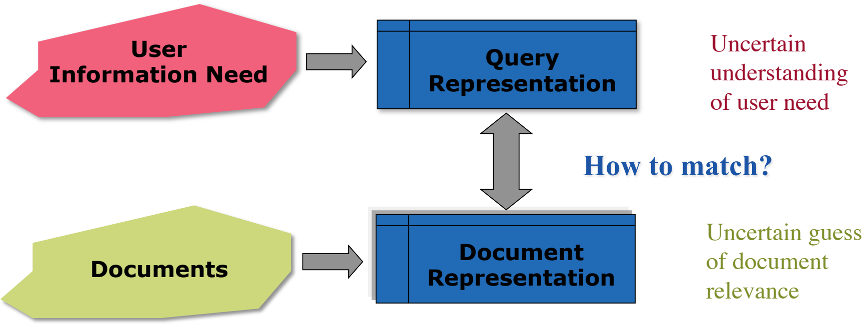
DD2476 Search Engines and Information Retrieval Systems Lecture 7: Probabilistic Information Retrieval, Language Models

Hedvig Kjellström
hedvig@kth.se
www.csc.kth.se/DD2476

Summary Vector Space Model

- Represent the query as a weighted tf-idf vector
- Represent each document as a weighted tf-idf vector
- Compute the cosine similarity score for the query vector and each document vector
- Rank documents with respect to the query by score
- Return the top *K* (e.g., *K* = 10) to the user

Only Simplified Model of Reality



Probabilistic Methods

- Traditionally in Artificial Intelligence, Information Retrieval, Machine Learning, Computer Vision, etc:
  - Probabilistic methods "the right way to do it"
  - Mathematically sound
- Probabilistic approaches a current hot topic
  - Old approach that has revival
  - Computationally expensive – not a big problem anymore

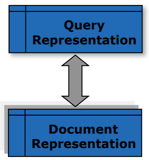
Today

- Probabilistic information retrieval (Manning Chapter 11)
  - Probability ranking principle
  - Binary independence model
- Language models (Manning Chapter 12)
  - Query likelihood model
  - Language model vs binary independence model

Probabilistic Information Retrieval

(Manning Chapter 11)

The Document Ranking Problem



- Have collection of documents
- User issues query
- Return list of documents
- Ranking method core of IR system:
  - Want "best" document first, second best second, etc….

Lecture 4

- Idea 1: Rank according to query-document similarity
  - CosineSim(tf-idf(document),tf-idf(query))

Today!

- Idea 2: Rank according to probability of document being relevant w.r.t. information need
  - p(relevant|document,query)

Bayes' Rule

$$p(A \mid D) = \frac{p(D \mid A)\, p(A)}{p(D)}$$

A is a Boolean-valued random variable (e.g. the document is relevant)

- p(A) = prior probability of hypothesis A – PRIOR
- p(D) = prior probability of data D – EVIDENCE
- p(D|A) = probability of D given A - LIKELIHOOD
- P(A|D) = probability of A given D – POSTERIOR

- Boolean hypotheses: Odds

$$O(A \mid D) = \frac{p(A \mid D)}{p(\neg A \mid D)} = \frac{p(A \mid D)}{1 - p(A \mid D)}$$

Probability Ranking Principle (PRP)

- Long version, van Rijsbergen (1979:113-114):
  - "If a reference retrieval system's response to each request is a ranking of the documents in the collection in order of decreasing probability of relevance to the user who submitted the request, where the probabilities are estimated as accurately as possible on the basis of whatever data have been made available to the system for this purpose, the overall effectiveness of the system to its user will be the best that is obtainable on the basis of those data."
- Short version:
  - Have representative training document collection and training queries, learn p(relevant|document,query)
  - Ranking of new documents w.r.t. new queries (similar to, or in, the training collection) with this function is bound to better than anything else

Task 1.4, 2.3

Probability Ranking Principle (PRP)

- Let *d* be a document in the collection
  - *R* represents relevance of doc w.r.t. given (fixed) query
  - *NR* represents non-relevance of doc w.r.t. given (fixed) query

