So Far…

- It's time for
  - Unsupervised learning
    - We are only given inputs
    - Goal: find "interesting patterns"

    - Discovering clusters
      - Clustering
    - Discovering latent factors
      - Dimensionality reduction
      - Topic modeling
      - Matrix factorization

Topic Modeling

Deng Cai (蔡登)

College of Computer Science
Zhejiang University

dengcai@gmail.com

Topic Modeling

- A topic model is a type of statistical model for discovering the latent "topics" that occur in a collection of documents.

- Application
  - Document Generation
  - Information Retrieval
  - …

Bag-of-Words (BOW)

- Assumes order of words has no significance

  e.g., the term "home made" has the same probability as "made home"

- It is a simplifying assumption used in natural language processing and information retrieval

Document Generation

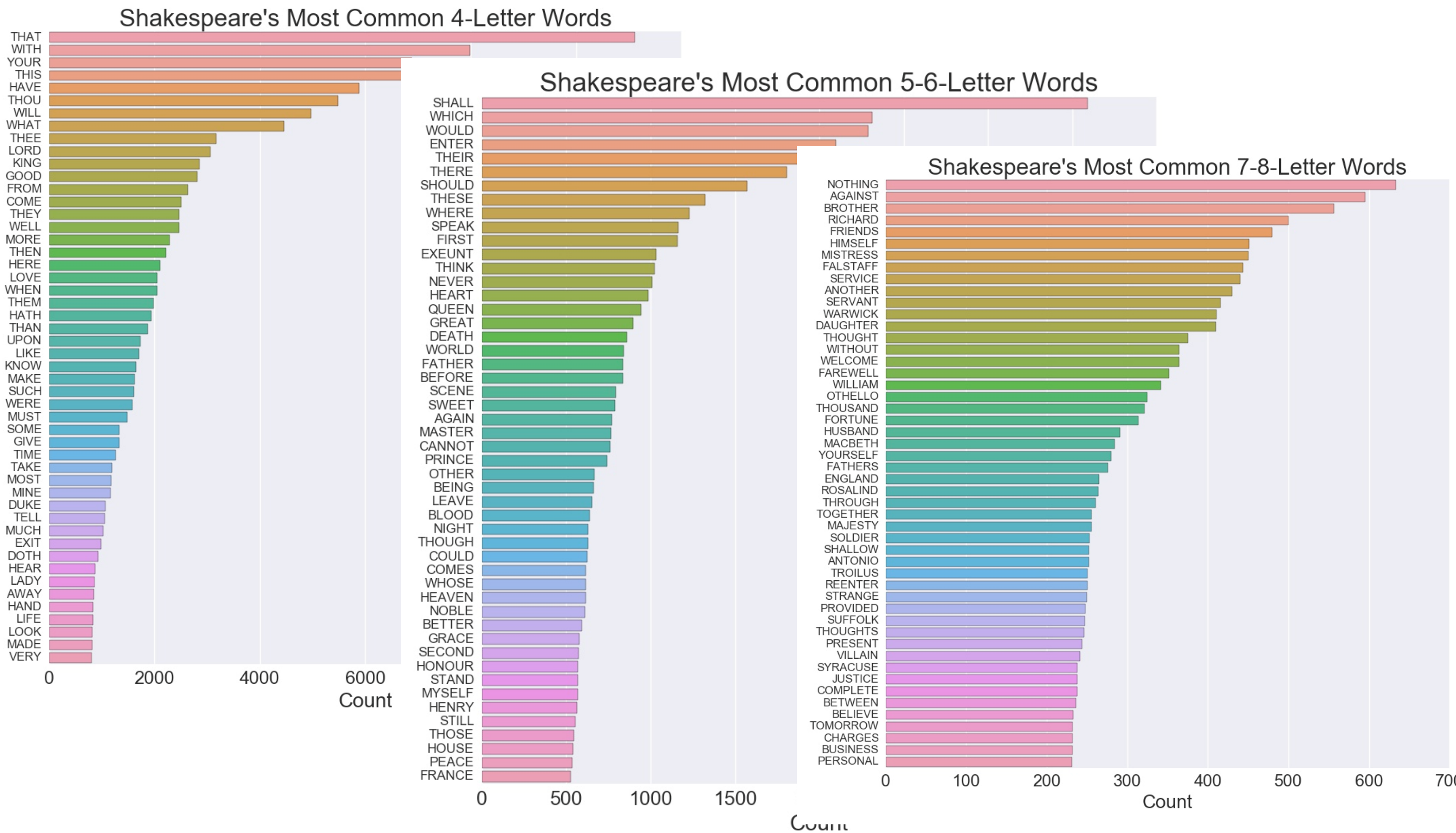
- Terminology
  - Word (w): element in a vocabulary set
  - Document (d): collection of words, not sequence (why?)
  - Corpus (D): collection of documents

- Generate a document by yielding each word one by one independently, wherein $p(w) = \theta_w$, $\sum_{w=1}^{V} \theta_w = 1$

θ_w ?

