Lec07. Data Analytics for Texts (cont’d)

Recap: Text Analytics Application

Social Media Analytics

1. **Brand Reputation Monitoring**
   - Social Media, Blogs, News sites
2. **Advertising Performance Metrics**
   - Social Media, Blogs

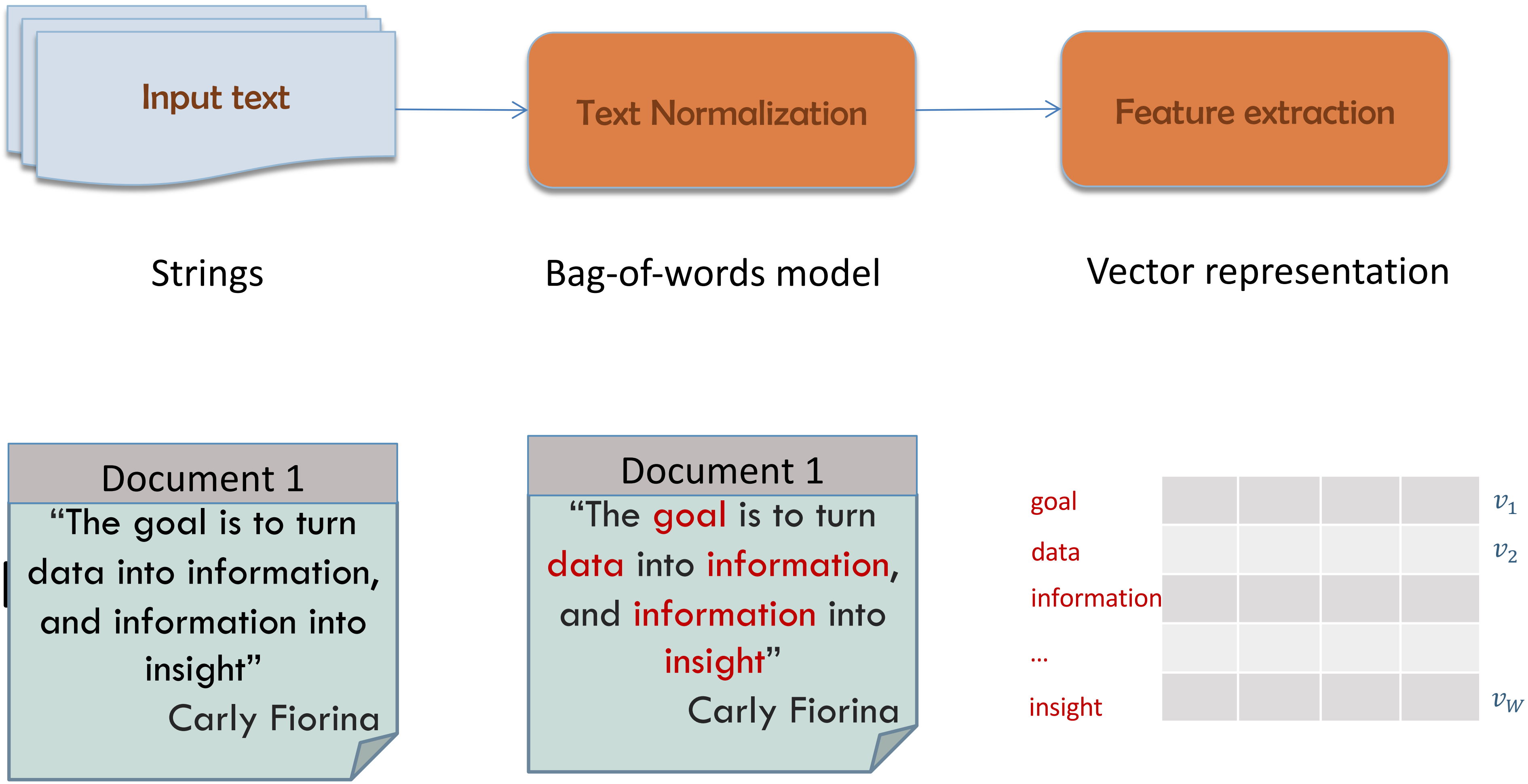
CRM

1. **Complaint Tracking**
   - Social Media, Blogs, News
2. **Call Center Analytics**
   - Call Center Transcripts
3. **Competitive Analysis**
   - Communication, Surveys
4. **Market Research**
   - Surveys, Feedback

Predictive Analytics

1. **Prediction of Stocks**
   - Financial News, Newspapers
2. **Prediction of Election Results**
   - Social Media
3. **Movie Intake**
   - Twitter

Recap: Text Syntactical Analysis



Recap: TF-IDF example

Document 1

"The goal is to turn data into information, and information into insight"

Carly Fiorina

Document 2

"You can have data without information, but you cannot have information without data."

Daniel Keys Moran

term frequency (tf)

Terms	goal	data	information	insight	you
Doc1	1	1	2	1	0
Doc2	0	2	2	0	1

document frequency (df)

Terms	goal	data	information	insight	you
df	1	2	2	1	1

inverse document frequency (idf)

Terms	goal	data	information	insight	you
idf	0.69	0	0	0.69	0.69

$\log\frac{2}{1}$ $\log\frac{2}{2}$

tfidf

Terms	goal	data	information	insight	you
Doc1	0.69	0	0	0.69	0
Doc2	0	0	0	0	0.69

$$\frac{0.69}{\sqrt{0.69^2 + 0.69^2}}$$

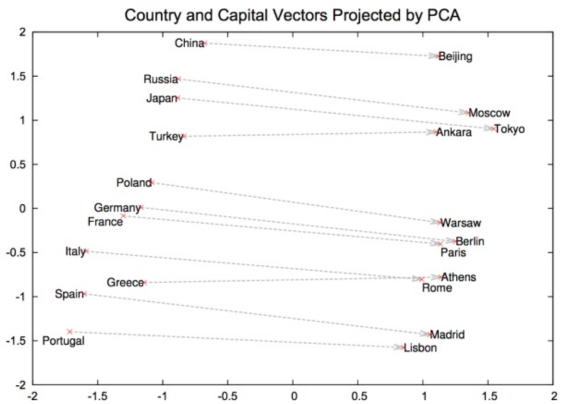
tfidf (l2 normalized)

Terms	goal	data	information	insight	you
Doc1	0.71	0	0	0.71	0
Doc2	0	0	0	0	0.69

Recap: Text Representation Learning

❖ Words in a document are not independent, but stand in a semantic relation to one another.