Relevance is binary variable with values {NR,R}

- Need to find p(*R*|*d*,*q*) - probability that a document *d* is relevant given query *q*
  - This distribution lives in a HUGE space
- How can this be done?
  - Reformulate using Bayes' rule to distributions easier to learn

Probability Ranking Principle (PRP)

$$p(R \mid d,q) = \frac{p(d \mid R,q)\,p(R \mid q)}{p(d \mid q)}$$

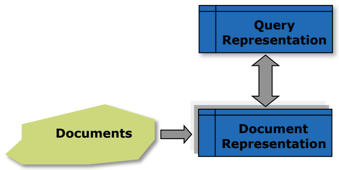
$$p(NR \mid d,q) = \frac{p(d \mid NR,q)\,p(NR \mid q)}{p(d \mid q)}$$

$$p(R \mid d,q) + p(NR \mid d,q) = 1$$

$$O(R \mid d,q) = \frac{p(R \mid d,q)}{p(NR \mid d,q)}$$

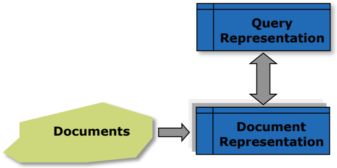
- p(*R*|*q*), p(*NR*|*q*) - prior probability of retrieving a (non-)relevant document given query *q*
- PRP in action: Rank all documents by p(*R*|*d*,*q*) or O(*R*|*d*,*q*)

Probability Ranking Principle (PRP)



- How compute p(*d*|*R*,*q*), p(*d*|*NR*,*q*), p(*R*|*q*), p(*NR*|*q*) ?
  - Do not know exact probabilities, have to use estimates
  - I.e., use simplified model of reality
  - Binary Independence Model (BIM) – simplest model
- Questionable assumptions
  - More later

Binary Independence Model (BIM)

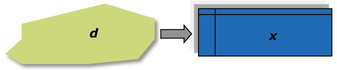


- Traditionally used with PRP
- "Binary" = Boolean: Documents *d* are boolean vectors $\vec{x} = (x_1, \ldots, x_n)$ where $x_i = 1$ iff term *i* is present in document *x*
- "Independence": terms occur in documents independently
  - Different documents can be modeled as same vector
- Bernoulli Naive Bayes model

Lecture 1

Binary Independence Model (BIM)

- Documents *d*: binary vectors $\vec{x} = (x_1, \ldots, x_n)$
- (Queries *q*: also binary vectors $\vec{q} = (q_1, \ldots, q_n)$)
- Given query *q*,
  - for each document *d*, compute p(*R*|*q*,*d*)
  - Equivalent to computing p(*R*|*q*,*x*) where *x* represents *d*
  - Interested only in ranking



- Will use odds and Bayes' Rule:

$$O(R|\vec{q},\vec{x}) = \frac{p(R|\vec{q},\vec{x})}{p(NR|\vec{q},\vec{x})} = \frac{\dfrac{p(R|\vec{q})p(\vec{x}|R,\vec{q})}{p(\vec{x}|\vec{q})}}{\dfrac{p(NR|\vec{q})p(\vec{x}|NR,\vec{q})}{p(\vec{x}|\vec{q})}}$$

Binary Independence Model (BIM)

$$O(R|\vec{q},\vec{x}) = \frac{p(R|\vec{q},\vec{x})}{p(NR|\vec{q},\vec{x})} = \frac{p(R|\vec{q})}{p(NR|\vec{q})} \cdot \frac{p(\vec{x}|R,\vec{q})}{p(\vec{x}|NR,\vec{q})}$$

Constant for a given query

Needs estimation for every document

Using independence assumption:

$$\frac{p(\vec{x}|R,\vec{q})}{p(\vec{x}|NR,\vec{q})} = \prod_{i=1}^{n} \frac{p(x_i|R,q_i)}{p(x_i|NR,q_i)}$$

So: $$O(R|q,d) = O(R|q) \cdot \prod_{i=1}^{n} \frac{p(x_i|R,q_i)}{p(x_i|NR,q_i)}$$