Unigram Model

- Different words appear with different frequency
  - Example Corpus: Shakespeare's Literature



Unigram Model

- Frequency based θ_w

$$\theta_w = \frac{n_w}{\sum_v^V n_v}$$

Number of appearances of a word in a corpus

- Theoretical basis

$$d_i = \{word_1, word_2, \dots, word_n\}$$

$$D = \{d_1, d_2, \dots, d_N\}$$

$$p(word_j) = \theta_w \text{ (if } word_j = w)$$

$$p(D;\theta) = \prod_{i=1}^{N} p(d;\theta) = \prod_{i=1}^{N}\prod_{j=1}^{n} p(word_j) = \prod_{w}^{V} \theta_w^{n_w}$$

Unigram Model

$$\log p(D;\ \theta) = \sum_{w \in V} n_w \log \theta_w$$

- Maximum Likelihood Estimation

$$\theta = \arg\max_{\theta} \log p(D; \theta)$$

$$\text{s.t.} \sum_{w \in V} \theta_w = 1$$

- Solved with Lagrange multiplier

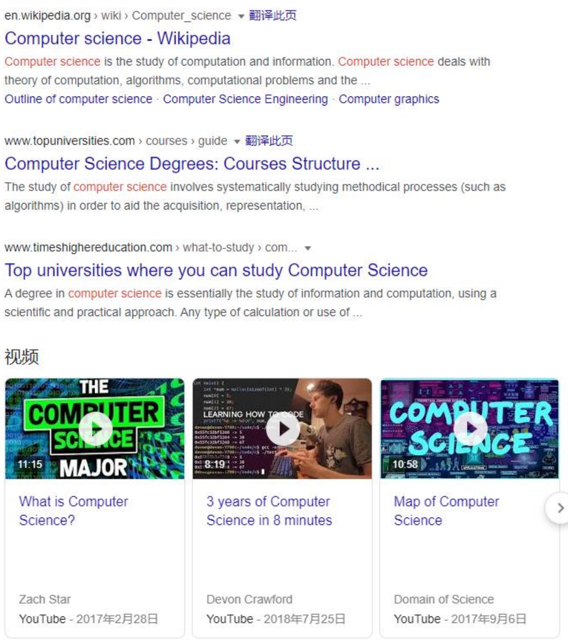
$$\max\ L(\theta) = \sum_{w \in V} n_w \log \theta_w + \lambda \left(\sum_{w \in V} \theta_w - 1 \right)$$

$$\theta = \frac{n_w}{\sum_{w \in V} n_w}$$

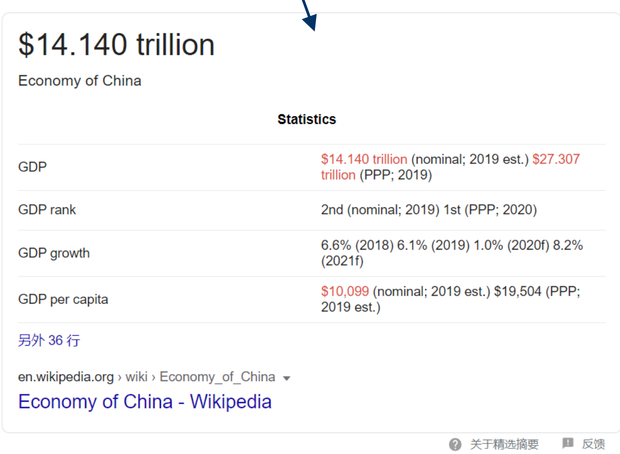
Information Retrieval（IR）

- Return the documents relevant to the query

Query = “computer science”



Query=“China GDP 2019”



Salton's Vector Space Model (Prior to 1988)

- Represent each document by a high-dimensional vector in the space of words


TOTAL

All About The Company

Global Activities
Corporate Structure
TOTAL's Story
Upstream Strategy
Downstream Strategy
Chemicals Strategy
TOTAL Foundation
Homepage

all about the **company**

Our energy exploration, production, and distribution operations span the globe, with activities in more than 100 countries.

At TOTAL, we draw our greatest strength from our fast-growing oil and gas reserves. Our strategic emphasis on natural gas provides a strong position in a rapidly expanding market.

