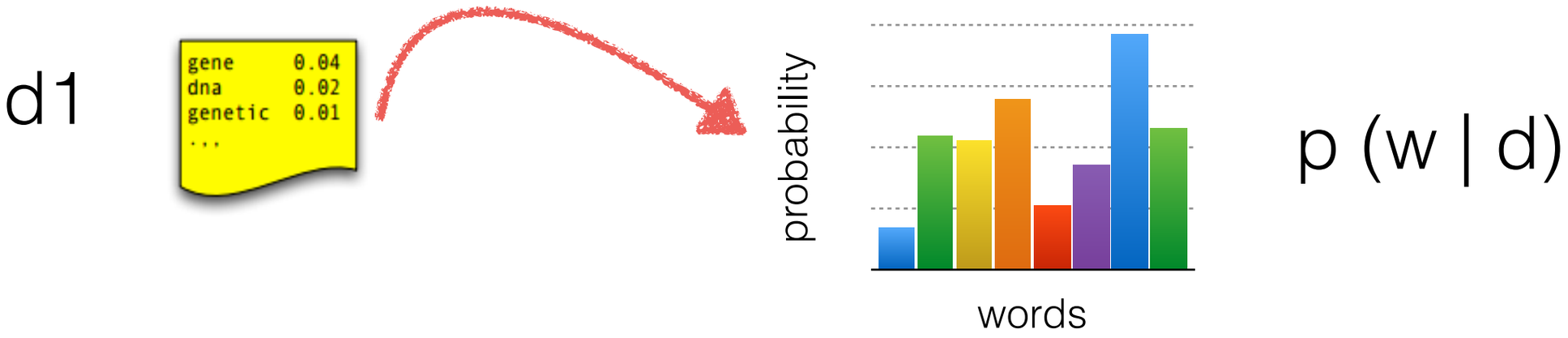
- What is the probability of the words "dna" and "gene" being generated from document d1 ? (assume independence between words)

$$p(\text{"gene"}, \text{"dna"} \mid d) = p(\text{"gene"} \mid d).\ p(\text{"dna"} \mid d)$$

A is **conditionally** independent to C given B → $P(A \mid B,C) = P(A \mid B)$

Probabilistic Interpretation

- What is the probability of the word "dna" is generated from document d1 ?



- What is the probability of the words "dna" and "gene" being generated from document d1 ? (assume independence between words)

$$p(\text{"gene"}, \text{"dna"} \mid d) = p(\text{"gene"} \mid d).\ p(\text{"dna"} \mid d)$$

A is **conditionally** independent to C given B → $P(A \mid B,C) = P(A \mid B)$

- For a query q = w1,w2 the probability that it came from d is:

$$p(w1,w2 \mid d) = p(w1 \mid d).\ p(w2 \mid d)$$

Parameter Estimation

Given **observed data** D

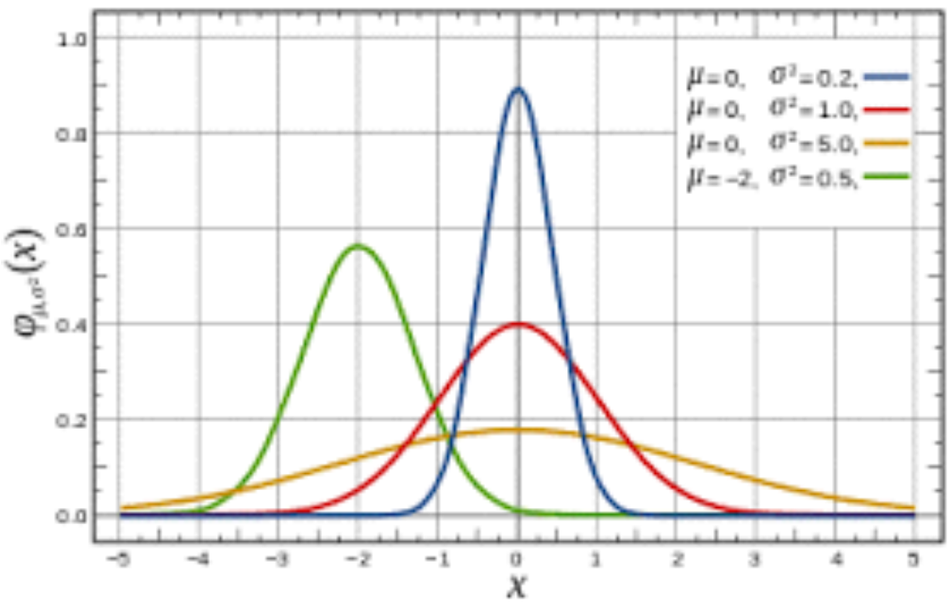
Generative model: Assume that you have a probability density func. p(x) with parameters θ which generates it

parameter estimation problem

What are the value of the parameters which best explain my data ?

