Information Retrieval

Modeling Information Retrieval 2: Probabilistic Relevance Ranking

Ulf Leser

Content of this Lecture

- IR Models
- Boolean Model
- Vector Space Model
- Relevance Feedback in the VSM
- Probabilistic Model
- Latent Semantic Indexing
- Outlook: Word Semantics and Word Embeddings

A Probabilistic Interpretation of Relevance

- VSM is fairly heuristic – some kind of similarity with some kind of weighting in some vector space
- Probabilistic models build on well-established and mathematically consistent probability theory
  - Derive relevance formulas from a few basic and sound principles
- Probabilistic model
  - Words appearing in docs are seen as independent events
  - A doc (or query) is a conjunction of events
  - Relevancy boils down to conditional probabilities of (sets of) documents given conjunctions of words
  - Results in a probability that a doc d is relevant to query q

Basic Model

- Given a corpus D and a vocabulary K
- Let R be a set of (relevant) docs, d be a doc, k be a term, and n=|d|
- We model terms as events and (sets of) documents as conjunction of events
  - $p(R) = |R| / |D|$
  - $p(k|R) = \{d \mid k \in d \wedge d \in R\} / |R|$
  - $p(d) = p(k_1,k_2,\ldots k_n) = p(k_1)*p(k_2)*\ldots p(k_n)$
    - Assuming statistical independence
  - $p(d|R) = p(k_1,k_2,\ldots k_n|R) = p(k_1|R)*p(k_2|R)*\ldots p(k_n|R)$
  - $p(k|d) = 1$ if $k \in d$ else 0

Example

	Text	verkauf	haus	italien	gart	miet	blüh	woll
1	Wir verkaufen Häuser in Italien	p(v|1)=1	1	1				
2	Häuser mit Gärten zu vermieten		1		1	1		
3	Häuser: In Italien, um Italien, um Italien herum		1	p(i|3)=1				
4	Die italienschen Gärtner sind im Garten			1	1			
5	Um unser italiensches Haus blüht's		1	1			1	
6	Wir verkaufen Blühendes	1					1	

Example

	V	H	I	G	M	B	W
1	1	1	1				
2		1		1	1		
3		1	1				
4			1	1			
5		1	1			1	
6	1					1	

R

N

- p(v|R)=1/2
- p(h|R)=2/2
- p(i|R)=1/2
- p(b|N)=2/4
- p(v|N)=1/4

- p(v,i|R)=1/2 * 1/2 = 1/4
- p(v,i|N)=1/4 * 3/4 = 3/16

Framework

- Process for answering q
  - Given $R \subseteq D$ of only relevant docs, $N \subseteq D$ of only irrelevant docs
  - Compute p(R|d), the probability that document d belongs to R
- But: At the beginning, we don't know what is relevant
  - Computation of something like p(R|d) must be rooted in probabilities of words in sets of documents
  - But initial queries too short for probabilistic reasoning
  - We need relevant docs to learn about "relevance"
  - Idea: Determine iteratively using feedback
    - Automatic, explicit, implicit
    - Recall VSM with relevance feedback

Odds-Score

- We want to compute rel(d,q), the relevance of d for q
- Since words k_i of d appear both in relevant and in irrelevant docs, we look at the ratio p(R|d) / p(N|d)
  - Also called odds-score

$$rel(d,q) = \frac{p(R \mid d)}{p(N \mid d)} = \frac{p(R \mid k_1,...,k_n)}{p(N \mid k_1,...,k_n)}$$

- Assuming statistical independence of words, we get

$$rel(d,q) = \frac{p(R \mid k_1,...,k_n)}{p(N \mid k_1,...,k_n)} = \frac{p(R \mid k_1) * ... * p(R \mid k_n)}{p(N \mid k_1) * ... * p(N \mid k_n)}$$

Using Bayes

- Using Bayes Theorem

$$rel(d,q) = \frac{p(R \mid d)}{p(N \mid d)} = \frac{p(d \mid R) * p(R) * p(d)}{p(d \mid N) * p(N) * p(d)} \sim \frac{p(d \mid R)}{p(d \mid N)}$$

- p(R), p(N): relative frequency of (ir-)relevant docs in D
  - A-Priori probability of a doc to be (ir-)relevant
  - Constant for a given q and thus irrelevant for ranking docs
- p(d|R) is the probability of drawing the combination of words forming d when drawing words at random from R
  - We need the probability of drawing the words in d from R
  - And we need the probability of not drawing the other words from R