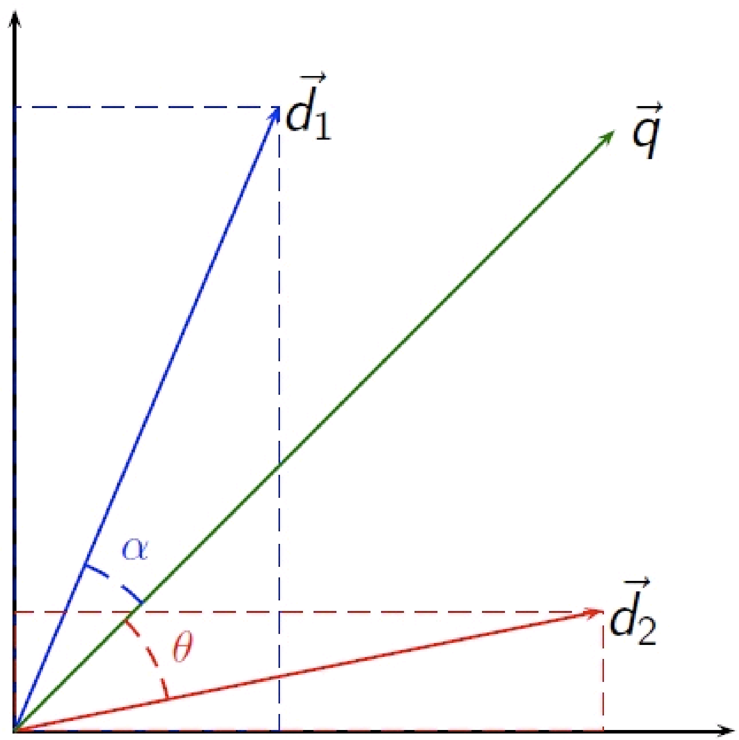
Our expanding refining and marketing operations in Asia and the Mediterranean Rim complement already solid positions in Europe, Africa, and the U.S.

Our growing specialty chemicals sector adds balance and profit to the core energy business.

aardvark	0
about	2
all	2
Africa	1
apple	0
anxious	0
...	
gas	1
...	
oil	1
...	
zebra	0

Query

- Compute the similarity between *queries(q)* and *documents(d)*



$$\cos(\boldsymbol{q}, \boldsymbol{d}) = \frac{\boldsymbol{q}^T \boldsymbol{d}}{\|\boldsymbol{q}\|\|\boldsymbol{d}\|}$$

Simple, intuitive

Fast to compute, because both they are sparse

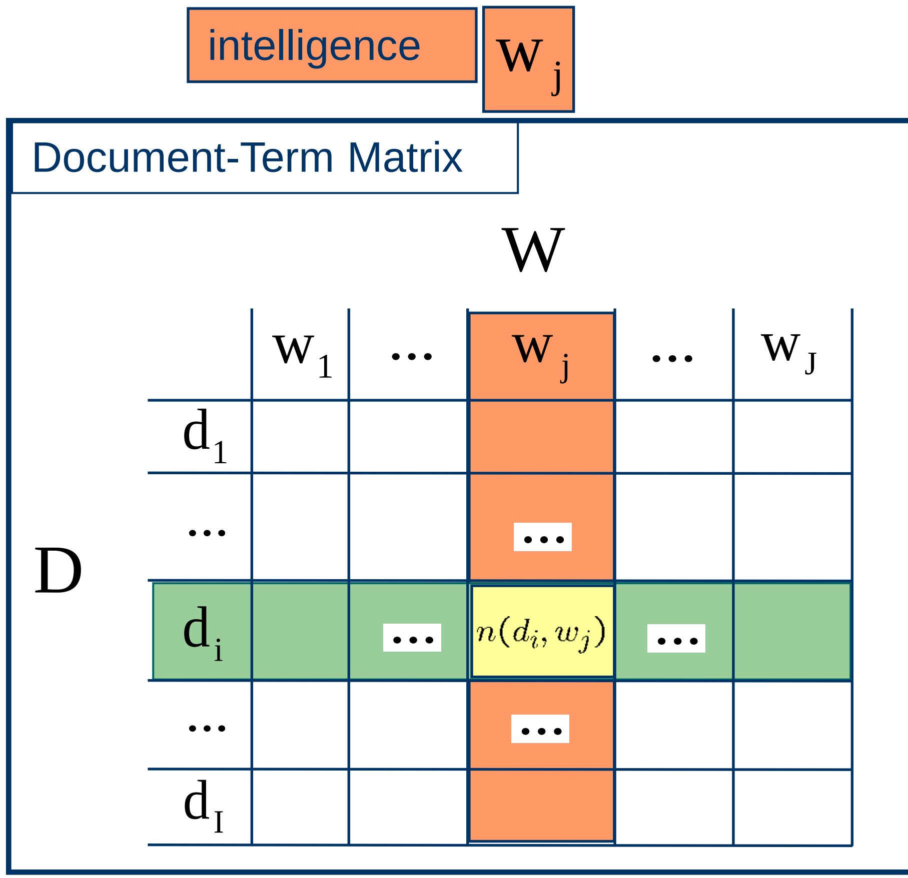
Retrieval Methods

- Rank documents according to similarity with query
- Term weighting schemes, for example, TF-IDF

