Text Mining

- Data stored in most text databases are **semistructured** data
- A document may contain:
  - **structured** fields: title, authors, publication date, category, and ……,
  - largely **unstructured** text components: abstract and contents…
  - Information retrieval techniques: text indexing methods (to handle **unstructured** documents)

Information Retrieval

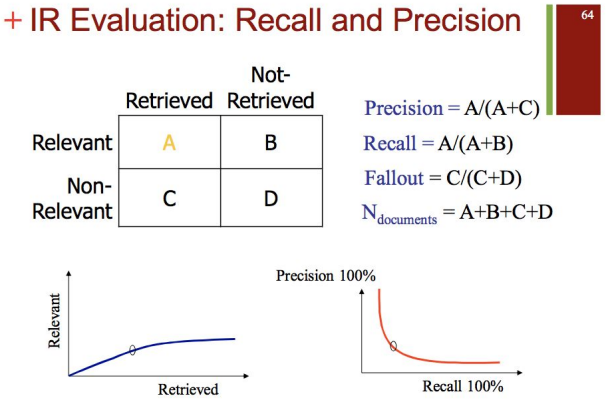
the science of searching for documents, for information within documents, and for metadata about documents, as well as that of searching relational databases and the World Wide Web.

Suppose that a text retrieval system has just retrieved a number of documents for me based on my input in the form of a query. How can we assess how accurate or correct the system was?

{Relevant}: the set of documents relevant to a query
{Retrieved}: the set of documents retrieved
{Relevant} ∩ {Retrieved}: the set of documents that are both relevant and retrieved

- Measure quality of text retrieval:
  - Precision: This is the percentage of retrieved documents that are in fact relevant to the query (i.e., “correct” responses)
  - Recall: This is the percentage of documents that are relevant to the query and were, in fact, retrieved. It is formally defined as
  - F-score: Trade-Off (harmonic mean of precision & recall)
  - *not sensitive* to the internal **ranking** of the documents in a retrieved set

$$precision = \frac{\{Relevant\}\cap\{Retrieved\}}{\{Retrieved\}} \quad recall = \frac{\{Relevant\}\cap\{Retrieved\}}{\{Relevant\}}$$



$$F - score = \frac{recall \times precision}{(recall + precision)/2}$$

- Measure the quality of a **ranked** list of documents:

- ➢ compute an average of **precisions** at all the ranks where a new relevant document is returned
- ➢ **Or**, plot a graph of **precisions** at many different levels of **recall**; a higher curve represents a better-quality information retrieval system.

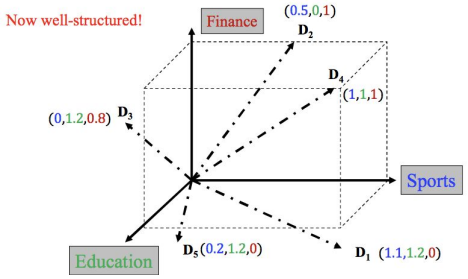
Nature Language Processing:

Part-of-speech Tagging	This sentence serves as an example. Det Noun Verb P Det Noun
Entity Recognition	The University of Queensland, St. Lucia Brisbane University Suburb City

Text Mining Challenges

- ❖ Data is not well-organized
- ❖ Natural language text contains ambiguities
- ❖ In traditional data mining: all data are “structured”

How to represent a document “structurally”?

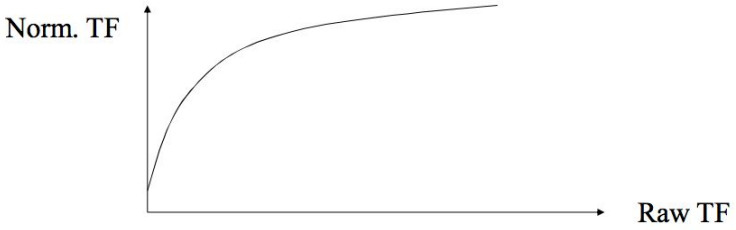
- ❖ Represent by a string?
  - ➢ No semantic meaning
- ❖ Represent by a list of sentences?
  - ➢ Sentence is just like a short document (recursive definition)
- ❖ Represent documents by concept vectors (**VSM: Vector Space Model**)
  - ➢ Each concept defines one dimension
  - ➢ k concepts define a k-dimensional space
  - ➢ A document is represented as a point in this k-dimensional space
    - ■ E.g., = (1, ... ,), is the coordinate of dimension in , and also the “importance” of concept in this document.
    - ■ 

VSM doesn't Say:

- How to define/select the "***basic concept***": Concepts are assumed to be **orthogonal**
  - Orthogonal:
    - Linearly independent basis vectors
    - "Non-overlapping"in meaning "sports" and "football"?
- How to assign ***weights***: Weights indicate **how well** the concept characterizes the document. (Set automatically and accurately)
  - Document-wise:
    - not all terms are equally important
  - Corpus-wise:
    - some terms carry more information about the document content
  - How to set weight? (2 basic heuristics)
    - TF (Term Frequency) = Within-doc-frequency
      - Idea: a term is more important if it occurs more frequently in a document
    - TF normalization (term frequency)
      - Two views of document length:
        - A doc is long because it is verbose / more content
      - Raw TF is inaccurate
        - Document length variation
        - "Repeated occurrences" are less informative than the "first occurrence"
        - Information about semantic does not increase proportionally with number of term occurrence
      - Generally penalize long document, but avoid over-penalizing
        - Pivoted length normalization