Binary Independence Model (BIM)

$$O(R|q,d) = O(R|q) \cdot \prod_{i=1}^{n} \frac{p(x_i|R,q_i)}{p(x_i|NR,q_i)}$$

Since x_i are either 0 or 1:

$$O(R|q,d) = O(R|q) \cdot \prod_{x_i=1} \frac{p(x_i=1|R,q_i)}{p(x_i=1|NR,q_i)} \cdot \prod_{x_i=0} \frac{p(x_i=0|R,q_i)}{p(x_i=0|NR,q_i)}$$

Let $p_i = p(x_i=1|R,q_i)$; $r_i = p(x_i=1|NR,q_i)$;

True positive

False positive

Assume, for all terms not occurring in the query (q_i=0): $p_i = r_i$

This can be changed (e.g., in relevance feedback)

Then...

$p_i = p(x_i = 1 | R, q_i);$
$r_i = p(x_i = 1 | NR, q_i);$

Binary Independence Model (BIM)

$$O(R|q,\vec{x}) = O(R|q) \cdot \prod_{x_i=q_i=1} \frac{p_i}{r_i} \cdot \prod_{\substack{x_i=0\\ q_i=1}} \frac{1-p_i}{1-r_i}$$

All matching terms

Non-matching query terms

$$= O(R|q) \cdot \prod_{x_i=q_i=1} \frac{p_i(1-r_i)}{r_i(1-p_i)} \cdot \prod_{q_i=1} \frac{1-p_i}{1-r_i}$$

All matching terms

All query terms

$p_i = p(x_i = 1 | R, q_i);$
$r_i = p(x_i = 1 | NR, q_i);$

Binary Independence Model (BIM)

$$O(R|q,\vec{x}) = O(R|q) \cdot \prod_{x_i=q_i=1} \frac{p_i(1-r_i)}{r_i(1-p_i)} \cdot \prod_{q_i=1} \frac{1-p_i}{1-r_i}$$

Constant for each query

Only quantity to be estimated for rankings

Retrieval Status Value:

$$RSV = \log \prod_{x_i=q_i=1} \frac{p_i(1-r_i)}{r_i(1-p_i)} = \sum_{x_i=q_i=1} \log \frac{p_i(1-r_i)}{r_i(1-p_i)}$$

$p_i = p(x_i = 1 | R, q_i);$
$r_i = p(x_i = 1 | NR, q_i);$

Binary Independence Model (BIM)

All boils down to computing RSV

$$RSV = \log \prod_{x_i=q_i=1} \frac{p_i(1-r_i)}{r_i(1-p_i)} = \sum_{x_i=q_i=1} \log \frac{p_i(1-r_i)}{r_i(1-p_i)}$$

$$RSV = \sum_{x_i=q_i=1} c_i; \qquad c_i = \log \frac{p_i(1-r_i)}{r_i(1-p_i)}$$

So, how do we compute c_i's from our data ?

$p_i = p(x_i = 1 | R, q_i);$
$r_i = p(x_i = 1 | NR, q_i);$

Binary Independence Model (BIM)

Estimating RSV coefficients from document training set

For each term *i* look at this table of document counts:

Documents	Relevant	Non-Relevant	Total
x_i=1	s	$n-s$	n
x_i=0	$S-s$	$N-n-S+s$	$N-n$
Total	S	$N-S$	N

- Estimates: $p_i \approx \frac{s}{S}$ $\quad r_i \approx \frac{(n-s)}{(N-S)}$

$$c_i \approx K(N,n,S,s) = \log \frac{s/(S-s)}{(n-s)/(N-n-S+s)}$$

For now, assume no zero terms. See book page 208.

Binary Independence Model (BIM)

$p_i = p(x_i = 1 | R, q_i);$
$r_i = p(x_i = 1 | NR, q_i);$

Lecture 4

- $S << (N - S)$, s << (n – s)
  - $r_i \approx n/N$
  - $\log (1 - r_i)/r_i \approx \log (N - n)/n$
- $n << (N - n)$
  - $\log (N - n)/n \approx \log N/n$ = idf
- Theoretical justification for the idf score!
  - Bayes-optimal

Exercise 5 Minutes

- Possible to get reasonable approximations of probabilities…

  …BUT it requires restrictive assumptions.
- Discuss in pairs/groups:
  - What are the assumptions made by PRP and BIM?