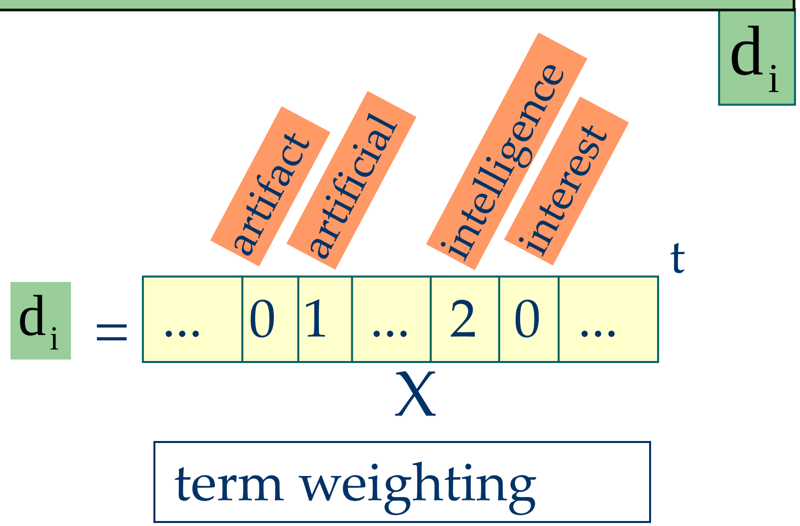
Document-Term Matrix

D = Document collection

W = Lexicon/Vocabulary



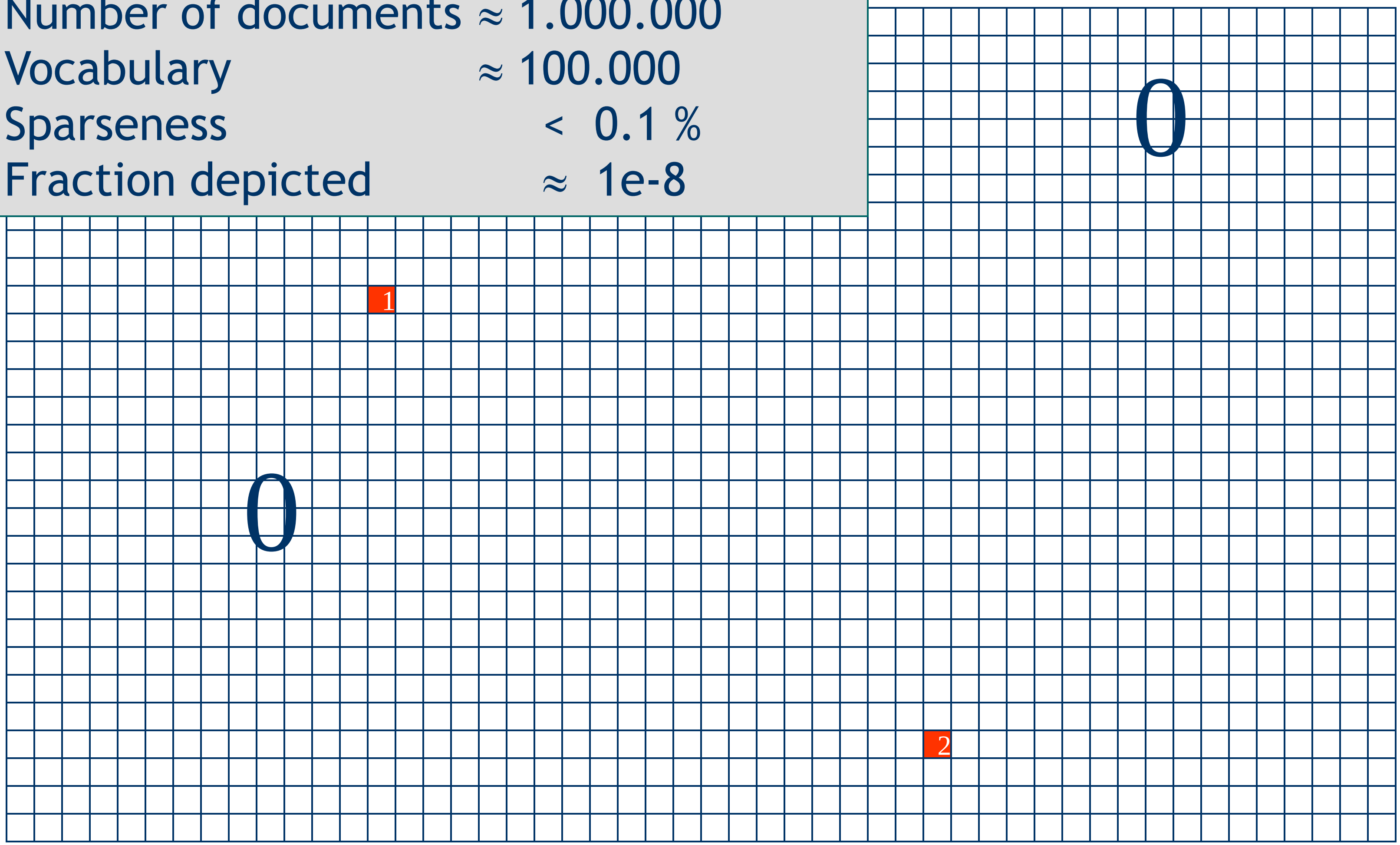
Texas Instruments said it has developed the first 32-bit computer chip designed specifically for artificial intelligence applications [...]



