COM3110/4115/6115: Text Processing

Information Retrieval: retrieval models

Mark Stevenson

Department of Computer Science
University of Sheffield

Overview

- Definition of the information retrieval problem
- Approaches to document indexing
  - ◇ manual approaches
  - ◇ automatic approaches
- **Automated retrieval models**
  - ◇ **boolean model**
  - ◇ **ranked retrieval methods (e.g. vector space model)**
- Term manipulation:
  - ◇ stemming, stopwords, term weighting
- Evaluation

Bag-of-Words Approach

- Standard approach to representing documents (and queries) in IR:
  - ◇ record what words (terms) are present
  - ◇ usually, plus count of term in each document

- Ignores relations between words
  - ◇ i.e. of order, proximity, etc
  - ◇ e.g. rabbit eating = eating rabbit

- Such representations known as **bag of words** approaches
  - ◇ c.f. mathematical structure "bag"
    — like a set (i.e. unordered), but records a count for each element

Information Retrieval: Methods

- Boolean search:
  - binary decision: is document relevant or not?
  - presence of term is necessary and sufficient for match
  - boolean operators are set operations (AND, OR)
- Ranked algorithms:
  - frequency of document terms
  - not all search terms necessarily present in document
  - Incarnations:
    - **The vector space model (SMART, Salton et al, 1971)**
    - The probabilistic model (OKAPI, Robertson/Spärck Jones, 1976)
    - Web search engines

The Boolean model

- Approach: construct *complex search commands*, by
  - ◇ combining *basic* search terms (keywords)
  - ◇ using *boolean operators*
- *Boolean Operators*:
  - ◇ AND, OR, NOT, BUT, XOR (*exclusive* OR)
- E.g.:
  `Monte-Carlo AND (importance OR stratification) BUT gambling`
- Boolean query provides a simple logical basis for deciding whether any document should be returned, based on:
  - ◇ whether basic terms of query do/do not appear in the document
  - ◇ the meaning of the logical operators

The Boolean model: set-theoretic interpretation

- Boolean operators have a **set-theoretic interpretation** for **efficient** retrieval

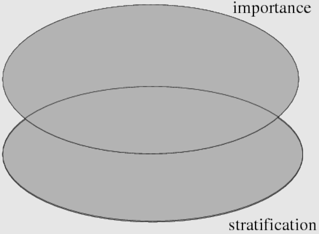
- Overall document collection forms maximal document set

- let $d(E)$ denote the document set for expression E
    - ◇ E either a basic term or boolean expression

- Boolean operators map to set-theoretic operations:
    - ◇ AND $\mapsto \cap$ (intersection): $d(E_1 \text{ AND } E_2) = d(E_1) \cap d(E_2)$
    - ◇ OR $\mapsto \cup$ (union): $d(E_1 \text{ OR } E_2) = d(E_1) \cup d(E_2)$
    - ◇ NOT $\mapsto {}^c$ (complement): $d(\text{NOT } E) = d(E)^c$
    - ◇ BUT $\mapsto -$ (difference): $d(E_1 \text{ BUT } E_2) = d(E_1) - d(E_2)$

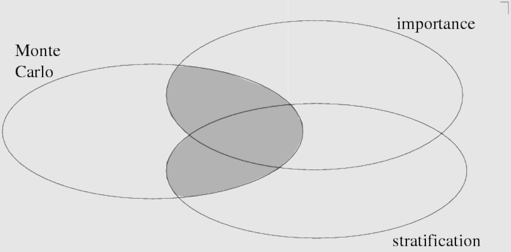
The Boolean model: set-theoretic interpretation (contd)

E.g. `Monte-Carlo AND (importance OR stratification) BUT gambling`



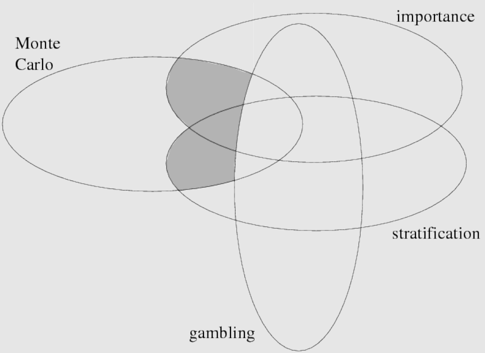
The Boolean model: set-theoretic interpretation (contd)

E.g. Monte-Carlo AND (importance OR stratification) BUT gambling



The Boolean model: set-theoretic interpretation (contd)

E.g. Monte-Carlo AND (importance OR stratification) BUT gambling



Boolean Queries: Complexity

- Question: **Magnetic resonance imaging, magnetic resonance arthrography and ultrasonography for assessing rotator cuff tears in people with shoulder pain for whom surgery is being considered**

- Query: `((Ultrasonography [mh] OR ultrasound [tw] OR ultrasonograph* [tw] OR sonograp*[tw] OR us [sh]) OR (Magnetic Resonance Imaging [mh] OR MR imag*[tw] OR magnetic resonance imag* [tw] OR MRI [tw])) AND (Rotator Cuff [mh] OR rotator cuff* [tw] OR musculotendinous cuff* [tw] OR subscapularis [tw] OR supraspinatus [tw] OR infraspinatus OR teres minor [tw]) AND (Rupture [mh:noexp] OR tear* [tw] OR torn [tw] OR thickness [tw] OR lesion* [tw] OR ruptur* [tw] OR injur* [tw])`

From Lenza, M., Buchbinder, R., Takwoingi, Y., Johnston, R. V., Hanchard, N. C., & Faloppa, F. (2013). Magnetic resonance imaging, magnetic resonance arthrography and ultrasonography for assessing rotator cuff tears in people with shoulder pain for whom surgery is being considered. The Cochrane Library.