Exercise 5 Minutes

- Possible to get reasonable approximations of probabilities…

  …BUT it requires restrictive assumptions:
  - term independence (term order, term co-occurrence)
  - terms not in query don't affect the outcome
  - boolean representation of documents/queries/relevance
  - document relevance values are independent
    - Really, it's bad to keep on returning duplicates
- Some of these assumptions can be removed
  - Trade-offs

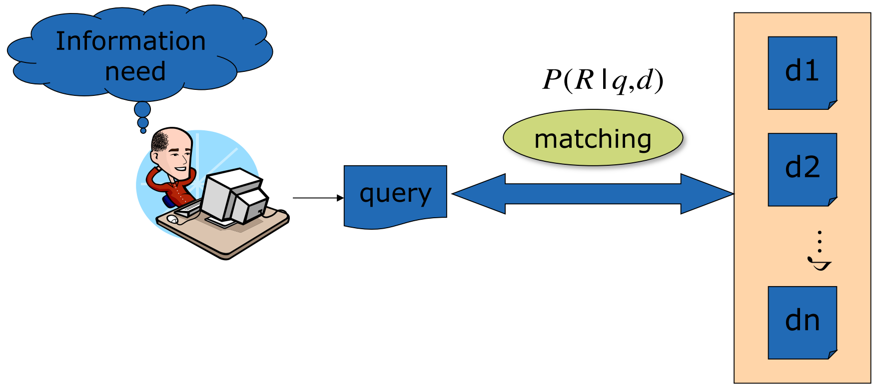
Summary Comparison

- Standard Vector Space Model
  - \+ Empirical for the most part; success measured by results
  - − Few properties provable
- Probabilistic Model
  - \+ Based on a firm theoretical foundation
  - \+ Theoretically justified optimal ranking scheme
  - − Binary word-in-doc weights (not using term frequencies)
  - − Independence of terms (can be alleviated)
  - − Amount of computation
  - − Has never worked convincingly better in practice

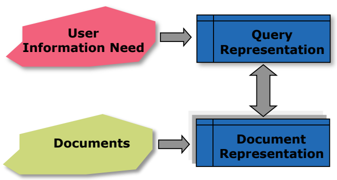
Language Models

(Manning Chapter 12)

Standard Probabilistic IR

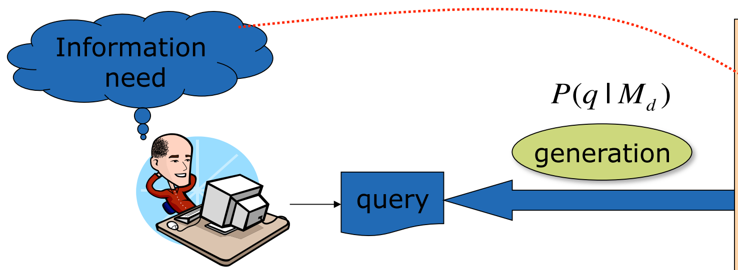


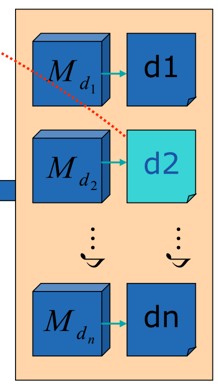
Dealing with the Query Part…



- But how does user generate query based on information need?