- Word embedding: neural embedding and vector representation of words
  - Similar words will stay closer
  - State-of-the-art: word2vec



word2vec

[Mikolov et al. 2013]

Recap: Sentiment Analysis

- Computational study of opinions, sentiments, evaluations, attitudes, appraisal, affects, views, emotions, subjectivity, etc., expressed in text.
  - E.g. extract from text how people feel about different products (Reviews, blogs, discussions, news, comments, feedback, …)
- Is a review positive or negative toward the movie?
  - “Unbelievably disappointing”
  - “Full of zany characters and richly applied satire, and some great plot twists”
  - “This is the greatest screwball comedy ever filmed”
  - “It was pathetic. The worst part about it was the boxing scenes”

http://nlp.stanford.edu:8080/sentiment/rntnDemo.html

Course structure

W1. Data Processing with Python

W2. Data Exploration with Python

W3. Data Modeling with Pytyhon

W4. Data Analytics for Timeseries

Holiday

W5-6-7. Data Analytics for Texts

W8. Data Analytics for Images

W9. Data Analytics for Graphs

W10-11. Data Analytics for Other Data

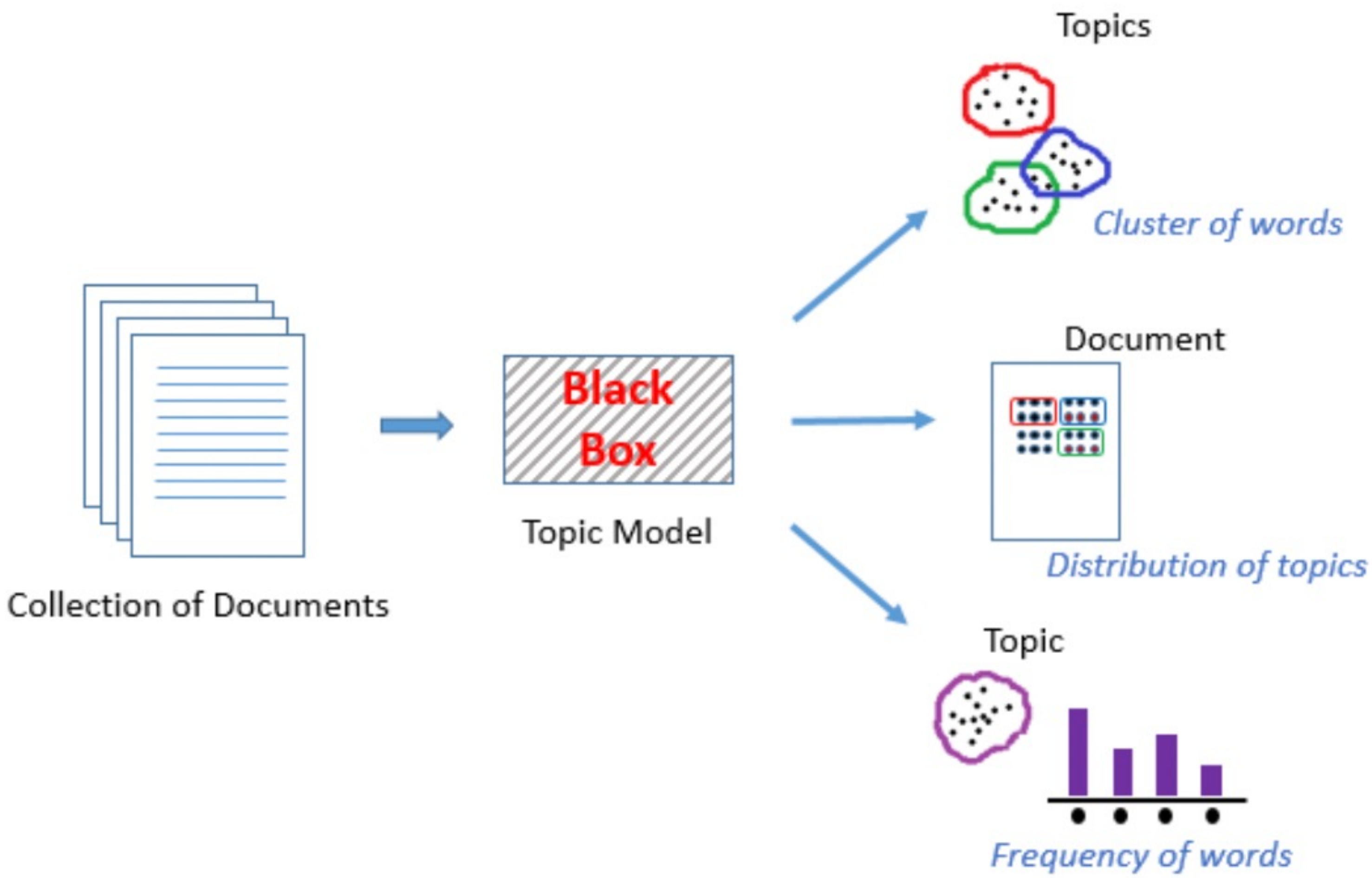
W12. Revision

Lecture Content

- Topic Modeling
- Named Entity Recognition

TOPIC MODELING

Topic Modeling: General



Why Topic Modeling

- Vector space retrieval (TF-IDF) is vague and noisy
  - Based on index terms (i.e. vocabulary)
  - Unrelated documents might be included in the answer set
    - apple (company) vs. apple (fruit)
  - Relevant documents that do not contain at least one index term are not retrieved
    - car vs. automobile