Say you feel that D is generated from a Normal distribution

$$\theta = N(\mu, \sigma)$$



Maximum Likelihood Estimator (MLE)

Say you feel that D is generated from a Normal distribution

$$\theta = N(\mu, \sigma)$$

The **likelihood** of observing this independent and identically distributed sample is

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = p(x \mid \theta)$$

parameters to be estimated

observed data

MLE chooses $\hat{\theta}$ maximizes this likelihood of generating the observed data

Maximum Likelihood Estimator (MLE)

- Desired probability distribution is the one that makes the observed data "most likely," which means that one must seek the value of the parameter vector $\hat{\theta}$ that maximizes the likelihood function $\mathcal{L}(\hat{\theta}|x)$

Maximum Likelihood Estimator (MLE)

- Desired probability distribution is the one that makes the observed data "most likely," which means that one must seek the value of the parameter vector $\hat{\theta}$ that maximizes the likelihood function $\mathcal{L}(\hat{\theta}|x)$

$$p(x = (x_1, x_2, \ldots, x_n) \mid \theta) = p(x_1|\theta) \, . \, p(x_2|\theta) \ldots p(x_n|\theta)$$

Maximum Likelihood Estimator (MLE)

- Desired probability distribution is the one that makes the observed data "most likely," which means that one must seek the value of the parameter vector $\hat{\theta}$ that maximizes the likelihood function $\mathcal{L}(\hat{\theta}|x)$

$$p(x = (x_1, x_2, \ldots, x_n) \mid \theta) = p(x_1|\theta) \, . \, p(x_2|\theta) \ldots p(x_n|\theta)$$

- For computational convenience, the MLE estimate is obtained by maximizing the log-likelihood function $ln(\, \mathcal{L}(\hat{\theta}|x) \,)$

$$ln(\, \mathcal{L}(\hat{\theta}|x) \,) = \sum_{x_i} ln \; p(x_i \,|\, \theta)$$

- If log-likelihood is differentiable find $\hat{\theta}$ partial derivatives

Maximum Likelihood Estimator (MLE) - Gaussian

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = \prod_{x_i} p(x_i \mid \theta) \longrightarrow ln(\, \mathcal{L}(\hat{\theta}|x)\,) = \sum_{x_i} ln\; p(x_i \,|\, \theta)$$

likelihood

log likelihood

- Find maxima by using partial derivatives i.e. $\dfrac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \theta_i} = 0$

Maximum Likelihood Estimator (MLE) - Gaussian

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = \prod_{x_i} p(x_i \mid \theta) \longrightarrow ln(\, \mathcal{L}(\hat{\theta}|x)\,) = \sum_{x_i} ln\; p(x_i \,|\, \theta)$$

likelihood

log likelihood

- Find maxima by using partial derivatives i.e. $\dfrac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \theta_i} = 0$

- Example :

Maximum Likelihood Estimator (MLE) - Gaussian

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = \prod_{x_i} p(x_i \mid \theta) \longrightarrow ln(\, \mathcal{L}(\hat{\theta}|x)\,) = \sum_{x_i} ln\; p(x_i \,|\, \theta)$$

likelihood

log likelihood

- Find maxima by using partial derivatives i.e. $\dfrac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \theta_i} = 0$

- Example :

gaussian

$$p(x_i \mid (\mu, \sigma)) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

generative model

Maximum Likelihood Estimator (MLE) - Gaussian

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = \prod_{x_i} p(x_i \mid \theta) \longrightarrow ln(\, \mathcal{L}(\hat{\theta}|x)\,) = \sum_{x_i} ln\; p(x_i \,|\, \theta)$$

likelihood

log likelihood

- Find maxima by using partial derivatives i.e. $\dfrac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \theta_i} = 0$

- Example :

$$p(x_i \mid (\mu, \sigma)) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

gaussian

generative model

$$\frac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \mu} = 0$$

Maximum Likelihood Estimator (MLE) - Gaussian

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = \prod_{x_i} p(x_i \mid \theta) \longrightarrow ln(\, \mathcal{L}(\hat{\theta}|x)\,) = \sum_{x_i} ln\; p(x_i \,|\, \theta)$$

likelihood

log likelihood

- Find maxima by using partial derivatives i.e. $\dfrac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \theta_i} = 0$

- Example :

gaussian

$$p(x_i \mid (\mu, \sigma)) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

generative model

$$\frac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \mu} = 0 \qquad \frac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \sigma} = 0$$

Maximum Likelihood Estimator (MLE) - Gaussian

$$\mathcal{L}(\theta \mid x = (x_1, x_2, \ldots, x_n)) = \prod_{x_i} p(x_i \mid \theta) \longrightarrow ln(\, \mathcal{L}(\hat{\theta}|x)\,) = \sum_{x_i} ln\; p(x_i \,|\, \theta)$$

likelihood

log likelihood

- Find maxima by using partial derivatives i.e. $\dfrac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \theta_i} = 0$

- Example :

gaussian

$$p(x_i \mid (\mu, \sigma)) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}$$

generative model

$$\frac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \mu} = 0 \qquad \frac{\partial\, ln\, \mathcal{L}(\theta \mid x)}{\partial\, \sigma} = 0$$

- What are the estimated parameters of the gaussian ?