IR Based on Language Model (LM)





- Common search heuristic: form query from words you expect to find in matching documents
- Language Model approach models this process

document collection

IR Based on Language Model (LM)

- Model user's generation of queries as a stochastic process
  - Assumption: User choosing queries in methodical way but with some noise, based on what documents they are looking for
- Approach:
  - Infer a LM M_d for each document d
  - Estimate $p(q|M_d)$ for each document d
  - Rank the documents according to $p(q|M_d)$
    - High $p(q|M_d)$ – probable that M_d generated q

Stochastic Language Models

- Model probability of generating string s in the language M:

Model *M*

0.2	the
0.1	a
0.01	man
0.01	woman
0.03	said
0.02	likes

the		man		likes		the		woman
0.2	*	0.01	*	0.02	*	0.2	*	0.01

$$p(s|M) = 0.00000008$$

Stochastic Language Models

- Compare languages/models

Model *M1*

0.2	the
0.01	class
0.0001	sayst
0.0001	pleaseth
0.0001	yon
0.0005	maiden
0.01	woman

Model *M2*

0.2	the
0.0001	class
0.03	sayst
0.02	pleaseth
0.1	yon
0.01	maiden
0.0001	woman

the		class		pleaseth		yon		maiden
0.2	*	0.01	*	0.0001	*	0.0001	*	0.0005
0.2	*	0.0001	*	0.02	*	0.1	*	0.01

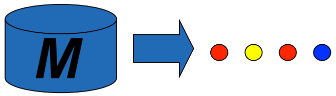
$$p(s|M2) > p(s|M1)$$

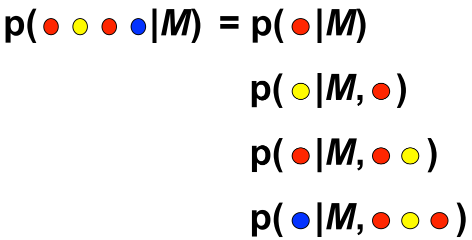
Exercise 2 Minutes

- Stochastic language model:
  - Probability of each word in language
  - Multiply probabilities of query terms to get probability of query
- Discuss in pairs/groups:
  - What are the limitations of this model?
  - How can the model be improved?

Stochastic Language Models

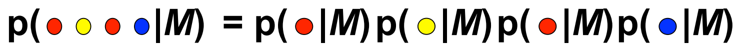
- A statistical model taking order into account





Model Approximations

- Unigram Language Models
  - What we had in first example

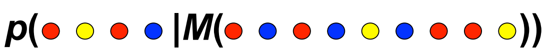


- Bigram (generally, n-gram) Language Models

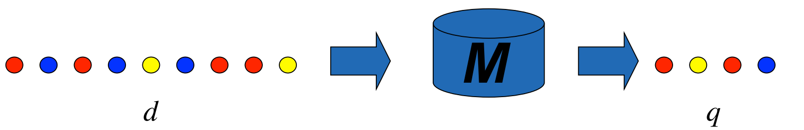
- Other Language Models
  - Grammar-based models (PCFGs), etc.
    - Generalizations of bigrams
    - Probably not the first thing to try in IR

Fundamental problem of LMs

- Usually we don't know the model *M*
  - But have a sample of text (document *d*) representative of that model



- Estimate language model *M* from sample *d*
- Then compute the observation probability of *q*



LMs vs. BIMs for IR

- LMs computationally tractable, intuitively appealing
  - Conceptually simple and explanatory
  - Formal mathematical model
  - Natural use of collection statistics
- LMs provide effective retrieval if
  - Language models are accurate representations of the data
  - Users have some sense of term distribution

LMs vs. BIMs for IR

- LM has no explicit “Relevance”
  - Relevance feedback is difficult to integrate, as are user preferences, and other general issues of relevance
  - Current extensions focus on putting relevance back into the model
- LM assumes that documents and expressions of information user need are of the same type
  - Unrealistic
  - Very simple models of language
  - Current extensions add more model layers

Next

- Computer hall session (April **8**, 15.00-19.00)
  - Orange (Osquars Backe 2, level 4)
  - Examination of computer assignment 3
  - Book a time, see the homepage news feed
- Project (April 8 - May 16)
  - Project proposals will be published on the homepage
  - We will divide you in groups of 4 students based on grades on Assignment 1-2
- Guest lectures (April 8, 15, 22, 29)
  - External lecturers, including Filip Radlinsky from Microsoft Research Cambridge!
  - I will announce each lecture on the homepage news feed