The Boolean model: summary

- Documents either match or don't match
  - ◇ Expert knowledge needed to create high-precision queries → OK for expert users
  - ◇ Often used by bibliographic search engines (library)
- Not good for the majority of users
  - ◇ Most users not familiar with writing Boolean queries → not natural
  - ◇ Most users don't want to wade through 1000s unranked result lists → unless very specific search in small collections
  - ◇ This is particularly true of web search → large set of docs

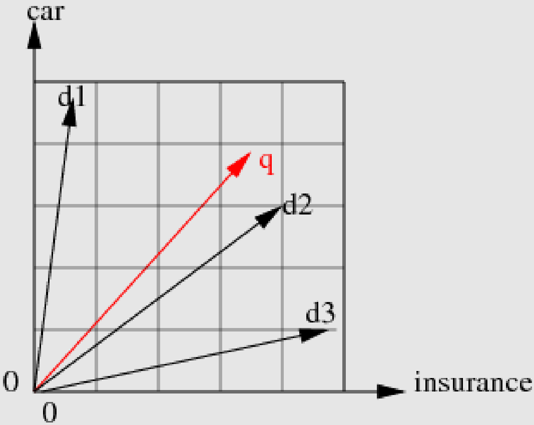
The Vector Space model

- Documents are also represented as "bags of words":
  - "John is quicker than Mary" = "Mary is quicker than John"
- Documents are points in high-dimensional vector space
  - each term in index is a dimension → sparse vectors
  - values are frequencies of terms in documents, or variants of frequency
- Queries are also represented as vectors (for terms that exist in index)
- Approach
  - Select document(s) with highest document–query similarity
  - Document–query similarity is a model for relevance (ranking)
  - With ranking, the number of returned documents is less relevant → users start at the top and stop when satisfied

The Vector Space model (contd)

2 dimensions:

Query: car insurance



The Vector Space Model (contd)

- Approach: compare vector of query against vector of each document
  - ◇ to rank documents according to their similarity to the query

	Term$_1$	Term$_2$	Term$_3$	...	Term$_n$
Doc$_1$	9	0	1	...	0
Doc$_2$	0	1	0	...	10
Doc$_3$	0	1	0	...	2
...	...	...	...	...	...
Doc$_N$	4	7	0	...	5

Q	0	1	0	...	1

How to measure similarity between vectors?

- Each document and the query are represented as a vector of n values:

$$\vec{d^i} = (d_1^i, d_2^i, \ldots, d_n^i), \qquad \vec{q} = (q_1, q_2, \ldots, q_n)$$

- Many metrics of similarity between 2 vectors, e.g.: Euclidean

$$\sqrt{\sum_{k=1}^{n}(q_k - d_k)^2}$$

- E.g.: Distance between:

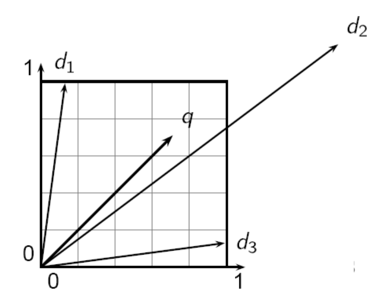
$$Doc_1 \text{ and } Q = \sqrt{(9-0)^2 + (0-1)^2 + (1-0)^2 + (0-1)^2} = \sqrt{84} = 9.15$$
$$Doc_2 \text{ and } Q = \sqrt{(0-0)^2 + (1-1)^2 + (0-0)^2 + (10-1)^2} = \sqrt{81} = 9$$
$$Doc_3 \text{ and } Q = \sqrt{(0-0)^2 + (1-1)^2 + (0-0)^2 + (2-1)^2} = \sqrt{1} = 1$$

Doc 3 is the closest (shortest distance)

How to measure similarity between vectors? (contd)

Is it a good idea?

- Distance is large for vectors of different lengths, even if by only one term (e.g. Doc_2 and Q)
- Frequency of terms **overweighted**



How to measure similarity between vectors? (contd)

- Better similarity metric, used in *vector-space* model: **cosine** of the angle between two vectors $\vec{x}$ and $\vec{y}$:

$$cos(\vec{x},\vec{y}) = \frac{\vec{x}\cdot\vec{y}}{|\vec{x}||\vec{y}|} = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}$$

- It can be interpreted as the normalised correlation coefficient:

  i.e. it computes how well the x_i and y_i correlate, and then divides by the length of the vectors, to scale for their magnitude

  ◇ The vector $\vec{x}$ is normalised by dividing its components by its length:

$$|\vec{x}| = \sqrt{\sum_{i=1}^{n} x_i^2}$$

How to measure similarity between vectors? (contd)

- The cosine value ranges from:
  - ◇ 1, for vectors pointing in the same direction, to
  - ◇ 0, for orthogonal vectors, to
  - ◇ -1, for vectors pointing in opposite directions

- Specialising the equation to comparing a query *q* and document *d*:

$$sim(\vec{q},\vec{d}) = cos(\vec{q},\vec{d}) = \frac{\sum_{i=1}^{n} q_i d_i}{\sqrt{\sum_{i=1}^{n} q_i^2}\sqrt{\sum_{i=1}^{n} d_i^2}}$$

i.e. computes how well occurrences of each term *i* correlate in query and document, then scales for the magnitude of the overall vectors

Summary

- Automated Retrieval Models
  - ◇ Boolean Model
  - ◇ Vector Space Model
- Next time
  - ◇ What counts as a term?
  - ◇ How are terms weighted?