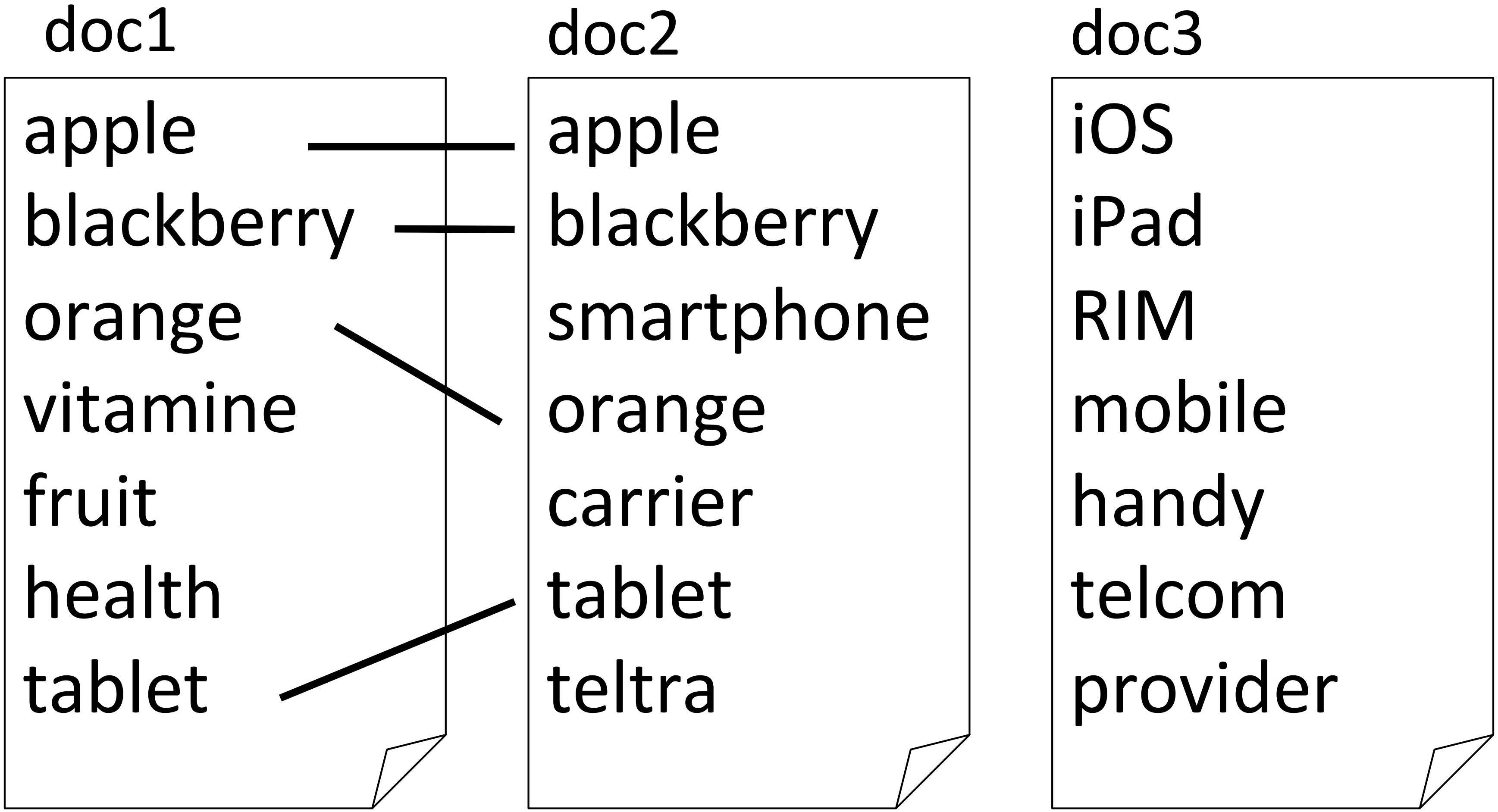
- Observation:
  - The user information need is more related to concepts and ideas than to index terms

The Problem

- Vector Space Retrieval (TF-IDF) handles poorly the following two situations

  1. ***Synonymy***: different terms refer to the same concept
     - E.g. get ‘car’ but does not get ‘automobile’
     - → Result: poor recall (miss relevant documents)
  2. ***Homonymy*:** the same term may have different meanings
     - e.g. apple (company vs fruit), bank (river vs. finance)
     - → Result: poor precision (return irrlevent documents)

Example: 3 documents



High similarity (but actually different)

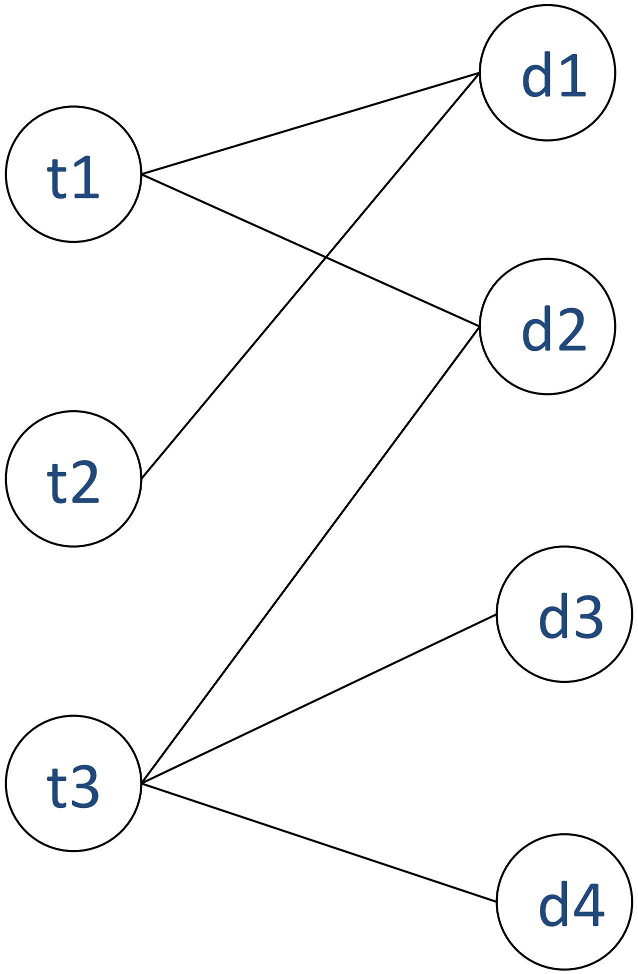
No similarity (but actually similar)

Key Idea

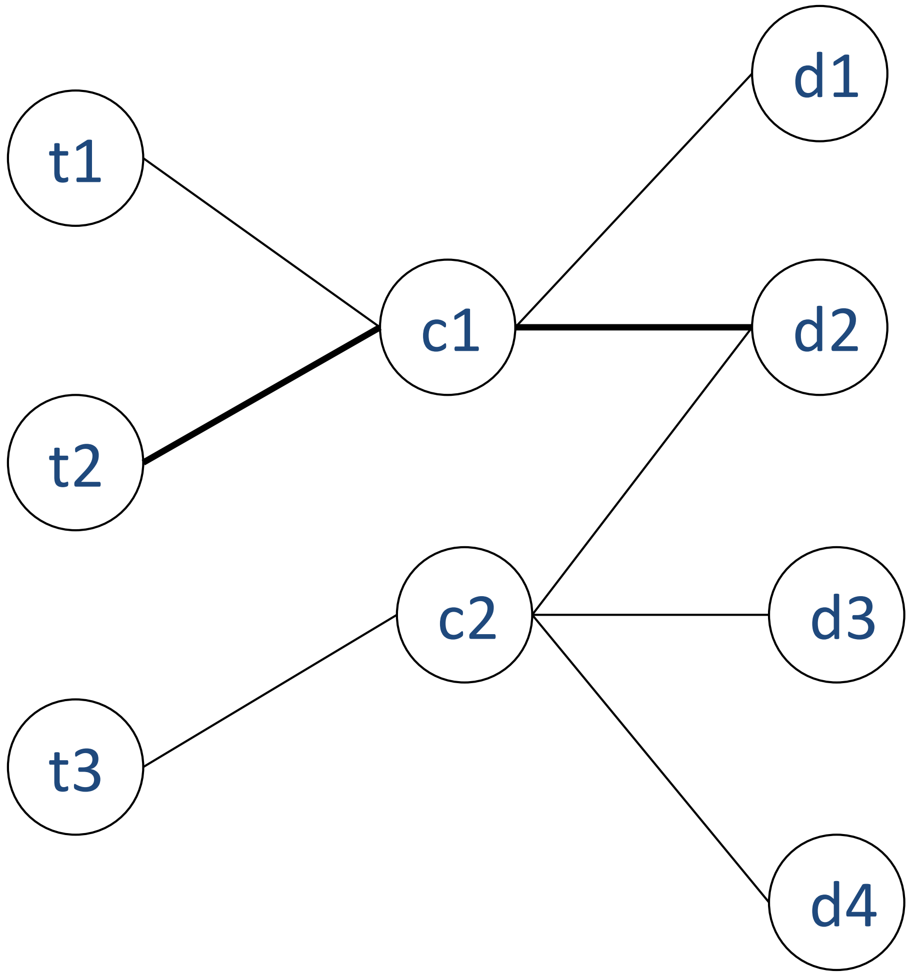
- Map documents and queries into a lower-dimensional space composed of higher-level concepts
  - Each concept represented by a combination of terms
  - Fewer concepts than terms
  - Vehicle = [car, automobile, wheels, auto car, motor car]