Sub-linear TF scaling

$$tf(t,d) = \begin{cases} 1 + \log c(t,d), if\ c(t,d) > 0 \\ 0, otherwise \end{cases}$$



- IDF (Inverse Document Frequency)

$$IDF(word) = \log \frac{1 + \text{total documents}}{\text{document frequency}}$$

- Assign higher weights to the rare terms
- Formula
  - $IDF(t) = \log(\frac{N+1}{df(t)})$ — N: Total number of docs in collection; $df(t)$: Number of docs containing term t
- A corpus-specific property
  - Independent of a single document

- TF-IDF weighting
  - Combining TF and IDF
    - Common in doc → high tf → high weight
    - Rare in collection→ high idf→ high weight
    - $w(t,d) = TF(t,d) \times IDF(t)$

❖ How to define the ***distance metric***: What kind of documents are **similar**?

➢ 2 approaches: **Edit Distance** && **Cosine**

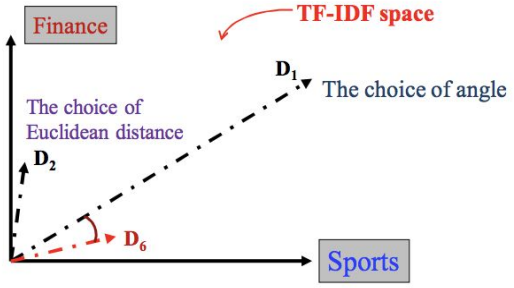
- Euclidean distance
  - $dist(d_i, d_j) = \sqrt{\sum_{t \in V}[tf(t,d_i)idf(t) - tf(t,d_j)idf(t)]^2}$
  - Longer documents will be penalized by the extra words
  - We care more about how these two vectors are overlapped

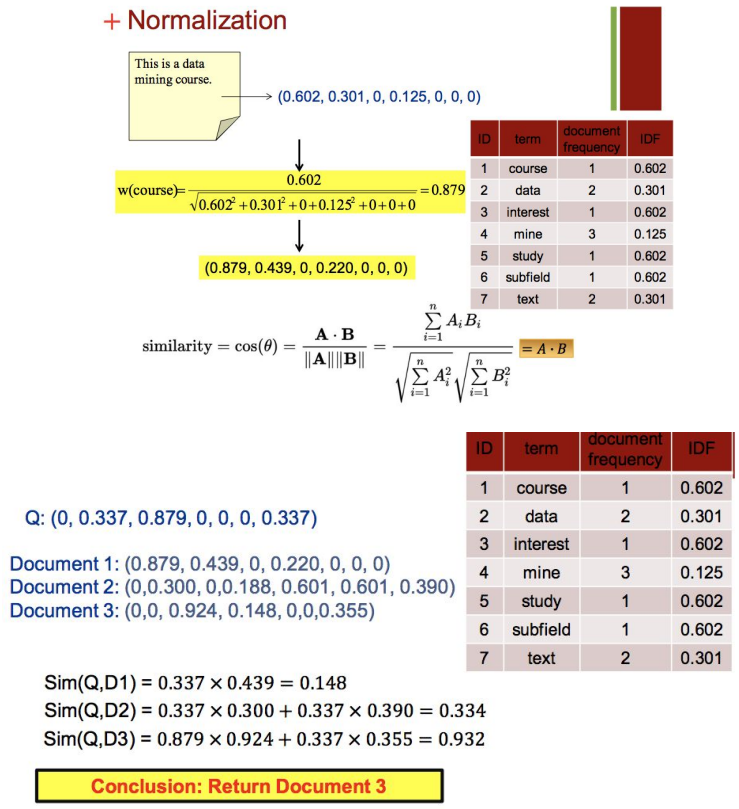
➢

- Angle: how vectors are overlapped

➢

- Cosine similarity – projection of one vector onto another





VSM existing Solutions:

- ❖ Topic modelling
- ❖ Bag-of-Words model or N-gram model
  - ➢ Bag-of-Words (**BOW**): Each distinct word represents a concept (most frequently used **document representation**: Image, speech, gene sequence)
  - ➢ **Tokenization**:Break texts into meaningful words
    - ■ Tokens: words, phrases, symbols. (Definition depends on language, corpus, or even context)
    - ■ Rule-based
      - so-called -> ‘so’, ‘called’
      - It’s -> ‘It’s’ instead of ‘It’, ‘’s’
    - ■ Statistical Method
      - ‘San Francisco’ instead of ‘San’, ‘Francisco’
    - ■ Existing Tool (eg. NLTK)
  - ➢ **N-grams** as concepts: a contiguous sequence of N tokens from a given piece of text

Bag-of-Words Advantages Vs. Disadvantages

Advantages	Disadvantages
• Simple	• Words are clearly not linearly independent!