A 100 Millionths of a Typical Document-Term Matrix

Typical:
- Number of documents ≈ 1.000.000
- Vocabulary ≈ 100.000
- Sparseness < 0.1 %
- Fraction depicted ≈ 1e-8

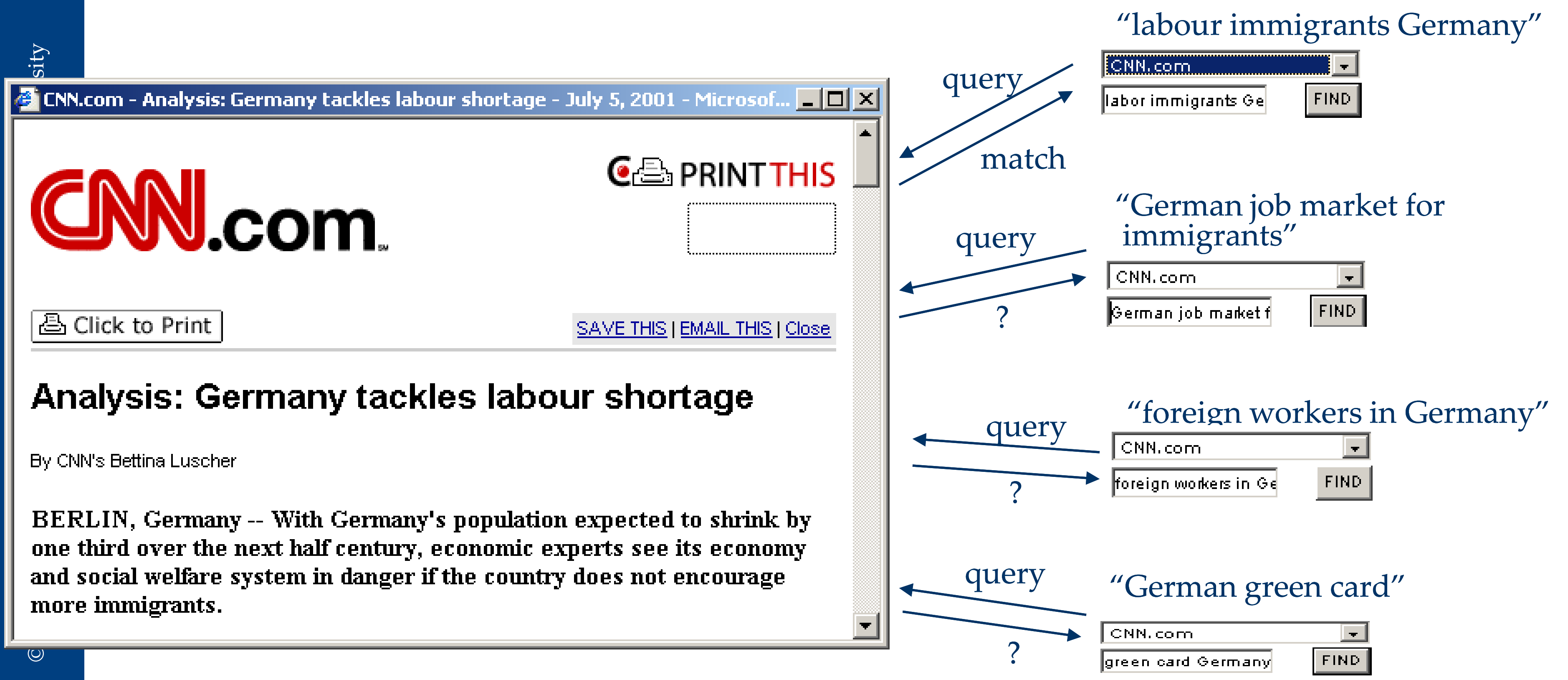
$\mathbf{A} =$



Robust Information Retrieval — *Beyond Keyword-based Search*

Vocabulary Mismatch Problem

- different people using different vocabulary to describe the same concept
- matching queries and documents based on keywords is insufficient



G. W. Furnas, T. K. Landauer, L. M. Gomez , S. T. Dumais, The Vocabulary Problem in Human-System Communication: an Analysis and a Solution, Bell Communications Research, 1987

The lost meaning of words

- Polysemy: words with multiple meanings
  - The vector space model is unable to discriminate between different meaning of the same word.

$$\text{sim}(\boldsymbol{d}, \boldsymbol{q}) < \cos\left(\angle\left(\vec{\boldsymbol{d}}, \vec{\boldsymbol{q}}\right)\right)$$

- Synonymy: separate words that have the same meaning.
  - No associations between words are made in the vector space representation

$$\text{sim}(\boldsymbol{d}, \boldsymbol{q}) > \cos\left(\angle\left(\vec{\boldsymbol{d}}, \vec{\boldsymbol{q}}\right)\right)$$

There is a disconnect between *topics* and *words*

Language Model Paradigm in IR

- Probabilistic relevance model
  - Random variables

$$R_d \in \{0,1\} \quad : \quad \text{relevance of document } d$$
$$q \subseteq \Sigma \quad : \quad \text{query, set of words}$$