- Dimensionality reduction
  - Retrieval (and clustering) in a reduced concept space might be superior to retrieval in the high-dimensional space of index terms

Using Concepts for Retrieval

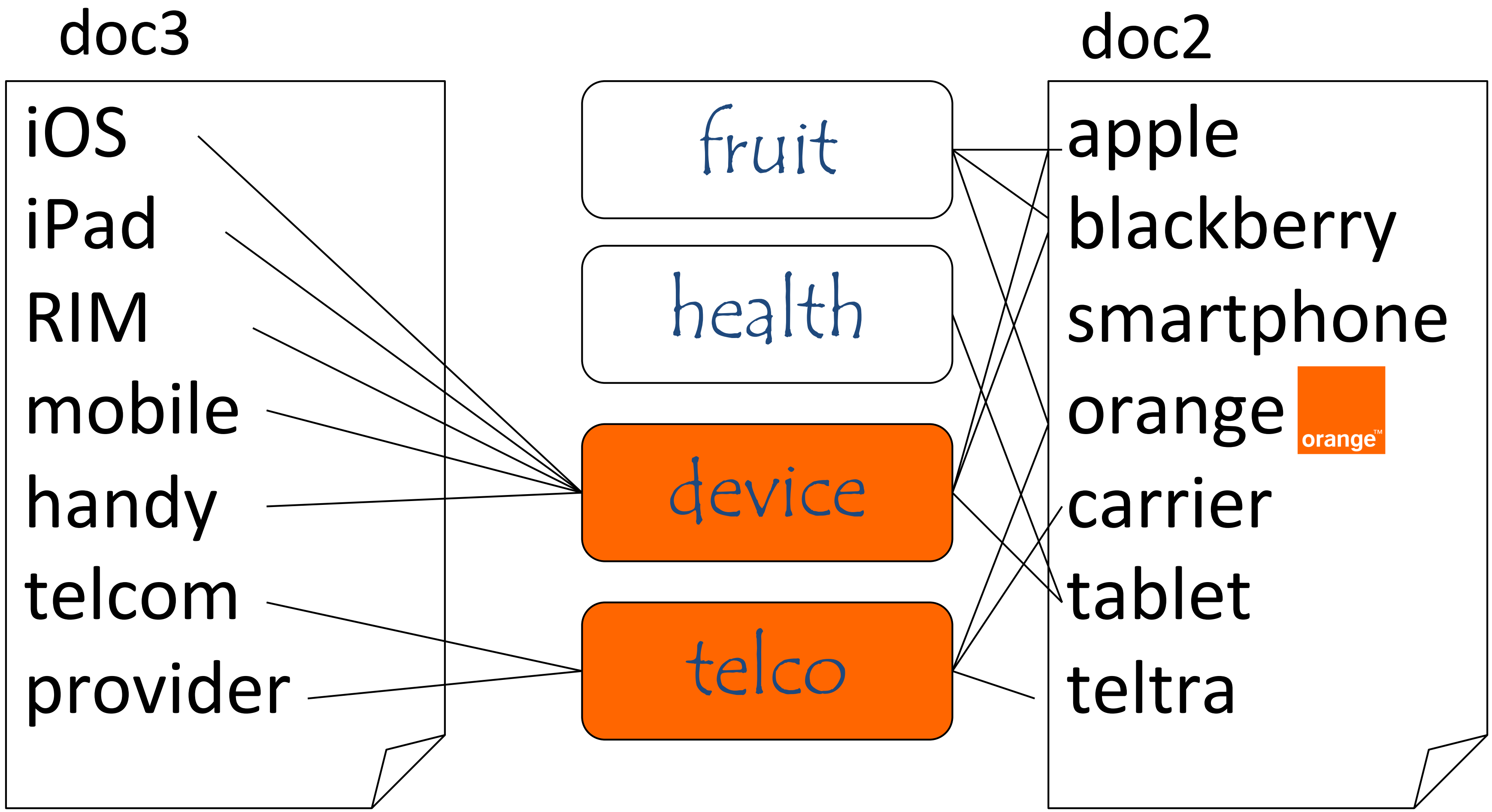


If query = t2 → return d1



If query = t2 → return d1 and d2 because t1 and t2 have the same concept c1

Example: Concept Space



Similarity Computation in Concept Space

- Concept represented by terms, e.g.
  - device = {iOS, iPad, RIM, mobile, handy, tablet, apple, blackberry}

- Document represented by concept vector, counting number of concept terms, e.g.
  - doc3 = (0, 0, 5, 2)

- Similarity computed using cosine or Euclidean

Result

doc1	doc2	doc3
apple	apple	iOS
blackberry	blackberry	iPad
orange	smartphone	RIM
vitamine	orange	mobile
fruit	carrier	handy
health	tablet	telcom
tablet	teltra	provider