Multinomial Distribution for Text

- Lets try to find a distribution for text, say a document
- You are given a document D with
  - tf ("game") = 5, tf ("of") = 15, tf ("thrones") = 5, tf ("arya") = 3, tf ("stark") = 4, ….
  - E = {all words in D} = {"game","thrones","stark",….}

- Multinomial distribution **best encodes** the generative process of text

multinomial distribution

$$\binom{n}{tf_1, tf_2, \ldots, tf_{|E|}} p(x_1)^{tf_1} \ldots . p(x_{|E|})^{tf_{|E|}}$$

term frequency of the word

prob. of the word occurrence

Language Model for a Document

- What are the parameters to be estimated and how can we estimate them ?

total words in Doc

$$\binom{|D|}{tf_1, tf_2, \ldots, tf_{|V|}} \; p(x_1)^{tf_1} \; \ldots . \; p(x_{|V|})^{tf_{|V|}}$$

multinomial distribution

term frequency of the word

- Now we have a Model or P (w|d) for each d

Language Model for a Document

- What are the parameters to be estimated and how can we estimate them ?

total words in Doc

observed values

$$\binom{|D|}{tf_1, tf_2, \ldots, tf_{|V|}} p(x_1)^{tf_1} \ldots . p(x_{|V|})^{tf_{|V|}}$$

multinomial distribution

term frequency of the word

- Now we have a Model or P (w|d) for each d

Language Model for a Document

- What are the parameters to be estimated and how can we estimate them ?

$$\binom{|D|}{tf_1, tf_2, \ldots, tf_{|V|}} \; p(x_1)^{tf_1} \; \ldots . \; p(x_{|V|})^{tf_{|V|}}$$

total words in Doc

observed values

multinomial distribution

term frequency of the word

parameter to be estimated

- Now we have a Model or P (w|d) for each d

Language Model for a Document

- What are the parameters to be estimated and how can we estimate them ?

$$\binom{|D|}{tf_1, tf_2, \ldots, tf_{|V|}} \; p(x_1)^{tf_1} \; \ldots . \; p(x_{|V|})^{tf_{|V|}}$$

total words in Doc

observed values

multinomial distribution

term frequency of the word

parameter to be estimated

- The MLE for each of the word prob. p(x) is the most natural estimate

$$p(x_i) \; = \; \frac{tf_1}{|D|}$$

- Now we have a Model or P (w|d) for each d

Language Models in IR

- Unigram Model typically used in IR
  - P(w1,w2,w3,w4, w5,..| d) = P(w1|d). P(w2|d)…
- Build a language model for each document D — { P (w_i | D) }
- **Ranking**:
  - Query Likelihood: Given a query Q, find the relevance of D (rank acc to P (D|Q))

- KL-Divergence Model:
  - Build language model for the query P (w | Q)
  - Rank acc. to **KL (D || Q)** compares two distributions

Ranking using Query Likelihood

- Query Likelihood: Given a query Q, find the relevance of D (rank acc to P (D|Q))

$$P(D|Q) = \frac{P(Q \mid D).P(D)}{P(Q)}$$

$$P(D|Q) \propto P(Q \mid D).P(D)$$

Ranking using Query Likelihood

- Query Likelihood: Given a query Q, find the relevance of D (rank acc to P (D|Q))

$$P(D|Q) = \frac{P(Q \mid D).P(D)}{P(Q)}$$

constant for all docs

$$P(D|Q) \propto P(Q \mid D).P(D)$$

Ranking using Query Likelihood

- Query Likelihood: Given a query Q, find the relevance of D (rank acc to P (D|Q))

Authority of the docs. determined by Page Rank etc.

$$P(D|Q) = \frac{P(Q \mid D).P(D)}{P(Q)}$$

constant for all docs

$$P(D|Q) \propto P(Q \mid D).P(D)$$

Ranking using Query Likelihood

- Query Likelihood: Given a query Q, find the relevance of D (rank acc to P (D|Q))

Authority of the docs. determined by Page Rank etc.

$$P(D|Q) = \frac{P(Q \mid D).P(D)}{P(Q)}$$

constant for all docs

$$P(D|Q) \propto P(Q \mid D).P(D)$$

Assuming all documents are equally likely

$$P(D|Q) \propto P(Q \mid D)$$

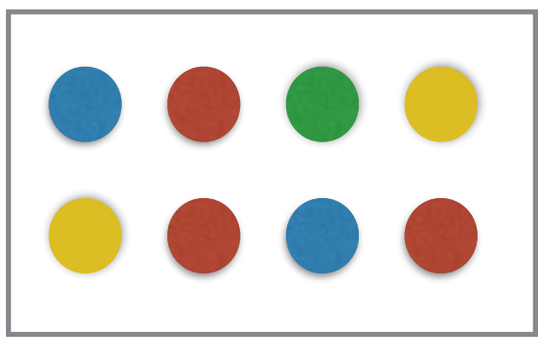
$$P(D|Q) \propto \prod_{w_i \in Q} P(w_i|D)$$

unigram language model

Language Model for a Document

- Unigram Language Model provides a probabilistic model for representing text in Information retrieval