  - Bayes' rule

probability of generating a query q to ask for relevant d

prior probability of relevance for document d (e.g. quality, popularity)

$$P(R_d = 1|q) = \frac{P(q|R_d = 1) \cdot P(R_d = 1)}{P(q)}$$

probability that document d is relevant for query q

J. Ponte and W.B. Croft, A Language Model Approach to Information Retrieval, ACM SIGIR, 1998.

Language Model Paradigm

$$P(R_d = 1|q) \propto \underbrace{P(q|R_d = 1)}_{2} \underbrace{P(R_d = 1)}_{1}$$

1 ▸ First contribution: **prior probability of relevance**

- simplest case: uniform (drops out for ranking)
- **popularity**: document usage statistics (e.g. library circulation records, download or access statistics, hyperlink structure)

▸ Second contribution: **query likelihood**

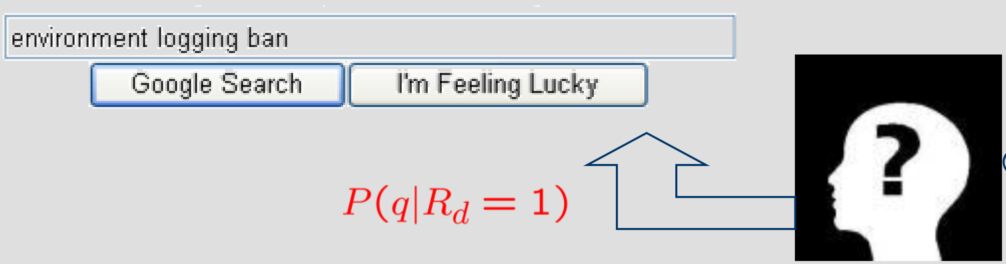
2

- query terms q are treated as a **sample** drawn from an (unknown) relevant document

Language Model Paradigm

- Query generation model: how might a query look like that would ask for a specific document?
  - Maron & Kuhns: Indexer **manually** assigns probabilities for pre-specified set of tags/terms
  - Ponte & Croft: **Statistical estimation** problem

Think of a relevant document. Formulate a query by picking some of the keywords as query terms.



Query Likelihood

$$P(q|R_d = 1) \equiv P(q|d)$$

- $q = (w_1, \cdots, w_q)$
- Independent Assumption

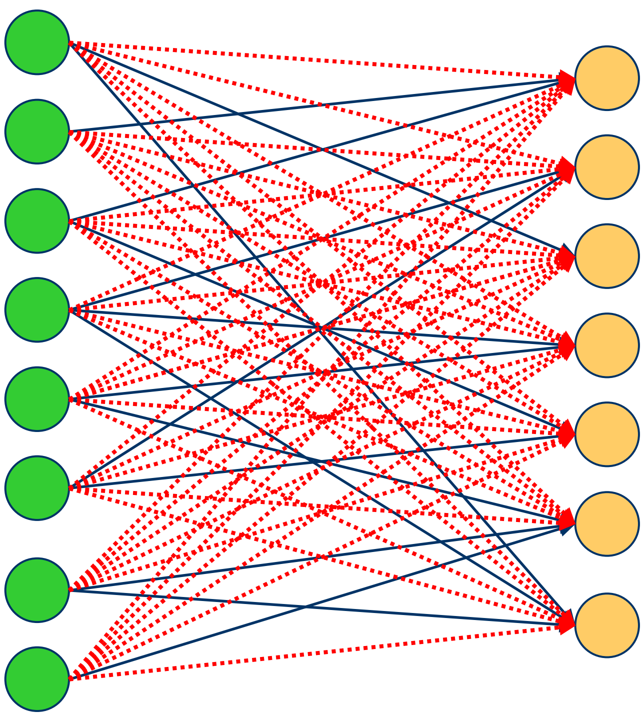
$$P(q|d) = \Pi_{w \in q} P(w|d)$$

$P(w|d)$?