cosine(doc1, doc2) = 0.245

cosine(doc2, doc3) = 0.3

cosine(doc1, doc3) = 0.22

Basic Definitions

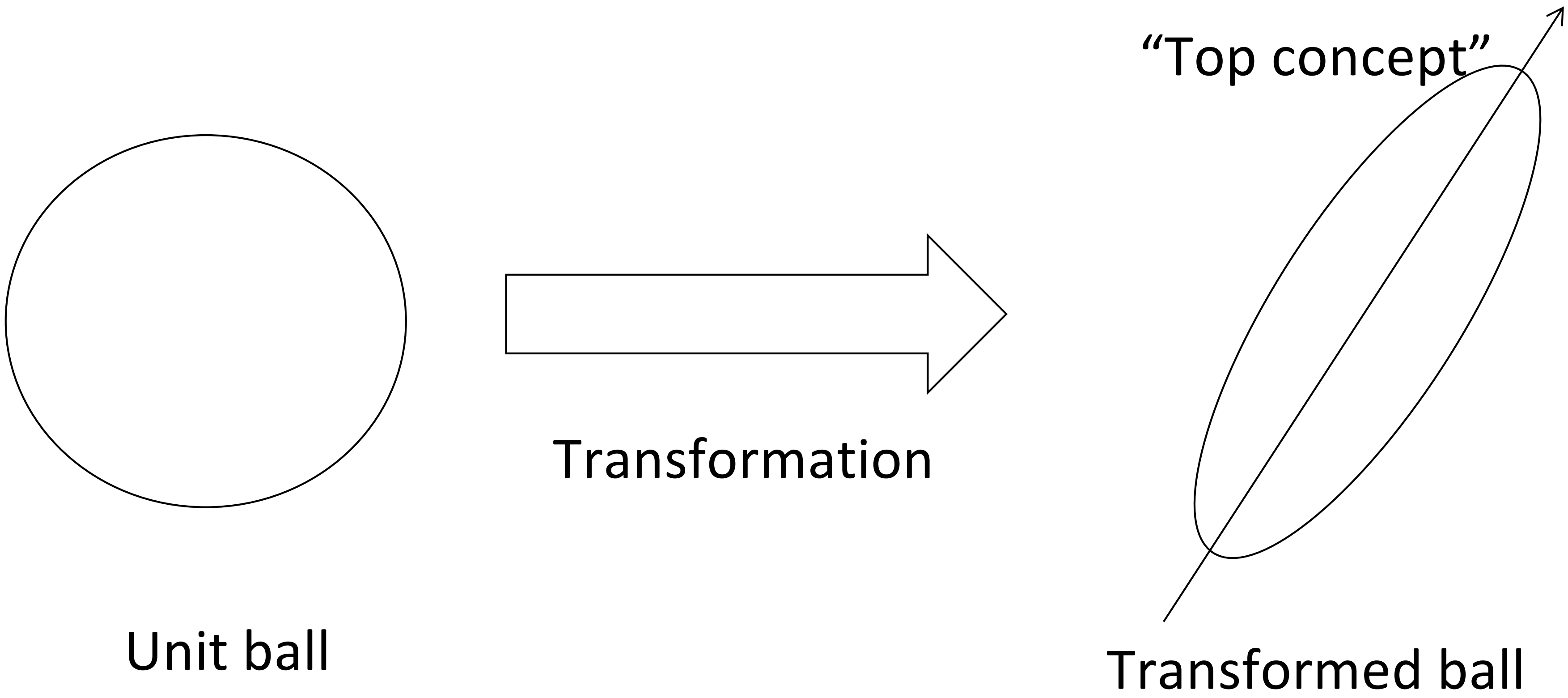
- Problem: how to identify and compute “concepts” ?

- Consider the term-document matrix
  - Let M_{ij} be a term-document matrix with m rows (terms) and n columns (documents)
  - To each element of this matrix is assigned a weight w_{ij} associated with t_i and d_j
  - The weight w_{ij} can be based on a tf-idf weighting scheme

Document	Word1	Word 2	---	Word N
1	0	0.12	---	0.03
2	0.42	0.03	---	0
.	.	.	---	.
M	0	0	---	0.28

Identifying Top Concepts (OPTIONAL)

- Key Idea: extract the essential features of M^t and approximate it by the most important ones



Singular Value Decomposition (SVD) (OPTIONAL)

- Represent Matrix M as M = K.S.D
  - Such a decomposition always exists and is unique

M = K x S x D

M (Words × Documents):

Words	0	1	2	3	4
banana	2	0	4	2	0
kiwi	1	0	5	0	0
apple	1	1	7	4	0
computer	0	1	0	0	4
screen	0	1	0	0	1

=

K (Words × Topics):

Words	A	B	C
banana	0.5	0	0.1
kiwi	0.3	0	0.2
apple	0.2	0.2	0.3
computer	0	0.4	0.2
screen	0	0.4	0.2

x

S:

a	0	0
0	b	0
0	0	c

x

D (Topics × Documents):

Topics	0	1	2	3	4
A	0.8	0	1	0.7	0
B	0	0.9	0	0	1
C	0.2	0.1	0	0.3	0

- S is a diagonal matrix of singular values in decreasing order: each value represents the weight of the corresponding topic
- K is the term-topic matrix
- D is the document-topic matrix

Construction of SVD (OPTIONAL)

- K is the matrix of eigenvectors derived from $M.M^t$
- D is the matrix of eigenvectors derived from $M^t.M$

- Algorithms for constructing the SVD of a m x n matrix have complexity $O(n^3)$ if $m \leq n$

Latent Semantic Indexing (OPTIONAL)

- Like PCA, we can select only the s largest singular values of S
  - Keep the corresponding columns in K and D
- The resultant matrix is called M_s and is given by
  - $M_s = K_s.S_s.D_s$ where s is the dimensionality of the concept space
- The parameter s should be
  - large enough to allow fitting the characteristics of the data
  - small enough to filter out the non-relevant representational details