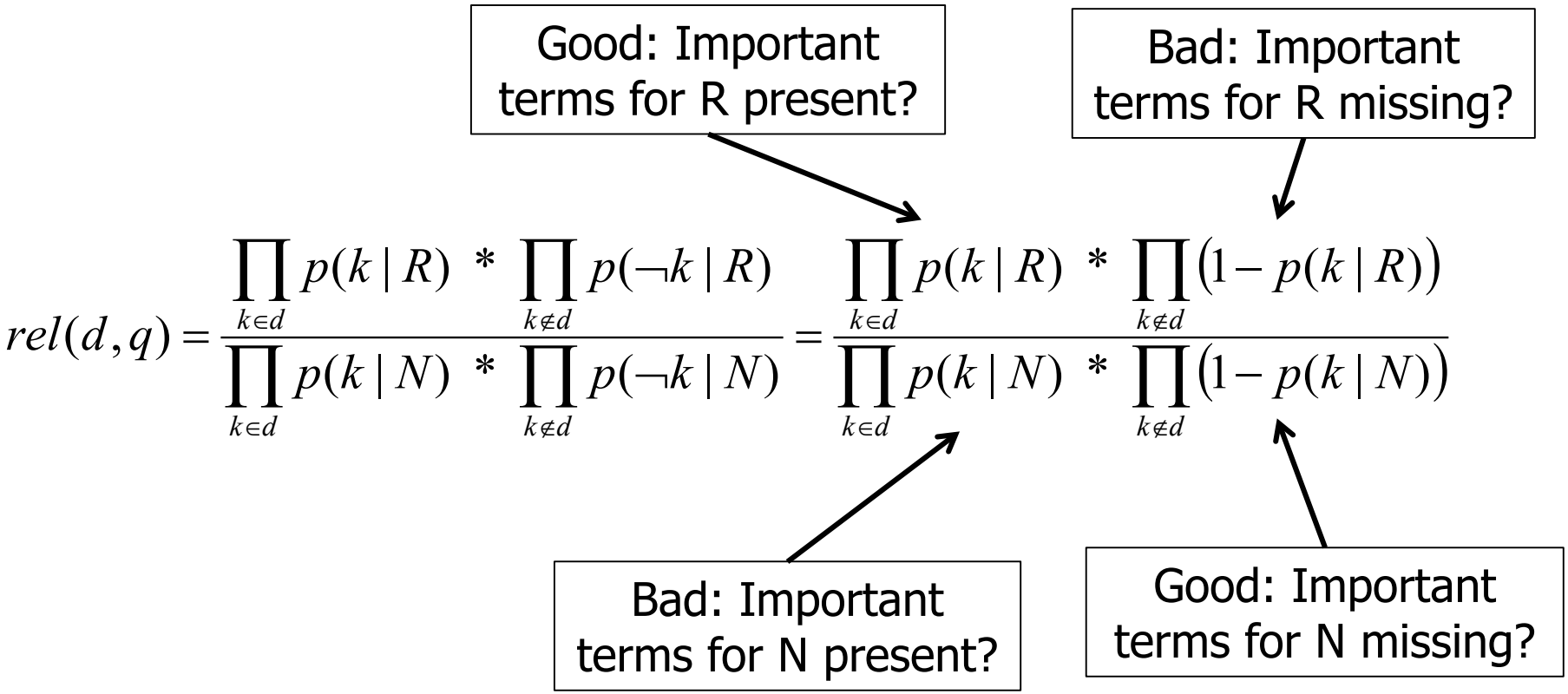
Binary Independence Model

- p(d|R) is the probability of drawing words in d from R and not drawing words not in d from R
- Binary Independence Model

$$rel(d,q) = \frac{p(d \mid R)}{p(d \mid N)} = \frac{\prod_{k \in d} p(k \mid R) \; * \; \prod_{k \notin d} p(\neg k \mid R)}{\prod_{k \in d} p(k \mid N) \; * \; \prod_{k \notin d} p(\neg k \mid N)}$$

  - Having words that are frequent in R raises the relevance of d
  - Not having words that are frequent in R lowers the relevance of d
  - Having words that are frequent in N lowers the relevance of d
  - Not having words that are frequent in N raises the relevance of d

Binary Independence Model



Continuation

- Rephrasing using q

$$rel(d,q) = \frac{\prod_{k \in d \cap q} p(k \mid R)}{\prod_{k \in d \cap q} p(k \mid N)} * \frac{\prod_{k \in d \setminus q} p(k \mid R)}{\prod_{k \in d \setminus q} p(k \mid N)} * \frac{\prod_{k \in q \setminus d} p(\neg k \mid R)}{\prod_{k \in q \setminus d} p(\neg k \mid N)} * \frac{\prod_{k \notin d \cup q} p(\neg k \mid R)}{\prod_{k \notin d \cup q} p(\neg k \mid N)}$$