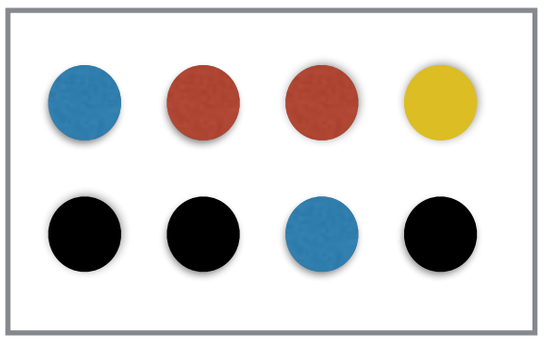
D_1



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/4$

$p(\bullet | D_1) = 1/8$ $p(\bullet | D_1) = 3/8$

D_2



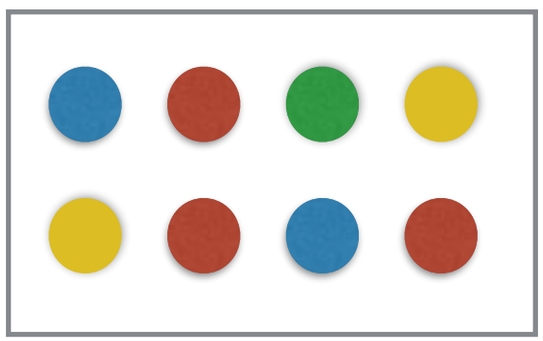
$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/8$

$p(\bullet | D_1) = 3/8$ $p(\bullet | D_1) = 1/4$

Language Model for a Document

- Unigram Language Model provides a probabilistic model for representing text in Information retrieval

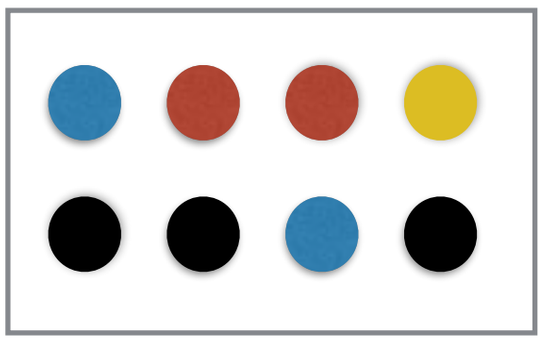
D_1



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/4$

$p(\bullet | D_1) = 1/8$ $p(\bullet | D_1) = 3/8$

D_2



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/8$

$p(\bullet | D_1) = 3/8$ $p(\bullet | D_1) = 1/4$

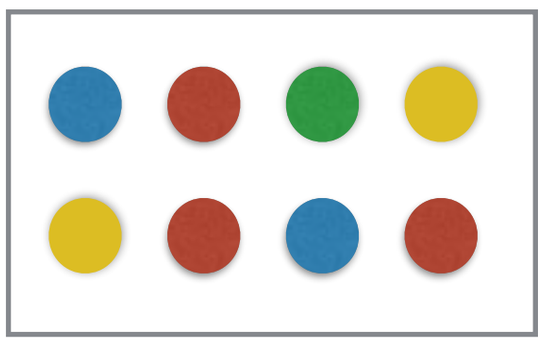
query

$p(\bullet\bullet | D_1) = 3/32$

Language Model for a Document

- Unigram Language Model provides a probabilistic model for representing text in Information retrieval

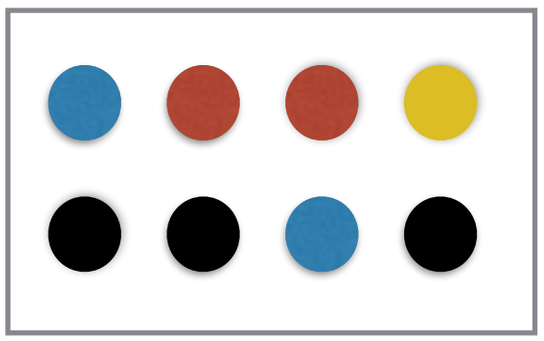
D_1



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/4$

$p(\bullet | D_1) = 1/8$ $p(\bullet | D_1) = 3/8$

D_2



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/8$

$p(\bullet | D_1) = 3/8$ $p(\bullet | D_1) = 1/4$

query

$p(\bullet\bullet | D_1) = 3/32$

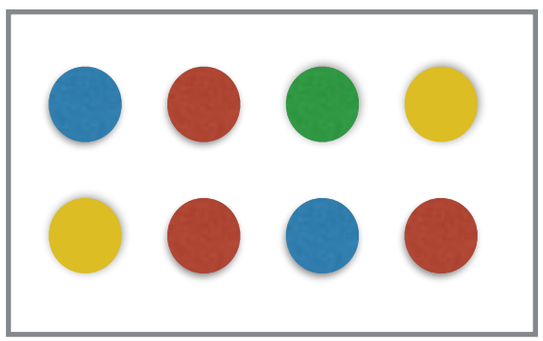
query

$p(\bullet\bullet | D_2) = 1/32$

Language Model for a Document

- Unigram Language Model provides a probabilistic model for representing text in Information retrieval