Summary of Latent Semantic Indexing

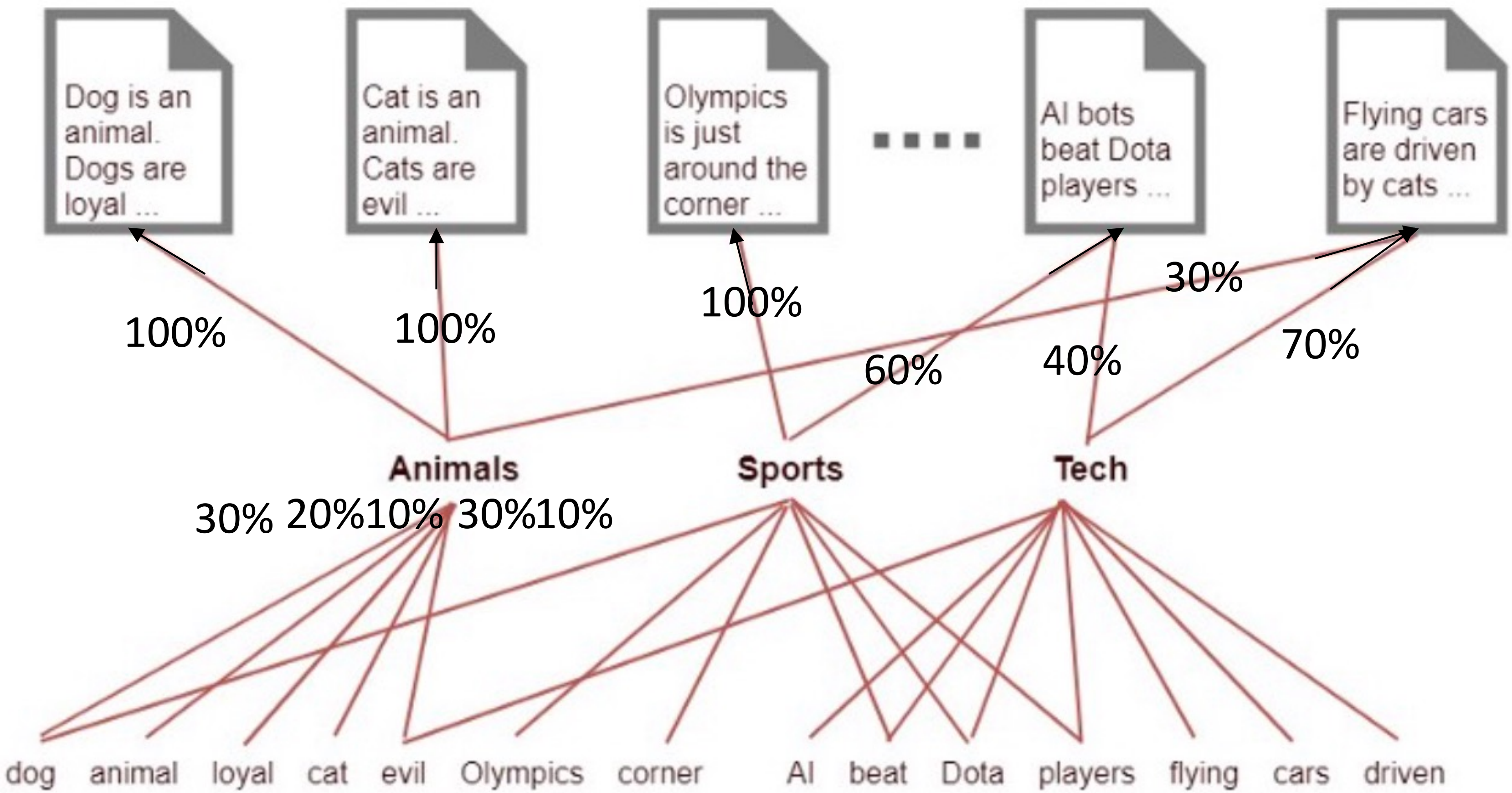
- Latent semantic indexing provides an interesting conceptualization of the IR problem
- Advantages
  - It allows reducing the complexity of the underlying concept representation
  - Facilitates interfacing with the user
- Disadvantages
  - Computationally expensive
  - Poor statistical explanation

Alternative Technique

- Latent Dirichlet Allocation
  - Based on Dirichlet Distribution
  - State-of-the-art method for concept extraction
  - Better explained mathematical foundation
  - Better experimental results

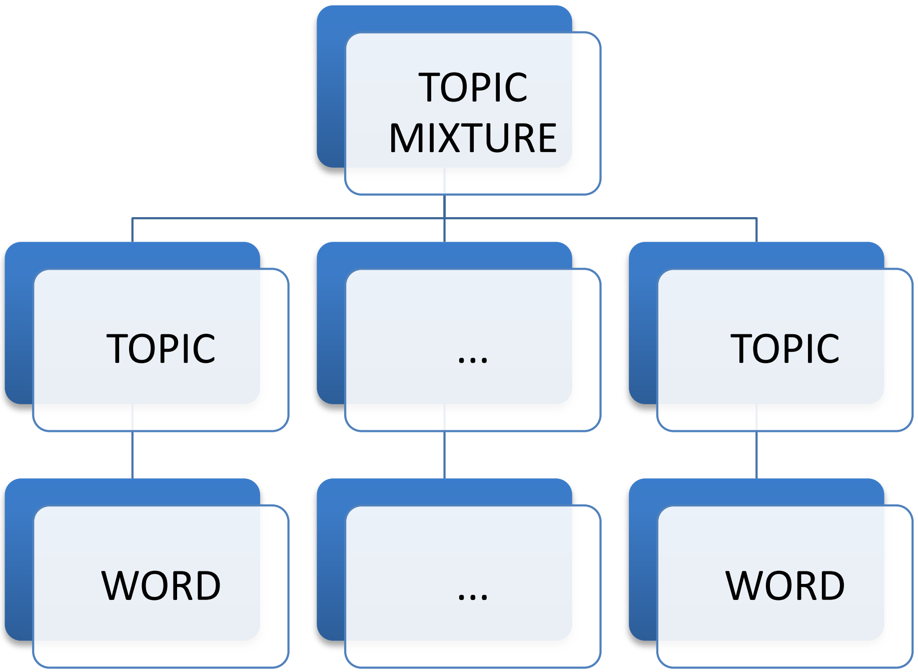
Latent Dirichlet Allocation (LDA)

❖ **Idea**: assume a document collection is (randomly) generated from a known set of topics (probabilistic generative model)



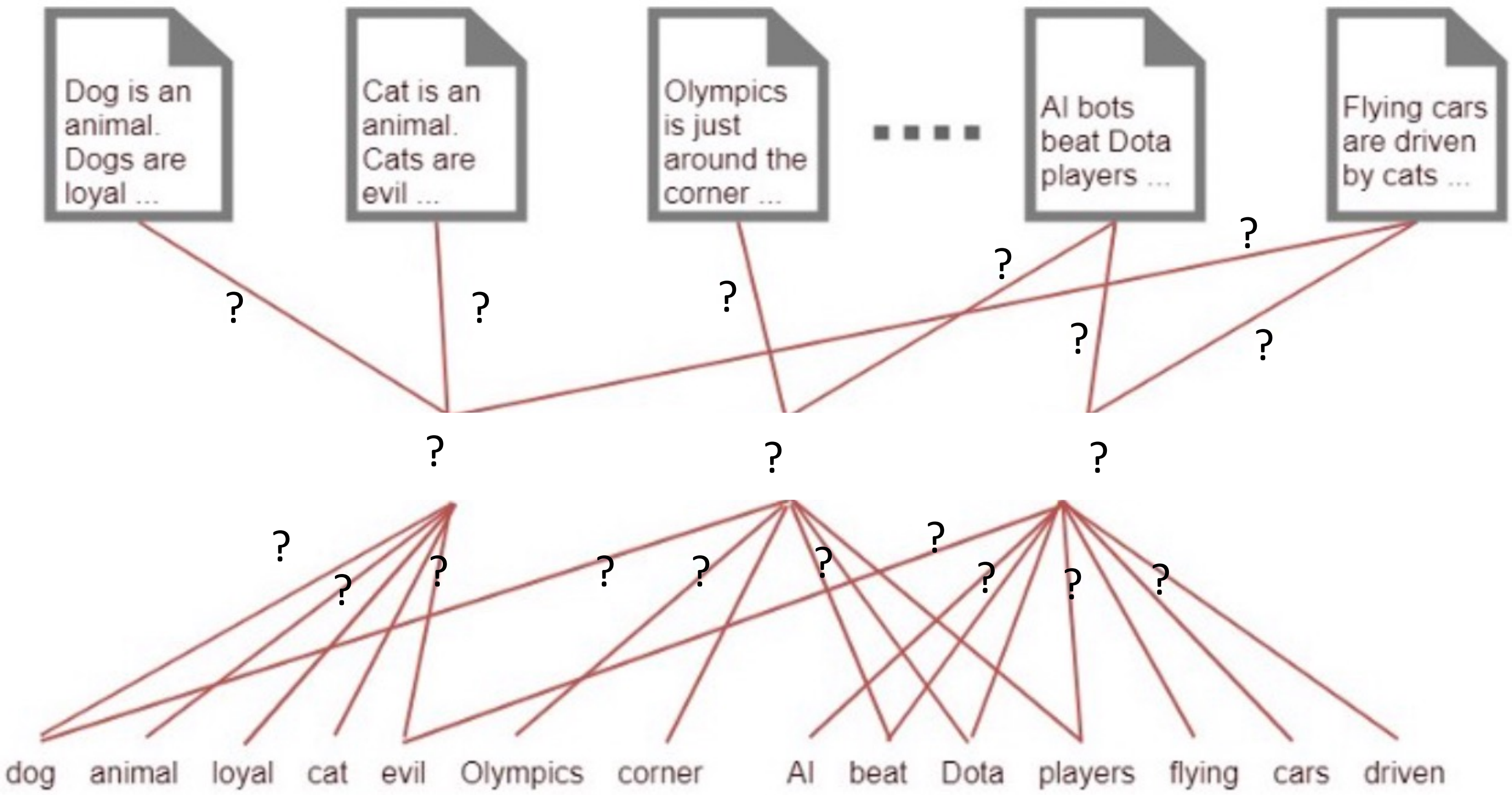
Document Generation using a Probabilistic Process