- Since we are not sure about R and N: Focus on terms in q

$$\ldots \approx \prod_{k \in d \cap q} \frac{p(k \mid R)}{p(k \mid N)} * \prod_{k \in q \setminus d} \frac{p(\neg k \mid R)}{p(\neg k \mid N)} = \prod_{k \in d \cap q} \frac{p(k \mid R)}{p(k \mid N)} * \prod_{k \in q \setminus d} \frac{1 - p(k \mid R)}{1 - p(k \mid N)}$$

Last Step

$$\prod_{k \in d \cap q} \frac{p(k \mid R)}{p(k \mid N)} * \prod_{k \in q \setminus d} \frac{1 - p(k \mid R)}{1 - p(k \mid N)}$$

All matching terms | All non-matching terms

- Some reformulating (duplicating the terms in q)

$$= \prod_{k \in d \cap q} \frac{p(k \mid R) * (1 - p(k \mid N)) * (1 - p(k \mid R))}{p(k \mid N) * (1 - p(k \mid R)) * (1 - p(k \mid N))} * \prod_{k \in q \setminus d} \frac{1 - p(k \mid R)}{1 - p(k \mid N)}$$

$$= \prod_{k \in d \cap q} \frac{p(k \mid R) * (1 - p(k \mid N))}{p(k \mid N) * (1 - p(k \mid R))} * \prod_{k \in q} \frac{1 - p(k \mid R)}{1 - p(k \mid N)}$$

All matching terms | All query terms

Problem

- Last quotient is identical for all d and can be dropped

$$rel(d,q) \approx \prod_{k \in d \cap q} \frac{p(k \mid R) * (1 - p(k \mid N))}{p(k \mid N) * (1 - p(k \mid R))}$$

- But: Computing rel(d,q) requires knowledge of R and N
  - If R and N were known for sure, we could simply use p(k|R) / p(k|N) as relative frequencies of terms in R/N and use these weights for ranking
  - [Also called maximum likelihood estimation]
- In reality, we actually want to find R and N

Back to Reality

- Idea: Approximation using an iterative process
  - Start with "educated guess" for R and set N=D\R
    - E.g. R ~ "all docs containing at least one word from q"
  - Compute relevance of all docs with respect to q
  - Chose relevant docs (by user feedback) or hopefully relevant docs (by selecting the top-r docs)
  - This gives new sets R and N
    - If top-r docs are chosen, we may decide to only change probabilities of terms in R (and disregard the questionable negative information)
  - Compute new conditional probabilities and new ranking
  - Iterate until satisfied
- [Variant of the Expectation Maximization Algorithm (EM)]

Initialization

$$rel(d,q) \approx \prod_{k \in d \cap q} \frac{p(k \mid R) * (1 - p(k \mid N))}{p(k \mid N) * (1 - p(k \mid R))}$$

- Typical simplifying assumptions for the start
  - Terms in non-relevant docs are equally distributed: $p(k|N) \sim df_k/|D|$
  - Terms in relevant doc get equal probability: $p(k|R)=0.5$
  - Much less computation, less weight to unstable first values
- Iterations: Assume we have a new R' and N'. Then

$$p(k|R') = \frac{|\{d|\ k \in d\ and\ d \in R'\}|}{|R'|}$$

$$p(k|N') = \frac{|\{d|\ k \in d\ and\ d \in N'\}|}{|N'|}$$

Example

	Text	verkauf	haus	italien	gart	miet	blüh	woll
1	Wir verkaufen Häuser in Italien	1	1	1				
2	Häuser mit Gärten zu vermieten		1		1	1		
3	Häuser: In Italien, um Italien, um Italien herum		1	1				
4	Die italienschen Gärtner sind im Garten			1	1			
5	Um unser italiensches Haus blüht's		1	1			1	
6	Wir verkaufen Blühendes	1					1	
Q	Wir wollen ein Haus mit Garten in Italien mieten		1	1	1	1		1

Example: Initialization

$$rel(d,q) \approx \prod_{k \in d \cap q} \frac{p(k \mid R) * (1 - p(k \mid N))}{p(k \mid N) * (1 - p(k \mid R))}$$