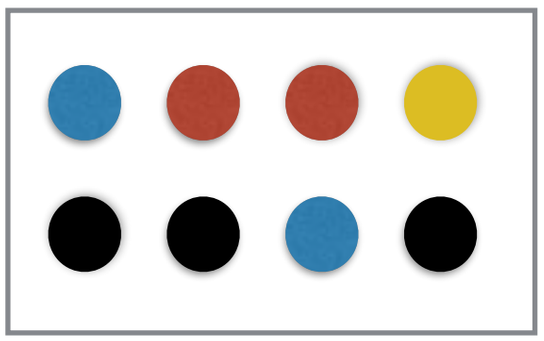
D_1



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/4$

$p(\bullet | D_1) = 1/8$ $p(\bullet | D_1) = 3/8$

D_2



$p(\bullet | D_1) = 1/4$ $p(\bullet | D_1) = 1/8$

$p(\bullet | D_1) = 3/8$ $p(\bullet | D_1) = 1/4$

query

$p(\bullet\bullet | D_1) = 3/32$

query

$p(\bullet\bullet | D_2) = 1/32$ $D_1 > D_2$

Example - Language Model + MLE

- Unigram Language Model provides a probabilistic model for representing text

Model M_1		Model M_2	
the	0.2	the	0.15
a	0.1	a	0.12
frog	0.01	frog	0.0002
toad	0.01	toad	0.0001
said	0.03	said	0.03
likes	0.02	likes	0.04
that	0.04	that	0.04
dog	0.005	dog	0.01
cat	0.003	cat	0.015
monkey	0.001	monkey	0.002
...	...	...	...

s	frog	said	that	toad	likes	that	dog
M_1	0.01	0.03	0.04	0.01	0.02	0.04	0.005
M_2	0.0002	0.03	0.04	0.0001	0.04	0.04	0.01

$$P(s|M_1) = 0.0000000000048$$
$$P(s|M_2) = 0.00000000000000384$$

- The MLE for each of the word prob. p(x) is the most natural estimate

$$p(x_i) = \frac{tf_1}{|D|}$$

Zero Probability Problem

- What if some of the queried terms are absent in the document ?
- MLE based estimation results in a zero probability for query generation

Model M_1		Model M_2	
the	0.2	the	0.15
a	0.1	a	0.12
frog	0.01	frog	0.0002
toad	0.01	toad	0.0001
said	0.03	said	0.03
likes	0.02	likes	0.04
that	0.04	that	0.04
dog	0.005	dog	0.01
cat	0.003	cat	0.015
monkey	0.001	monkey	0.002
…	…	…	…

- P("frog","ape" | M_1) = 0.01x **0**
- P("frog","ape" | M_2) = 0.00021x **0**

- Need to smooth the probability estimates for terms to avoid zero probabilities
- Take the prob. mass from each term and redistribute among missing terms

Smoothing Methods

- **Jelinek-Mercer Smoothing :** Linear combination of document and corpus statistics to estimate term probabilities

$$P(Q|D) = \prod_{w_i \in Q} \underbrace{\lambda.P(w_i|D)}_{\text{doc. contrib.}} + \underbrace{(1-\lambda).P(w_i|C)}_{\text{corpus contrib.}}$$

- Collection frequency: fraction of occurence of term in the entire collection
- Document frequency: fraction of document occurence of term in the entire collection

Smoothing Methods

- **Jelinek-Mercer Smoothing :** Linear combination of document and corpus statistics to estimate term probabilities

$$P(Q|D) = \prod_{w_i \in Q} \lambda.P(w_i|D) + (1-\lambda).P(w_i|C)$$

doc. contrib.

corpus contrib.

collection freq. or document fréquency

- Collection frequency: fraction of occurence of term in the entire collection
- Document frequency: fraction of document occurence of term in the entire collection

Smoothing Methods

- **Jelinek-Mercer Smoothing :** Linear combination of document and corpus statistics to estimate term probabilities

$$P(Q|D) = \prod_{w_i \in Q} \lambda.P(w_i|D) + (1-\lambda).P(w_i|C)$$

doc. contrib.

corpus contrib.

param. regulates contribution

collection freq. or document fréquency

- Collection frequency: fraction of occurence of term in the entire collection
- Document frequency: fraction of document occurence of term in the entire collection

Smoothing Methods

- Smoothing with **Dirichlet Prior:**

term freq of word in Doc

$$P(Q|D) = \prod_{w_i \in Q} \frac{tf(w_i; D) + \mu.P(w_i|C)}{|D| + \mu}$$

- Takes the corpus distribution as a prior to estimating the prob. for terms
- works well for short queries

Smoothing Methods

- Smoothing with **Dirichlet Prior:**

$$P(Q|D) = \prod_{w_i \in Q} \frac{tf(w_i; D) + \mu.P(w_i|C)}{|D| + \mu}$$

term freq of word in Doc

dirichlet prior

- Takes the corpus distribution as a prior to estimating the prob. for terms
- works well for short queries