- For each document, choose a mixture of topics
- For every word position, sample a topic from the topic mixture
- For every word position, sample a word from the chosen topic



LDA: Topic Identification

- **Approach**: Inverting the process: given a document collection, reconstruct the topic model



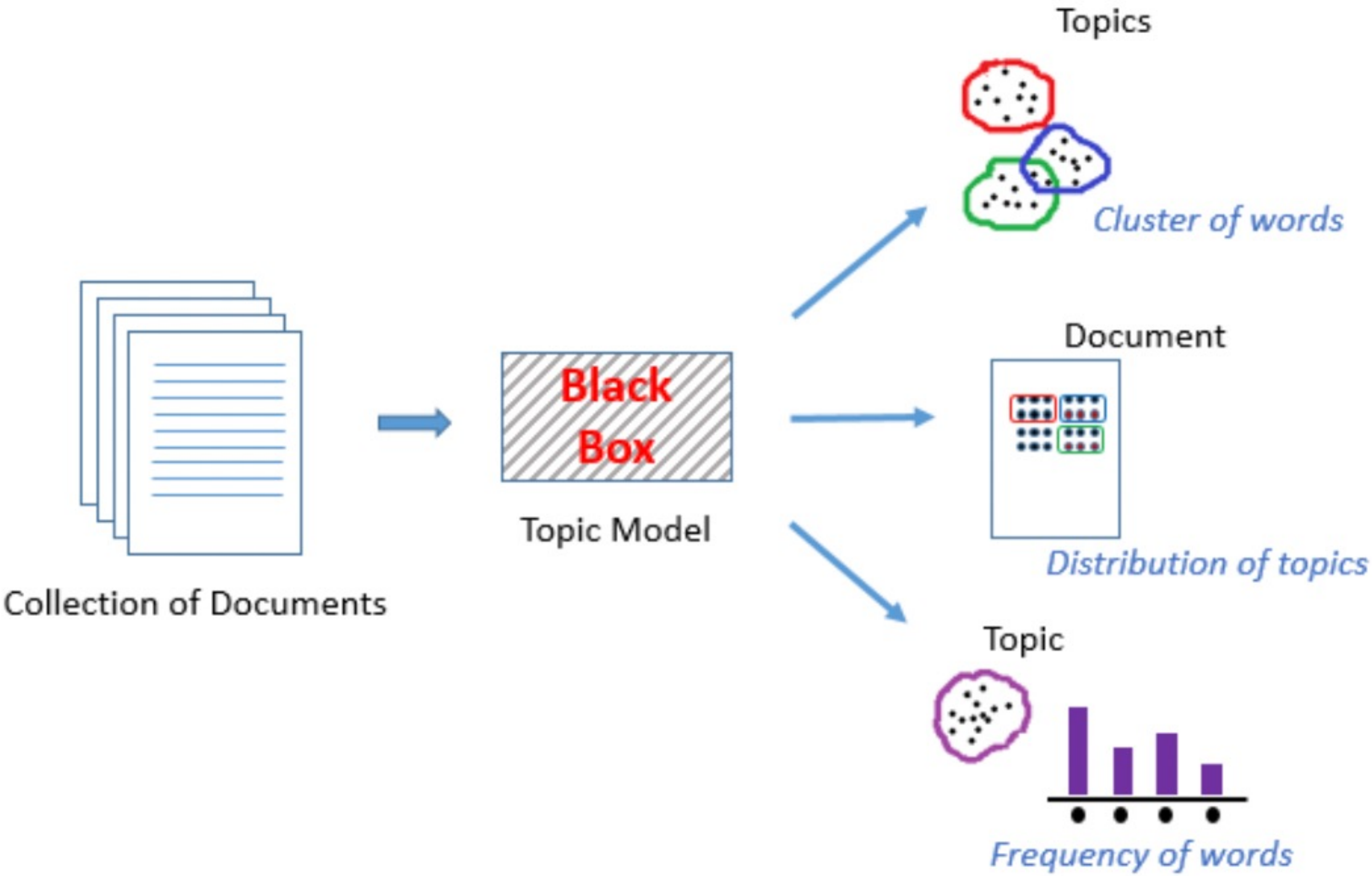
Latent Dirichlet Allocation

- Topics are **interpretable** unlike the arbitrary dimensions of LSI
- Distributions follow a Dirichlet distribution
- Construction of topic model is mathematically involved, but computationally feasible
- Considered as the state-of-the art method for topic identification

Use of Topic Models

- Unsupervised Learning of topics
  - Understanding main topics of a topic collection
  - Organizing the document collection
- Use for document retrieval: use topic vectors instead of term vectors to represent documents and queries
- Document classification (Supervised Learning): use topics as features

Topic Modeling: Summary



NAMED ENTITY RECOGNITION

Named Entity Recognition (NER)

Task: Find and classify names of people, organizations, places, brands etc. that are mentioned in documents

organization

The **United Nations (UN)** is an intergovernmental organization whose purpose is to maintain international peace and security, develop friendly relations among nations, achieve international cooperation, and be a centre for harmonizing the actions of nations. It is the world's largest and most f[illegible] place [illegible]national organization. The UN is headquar[illegible] place [illegible]ternational territory in **New York City**, and has other main offices in **Geneva, Nairobi, Vienna, and The Hague** (home to the International Court of Justice).