	• Grammar and order are missing

N-grams Advantages Vs. Disadvantages

Advantages	Disadvantages
• capture local dependency and order	• A purely statistical view, increase the vocabulary size $O(V^N)$

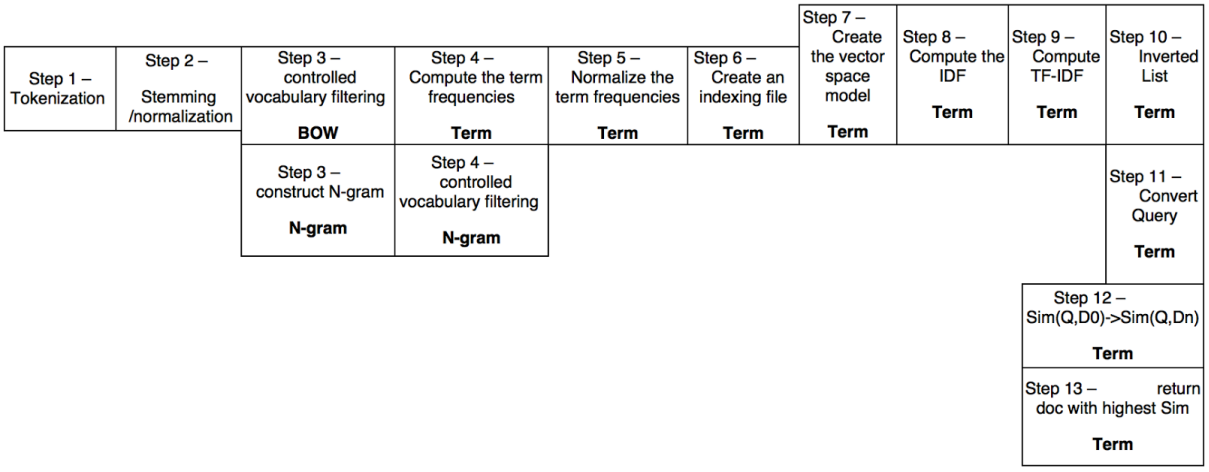
Document Representation Advantages Vs. Disadvantages

Advantages	Disadvantages
• Preserve all information in the text (hopefully) • Fully automatic	• Vocabulary gap: cars v.s., car, talk v.s., talking • Large storage:N-grams needs $O(V^N)$ ○ Solution: Construct controlled vocabulary

- ❖ **Normalisation**
  - ➢ Convert different forms of a word to a normalized form in the vocabulary
    - ■ UQ -> University of Queensland, St. Lucia -> Saint Lucia
  - ➢ Rule-based
    - ■ Delete periods “.” and hyphens ” - ”
    - ■ All in **lower cases**
    - ■ Change it to original form
      - • interesting -> interest
      - • Mining -> mine
  - ➢ Dictionary-based
    - ■ Construct equivalent class
      - • Car->“automobile,vehicle”
      - • UQ->“UniversityofQueensland”
      - • Mobile phone -> “cellphone”
- ❖ **Stemming**
  - ➢ Reduce inflected or derived words to their root form
    - ■ Plurals, adverbs, inflected word forms
      - • E.g., ladies -> lady, referring -> refer, forgotten -> forget
    - ■ Reduce the vocabulary size
    - ■ Existing tools (for English)
  - ➢ Risk: lose precise meaning of the word
- ❖ **Controlled vocabulary**
  - ➢ Remove non-informative & rare words

- **Zipf's law**
  - If the word ranked 1st has frequency 10000, what is the frequency of the word ranked 20-th ?
  - f(1) /1= f(20)/(1/20)→ f(20) = f(1) /20=500
  - Head words take large portion of occurrences, but they are semantically meaningless
    - E.g., the, a, an, we, do, to, in, with, of
  - Tail words take major portion of vocabulary, but they rarely occur in documents
    - E.g., dextrosinistral
  - The rest is more representative
    - To be included in the controlled vocabulary

Text Mining Tasks:



Query A Document

- similar to the previous steps:
  - 1. Tokenization/Stemming.
  - 2. Controlled vocabulary filtering.
  - 3. Transform the query string into a vector space model (VSM) by using TD-IDF schema.
  - 4. Normalize the VSM into unit length.

- Improving Query efficiency:
  - Inverted list
    - Query Q="interesting course"
  - Only calculate the doc exist in the invert list; By Sim(Q,Dn)

Vector Space Model Advantages Vs. Disadvantages

Advantages	Disadvantages

• Empirically effective!
• Intuitive
• Easy to implement
• Well-studied/mostly evaluate
• The Smart system
  ○ Developed at Cornell: 1960-1999
  ○ Still widely used
• Warning: many variants of TF-IDF!
	• Assume term independence
	• Lack of “predictive adequacy”
	  ○ Arbitrary term weighting
	  ○ Arbitrary similarity measure
	• Lots of parameter tuning!