Smoothing Methods

- Smoothing with **Dirichlet Prior:**

$$P(Q|D) = \prod_{w_i \in Q} \frac{tf(w_i; D) + \mu.P(w_i|C)}{|D| + \mu}$$

term freq of word in Doc

collection freq. or document fréquency

dirichlet prior

- Takes the corpus distribution as a prior to estimating the prob. for terms
- works well for short queries

Topic Models

Vocabulary Mismatch Problem

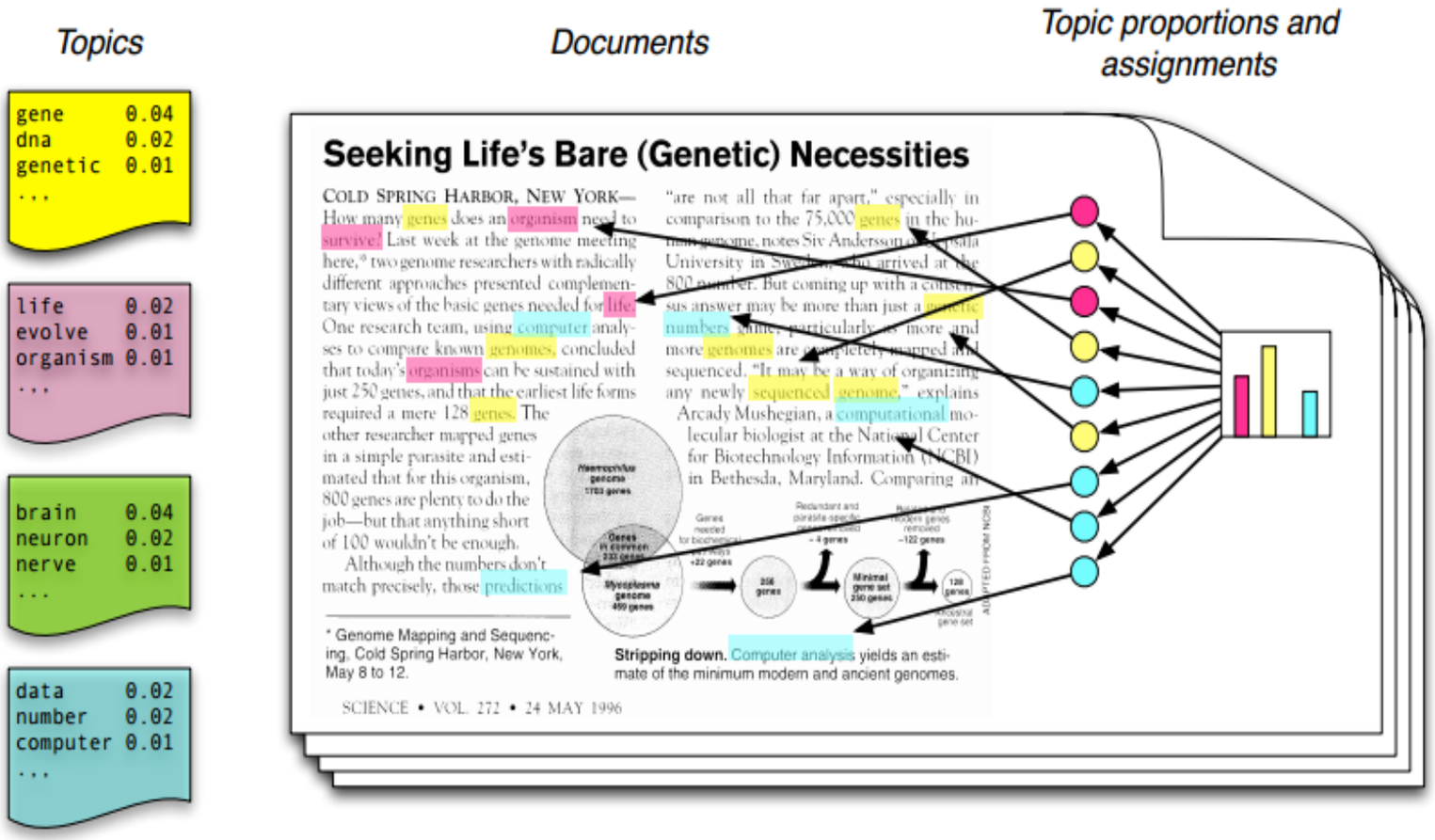
- One concept can be represented by several different words!
- Two documents might not contain similar terms (for instance due to writing styles) but refer to a single concept.
- Queries can contain words not present in a document and still be very relevant to that document!

Topic model (Probabilistic Latent Semantic Indexing — PLSI)

- Given a set documents D
- A set of topics, classes, concepts {z1, z2,….,zk}
- A set of words {w1, w2,…}
- How do we find topics (word distributions) and documents (topic distribution) ?

Generative Process

The generative process:



- Each **document** is a mixture of corpus-wide topics
- Each **concept** is distribution over words
- Each **word** is drawn from one of these topics
- We only observe the words within the documents and the other structures are **hidden variables**.