The UN was established after World War II with the aim of preventing future wars, succeeding the rather ineffective League of Nations. On 25 April 1945, 50 governments met in San Francisco for a conference and started drafting the UN Charter, which was adopted on 25 June 1945 and took effect on 24 October 1945, when the UN began operations. Pursuant to the Charter, the organization's objectives include maintaining international peace and security, protecting human rights, delivering humanitarian aid, promoting sustainable development, [illegible] place [illegible]ng international law. At its founding, the UN had 51 member states; with the addition of **South Sudan** in 2011, membership is now 193, representing almost all of the world's sovereign states.

Named Entity Recognition (NER)

Uses of NER

- Named entities can be indexed, linked, etc.
- Sentiment can be attributed to companies or products
- Information extraction can use named entities as anchors

Commercial tools available

- Reuters’ OpenCalais, AlchemyAPI (now IBM)

NER as Sequence Labelling Task

Sequence of tags, indicating whether a word is inside (I) or outside of an entity (O)

The occurrences of entities (can be) typed

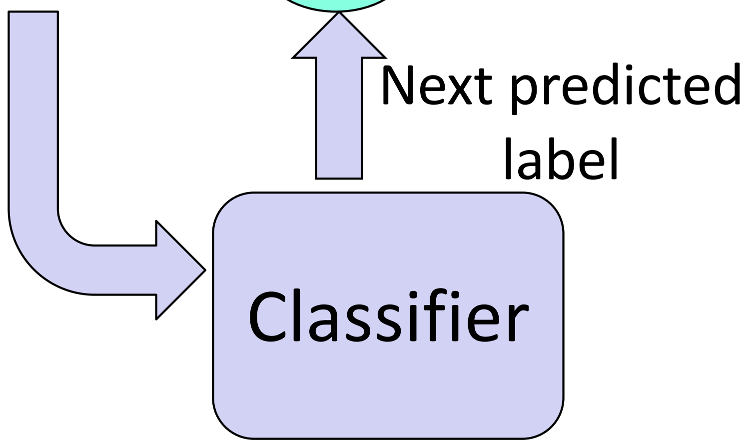
United Nation	is	headquartered	at	**New York City**	in	**USA**
I	O	O	O	I	O	I
ORG				GEO		GEO

A classification problem!

NER as Classification Task

United Nation is headquartered at **New York City** in **USA**

I O O O I O I



Features:

- Neighboring words
- Preceding labels

Naïve Bayes, HMM, MEMM, CRF, ...

Generative Probabilistic Model

Sequence of words (known): W = (w_1, w_2, w_3, …, w_n)

Sequence of states (unknown): E = (e_1, e_2, e_3, …, e_n)

Assume the text is produced by a probabilistic process:

$$P(E,W)$$

Find the most probable model

$$\underset{E}{\operatorname{argmax}} P(E|W)$$

Bayes Law

$$\underset{E}{\operatorname{argmax}} P(E|W) = \underset{E}{\operatorname{argmax}} P(E)P(W|E)$$

Approximation

Label transition probabilities (bigram model)

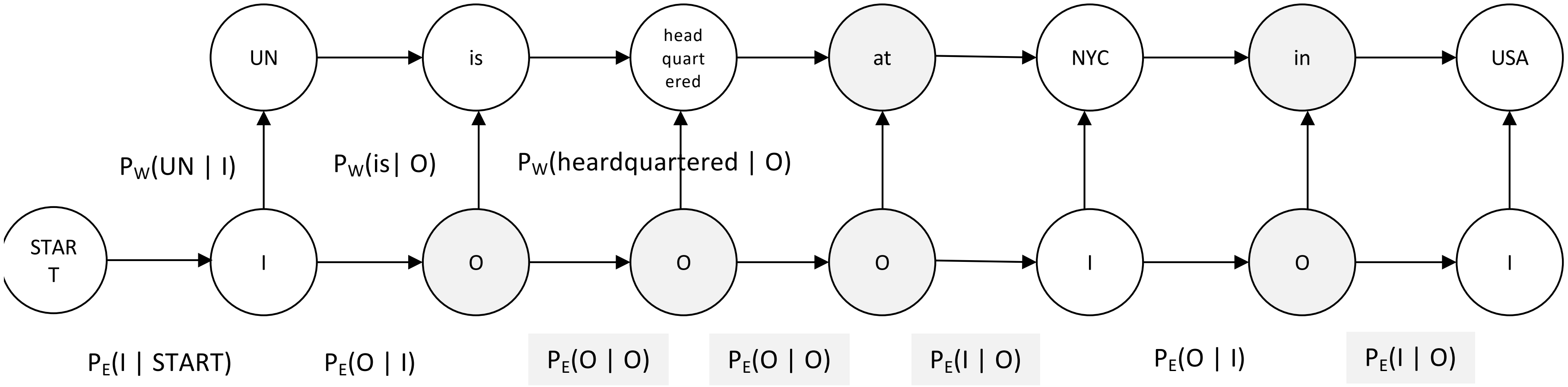
$$P(E) = P(e_1, \dots, e_n) \approx \prod_{i=2,\dots,n} P_E(e_i | e_{i-1})$$

Word emission probabilities

$$P(W|E) \approx \prod_{i=1,\dots,n} P_W(w_i | e_i)$$

Hidden Markov Model (HMM)

Assume the text is produced by a probabilistic process (with unknown transition probabilities)



Maximum Likelihood Estimation, e.g.,

$$P_E(I \mid O) = 2 / 4 \text{ , } P_W(\text{at} \mid O) = 1 / 4$$

Quiz: $P(W|E)$ is approximated by $P_E(w_i|e_{i-1})$ **and not using** $P_E(w_i|e_{i-1}, \dots, e_1)$

A. Because there is not enough data to estimate $P_E(w_i|e_{i-1}, \dots, e_1)$

B. Because it would not result in a Markov model

C. Because it is much less expensive to compute $P_E(w_i|e_{i-1})$

D. Because smoothing could not be applied

Summary

Topic Modeling:

- Latent Semantic Indexing (LSI) uses Singular Value Decomposition (SVD) to infer the importance of each word in a topic.
- Latent Dirichlet Allocation (LDA) uses Dirichlet generative process to infer topic as a distribution of words.

Named Entity Recognition:

- Classify a word as entity or not in a document
- Use Hidden Markov Model that takes into account the order of words

References

[1] Deerwester, Scott, et al. "Indexing by latent semantic analysis." Journal of the American society for information science 41.6 (1990): 391.

[2] Mikolov, Tomas, et al. "Distributed representations of words and phrases and their compositionality." Advances in neural information processing systems. 2013.

[3] Goldberg, Yoav, and Omer Levy. "word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method." arXiv preprint arXiv:1402.3722 (2014).

[4] http://videolectures.net/deeplearning2015_manning_language_vectors/