Naive Approach

Documents

Terms



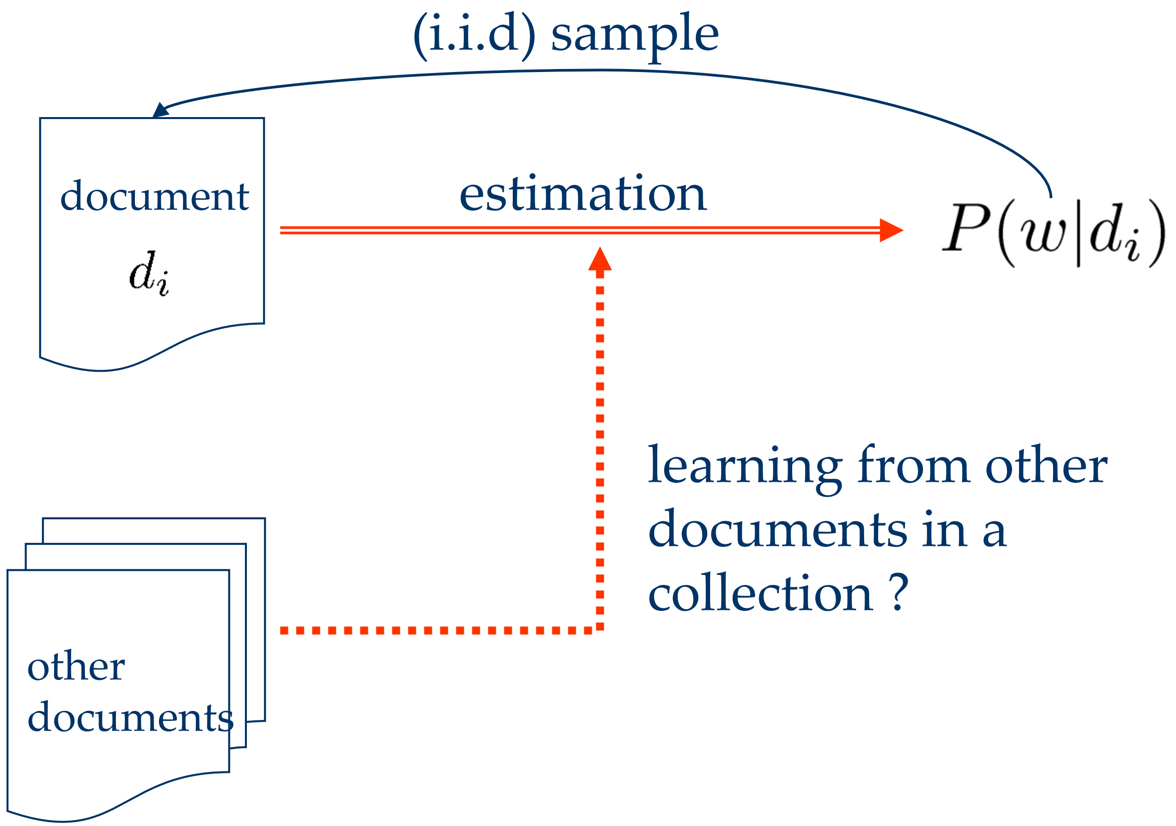
Maximum Likelihood Estimation

number of occurrences of term w in document d

$$\hat{P}(w|d) = \frac{n(d,w)}{\sum_{w'} n(d,w')}$$

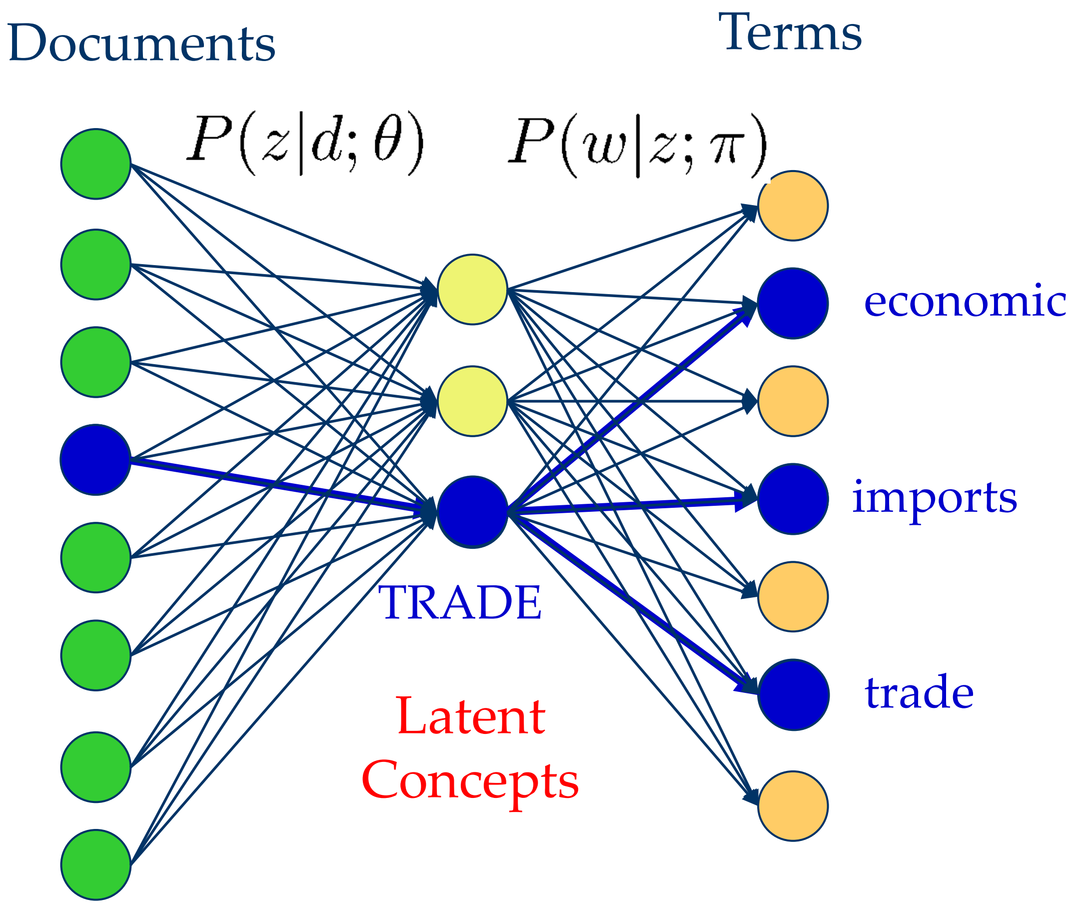
Zero frequency problem: terms not occurring in a document get zero probability