PLSI representation

What is the probability of seeing a word w **and** a document d ?

$$p(d,w) = p(w|d).p(d)$$

PLSI representation

What is the probability of seeing a word w **and** a document d ?

$$p(d,w) = p(w|d).p(d)$$

- Select a document with probability $P(d)$
- Pick a latent class z with probability $P(z|d;\theta)$
- Generate a word w with probability $P(w|z;\pi)$

PLSI representation

What is the probability of seeing a word w **and** a document d ?

$$p(d,w) = p(w|d).p(d)$$

- Select a document with probability $P(d)$
- Pick a latent class z with probability $P(z|d;\theta)$
- Generate a word w with probability $P(w|z;\pi)$

$$\hat{P}_{LSA}(w|d) = \sum_{z \in Z} P(w|z;\theta)P(z|d;\pi)$$

$$\hat{P}_{LSA}(d,w) = P(d)\sum P(w|z)P(z|d) = \sum P(d|z)P(z)P(w|z)$$

PLSI representation

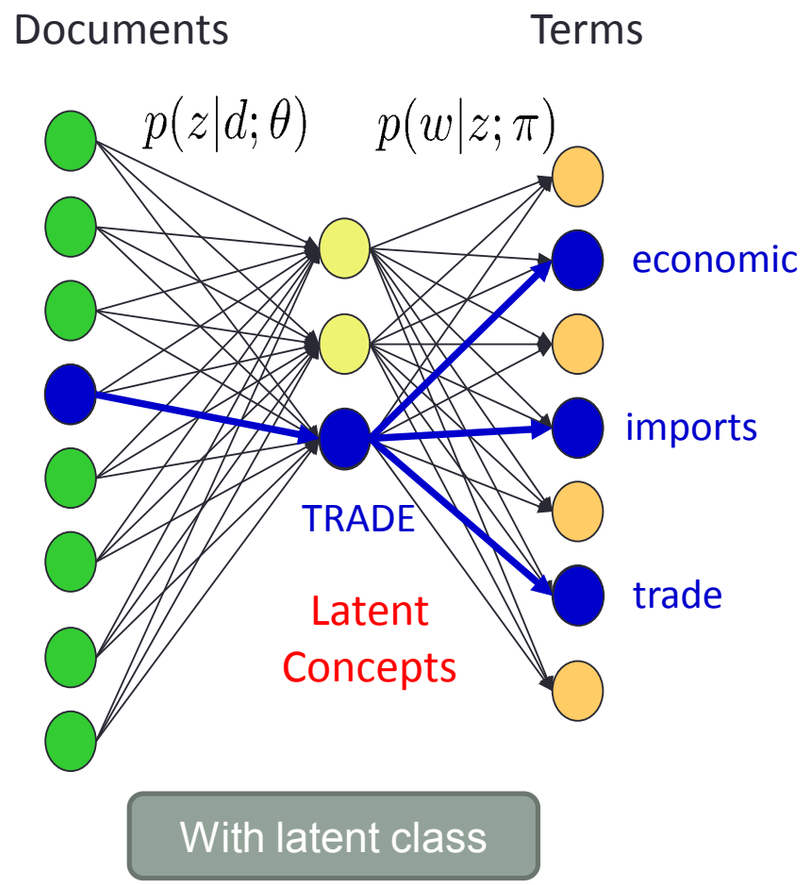
What is the probability of seeing a word w **and** a document d ?

$$p(d,w) = p(w|d).p(d)$$

- Select a document with probability $P(d)$
- Pick a latent class z with probability $P(z|d;\theta)$
- Generate a word w with probability $P(w|z;\pi)$

$$\hat{P}_{LSA}(w|d) = \sum_{z \in Z} P(w|z;\theta)P(z|d;\pi)$$

$$\hat{P}_{LSA}(d,w) = P(d)\sum P(w|z)P(z|d) = \sum P(d|z)P(z)P(w|z)$$



MLE Formulation

Find all the parameters such that the probability of observing the corpus is maximized.

Likelihood function to be maximized: $L = \prod_{i=1}^{N} \prod_{j=1}^{M} P(d_i, w_j)^{n(d_i, w_j)}$

$$\log L = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(d_i) P(z_k|d_i) P(w_j|z_k)$$

$$= \sum_{i=1}^{N} n(d_i) \left[\log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[\sum_{k=1}^{K} P(z_k|d_i) P(w_j|z_k) \right] \right]$$

MLE Formulation

Find all the parameters such that the probability of observing the corpus is maximized.

Likelihood function to be maximized: $L = \prod_{i=1}^{N} \prod_{j=1}^{M} P(d_i, w_j)^{n(d_i, w_j)}$

Document

$$\log L = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(d_i) P(z_k|d_i) P(w_j|z_k)$$

$$= \sum_{i=1}^{N} n(d_i) \left[\log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[\sum_{k=1}^{K} P(z_k|d_i) P(w_j|z_k) \right] \right]$$

MLE Formulation

Find all the parameters such that the probability of observing the corpus is maximized.