Estimation Problem



- **Crucial question**: In which way can the document collection be utilized to improve probability estimates?

Probabilistic Latent Semantic Analysis



$$\hat{P}(w|d) = \sum_{z} P(w|z)P(z|d)$$

Concept expression probabilities are estimated based on all documents that are dealing with a concept.

"Unmixing" of superimposed concepts is achieved by statistical learning algorithm.

Probabilistic Latent Semantic Analysis

- PLSA evolved from Latent semantic analysis, adding a sounder probabilistic model
- It was introduced in 1999 by Thomas Hofmann (UAI'99)
- It is related to non-negative matrix factorization (NMF)

pLSA – Latent Variable Model

- Structural modeling assumption (mixture model)

$$\hat{P}_{\mathrm{LSA}}(w|d) = \sum_z P(w|z;\theta) P(z|d;\pi)$$

Document language model

Latent concepts or topics

Concept expression probabilities

Document-specific mixture proportions

Model fitting

pLSA via Likelihood Maximization

- **Log-Likelihood**

$$l(\theta, \pi; \mathbf{N}) = \sum_{d,w} \underbrace{n(d,w)}_{\text{Observed word frequencies}} \log\Big(\underbrace{\sum_{z} P(w|z;\theta)P(z|d;\pi)}_{\hat{P}_{\text{LSA}}(w|d)}\Big)$$

$$l(\theta, \pi; \mathbf{N}) \xrightarrow{\text{argmax}} (\hat{\theta},\ \hat{\pi})$$

Observed
word frequencies

$\hat{P}_{\text{LSA}}(w|d)$

Predictive probability
of pLSA mixture model

- **Goal**: Find model parameters that maximize the log-likelihood, i.e. maximize the average predictive probability for observed word occurrences (**non-convex optimization problem**)

EM Algorithm: Derivation

- Q-parameterized lower bound on log-likelihood

$$l(\theta,\pi;Q) = \sum_{\langle d,w,r\rangle} \sum_z Q_r(z) \log \frac{P(w|z;\theta)P(z|d;\pi)}{Q_r(z)}$$

observed pairs
with index r

Q distribution

- Follows from **Jensen's inequality**

$$\begin{aligned} l(\theta,\pi) &= \sum_{\langle d,w,r\rangle} \log \sum_z Q_r(z) \frac{P(w|z;\theta)P(z|d;\pi)}{Q_r(z)} \\ &\geq \sum_{\langle d,w,r\rangle} \sum_z Q_r(z) \log \frac{P(w|z;\theta)P(z|d;\pi)}{Q_r(z)} = l(\theta,\pi;Q) \end{aligned}$$

Expectation Maximization Algorithm

- E step: posterior probability of latent variables ("concepts")

$$P(z|d,w) = \frac{P(z|d;\pi)P(w|z;\theta)}{\sum_{z'} P(z'|d;\pi)P(w|z';\theta)}$$

Probability that the occurence of term w in document d can be "explained" by concept z

- M step: parameter estimation based on "completed" statistics

$$P(w|z;\theta) \propto \underbrace{\sum_d n(d,w)P(z|d,w)}_{\text{how often is term w associated with concept z ?}}, \quad P(z|d;\pi) \propto \underbrace{\sum_w n(d,w)P(z|d,w)}_{\text{how often is document d associated with concept z ?}}$$

how often is term w associated with concept z ?

how often is document d associated with concept z ?

A.P. Dempster, N.M. Laird, and D.B. Rubin, Maximum Likelihood from Incomplete Data via the EM Algorithm, Journal of Royal Statistical Society B, vol. 39, no. 1, pp. 1-38, 1977

"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
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

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.

$P(w|z)$ Matrix

Variations of pLSA

- Hierarchical extensions:
  - Asymmetric: MASHA ("Multinomial Asymmetric Hierarchical Analysis")
  - Symmetric: HPLSA ("Hierarchical Probabilistic Latent Semantic Analysis")
- Manifold regularizer:
  - Probabilistic Dyadic Data Analysis with Local and Global Consistency
- Generative models:
  - **Latent Dirichlet allocation** - adds a Dirichlet prior on the per-document topic distribution, trying to address an often-criticized shortcoming of PLSA, namely that it is not a proper generative model for new documents and at the same time avoid the overfitting problem.