Likelihood function to be maximized: $L = \prod_{i=1}^{N} \prod_{j=1}^{M} P(d_i, w_j)^{n(d_i, w_j)}$

corpus/ collection

Document

$$\log L = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(d_i) P(z_k|d_i) P(w_j|z_k)$$

$$= \sum_{i=1}^{N} n(d_i) \left[\log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[\sum_{k=1}^{K} P(z_k|d_i) P(w_j|z_k) \right] \right]$$

MLE Formulation

Find all the parameters such that the probability of observing the corpus is maximized.

Likelihood function to be maximized: $L = \prod_{i=1}^{N} \prod_{j=1}^{M} P(d_i, w_j)^{n(d_i, w_j)}$

corpus/ collection

Document

$$\log L = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log P(d_i, w_j) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) \log \sum_{k=1}^{K} P(d_i) P(z_k|d_i) P(w_j|z_k)$$

$$= \sum_{i=1}^{N} n(d_i) \left[\log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[\sum_{k=1}^{K} P(z_k|d_i) P(w_j|z_k) \right] \right]$$

Only this is the difficult part

Expectation Maximisation

$$\log L = \sum_{i=1}^{N} n(d_i) \left[\log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[\sum_{k=1}^{K} P(z_k|d_i) P(w_j|z_k) \right] \right]$$

marginal prob. w/ sum

Estimated directly from data:

- $P(d_i)$: uniform or related to popularity of the document d_i
- $n(d_i)$: number of words in d_i
- $n(d_i, w_j)$: count of word w_j in d_i

- Use Expectation Maximisation (EM Algorithm) procedure when there are **latent variables**
- **Issue:** we have a marginal probability which is difficult to maximize analytically (mainly because of the sum)

Expectation Maximisation

EM for our problem (repeat until convergence):

1. **E-step:** Calculate posterior probabilities for latent variables given the observations and current estimates
2. **M-step:** Update parameters using the posterior probabilities in E-step to increase $\log L$

$$\log L = \sum_{i=1}^{N} n(d_i) \left[\log P(d_i) + \sum_{j=1}^{M} \frac{n(d_i, w_j)}{n(d_i)} \log \left[\sum_{k=1}^{K} P(z_k|d_i) P(w_j|z_k) \right] \right]$$

1. **E-step:** Calculating posterior probabilities using the current estimates

$$P(z_k|d_i, w_j) = \frac{P(w_j, z_k|d_i)}{P(w_j|d_i)} = \frac{P(w_j|z_k, d_i) P(z_k|d_i)}{\sum_{i=1}^{K} P(w_j|z_i, d_i) P(z_i|d_i)}$$

2. **M-step:** Maximizing $\log L$ having the posterior probability

$$P(w_j|z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j) P(z_k|d_i, w_j)}{\sum_{m=1}^{M} \sum_{i=1}^{N} n(d_i, w_m) P(z_k|d_i, w_m)} \qquad P(z_k|d_i) = \frac{\sum_{j=1}^{M} n(d_i, w_j) P(z_k|d_i, w_j)}{n(d_i)}$$

References and Further Readings

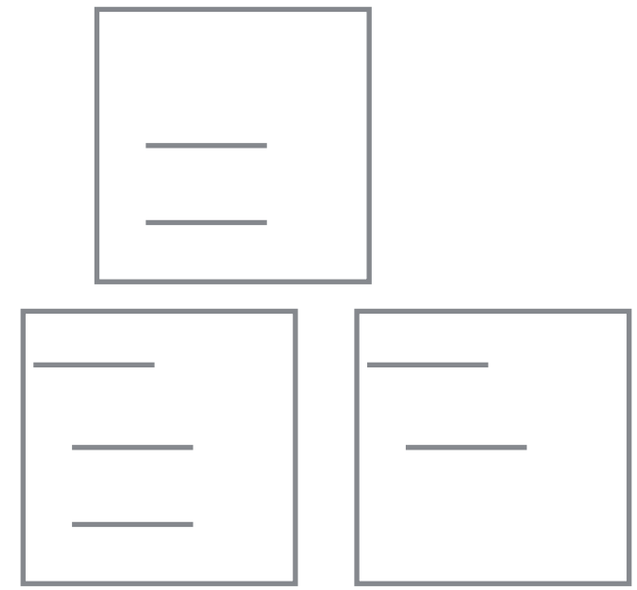
- Information retrieval: (http://www.ir.uwaterloo.ca/book/)
  - Stefan Büttcher, Google Inc. , Charles L. A. Clarke, Univ. of Waterloo, Gordon V. Cormack, Univ. of Waterloo

- Foundations of Information retrieval: Manning, Schutze, Raghavan
  - http://nlp.stanford.edu/IR-book/pdf/12lmodel.pdf

- Hofmann, Thomas. "Probabilistic latent semantic indexing." Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 1999.

Temporal Ranking

- Queries with temporal expressions
  - fifa world cup in 1998
- Documents also mention temporal expressions
  - "Clinton was the president in 1990s"
  - "Interstellar was slated for release in july 2014"



$$P(Q|D) = P(Q_{text}|D_{text}).P(Q_{time}|D_{time})$$

- Assume time mentions are independent of text
- Assume temporal expressions are independent of each other

How do we compute $P(Q_{time}|D_{time})$