	V	H	I	G	M	B	W
1	1	1	1				
2		1		1	1		
3		1	1				
4			1	1			
5		1	1			1	
6	1					1	
Q		1	1	1	1		1

- All docs with at least one word from q
  - R={1,2,3,4,5}, N={6}
- Initial estimations
  - p(k|R)=0.5, p(k|N)= df_k/|D| -> p(verkauf|N)=p(blüh|N)=2/6
  - Smoothing: If p(k|X)=0, set p(k|X)=0.01
- Initial ranking
  - rel(1,q)= p(haus|R)*(1-p(haus|N))*p(italien|R)*(1-p(italien|N)) / p(haus|N)*(1-p(haus|R))*p(italien|N)*(1-p(italien|R))
    = .5*(1-4/6)*.5*(1-4/6) / (4/6*(1-0.5)*4/6*(1-0.5))=
    = 0,66
  - rel(2,q)= p(haus|R)*(1-p(haus|N)*p(garten|R)*(1-p(garten|N)*p(mieten|R)*(1-p(mieten|N) / …
  - rel(3,q)= …

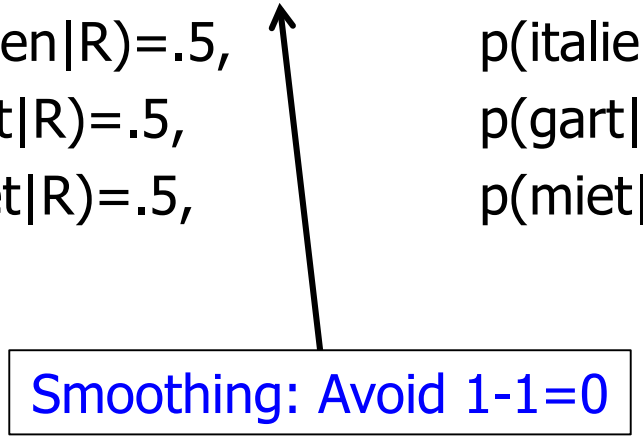
Adjustment

$$P(k \mid R) = \frac{|\{d \mid k \in d, d \in R\}|}{|R|}$$

$$P(k \mid N) = \frac{df_k - |\{d \mid k \in d, d \in R\}|}{|D| - |R|}$$

	V	H	I	G	M	B	W
1	1	1	1				
2		1		1	1		
3		1	1				
4			1	1			
5		1	1			1	
6	1					1	
Q		1	1	1	1		1

- Resulting ranking: <2, {1,3,4,5}, 6>
- Let's use the top-2 docs as new R
  - Second chosen arbitrarily among 1,3,4,5
  - R={1,2}, N={3,4,5,6}
- Adjust scores
  - p(verkauf|R)=.5, p(verkauf|N)=(2-1)/(6-2)=1/4
  - p(haus|R)=1 (~.99), p(haus|N)=(4-2)/(6-2)=2/4
  - p(italien|R)=.5, p(italien|N)=(4-1)/(6-2)=3/4
  - p(gart|R)=.5, p(gart|N)=(2-1)/(6-2)=1/4
  - p(miet|R)=.5, p(miet|N)=(1-1)/(6-2)=0~0.01



Re-Ranking

$$rel(d,q) \approx \prod_{k \in d \cap q} \frac{p(k \mid R) * (1 - p(k \mid N))}{p(k \mid N) * (1 - p(k \mid R))}$$

	V	H	I	G	M	B	W
1	1	1	1				
2		1		1	1		
3		1	1				
4			1	1			
5		1	1			1	
6	1					1	
Q		1	1	1	1		1

- New ranking
  - rel(1,q) = p(haus|R)*(1-p(haus|N))*p(italien|R)*(1-p(italien|N))
    p(haus|N)*(1-p(haus|R))*p(italien|N)*(1-p(italien|R))
    = …
  - rel(2,q) = …
  - …

Pros and Cons

- Advantages
  - Sound (probabilistic) framework
    - Many researchers feel more comfortable – explanations for all steps
  - Results converge to most relevant docs (empirically shown)
    - Under the assumption that relevant docs are similar by sharing term distributions that are different from distributions in irrelevant docs
- Disadvantages
  - Iterative process incurs danger of "drift
  - Assumes statistical independence of terms (as many methods)
  - Difficult to implement efficiently
  - "Has never worked convincingly better in practice" [MS07]

Probabilistic Model versus VSM with Rel. Feedback

- Published 1990 by Salton & Buckley
- Comparison based on various corpora
- Improvement after 1 feedback iteration

eingesetzte Methode		CACM 1033 Dok. 30 Anfr.	CISI 12684 Dok. 84 Anfr.	CRAN 1397 Dok. 225 Anfr.	INSPEC 1460 Dok. 112 Anfr.	MED 3204 Dok. 64 Anfr.	Durchschnitt
initiale Anfrage							
	Precision	0,1459	0,1184	0,1156	0,1368	0,3346	
IDE (dec hi)							
mit allen Termen	Precision	0,2704	0,1742	0,3011	0,2140	0,6305	
	Verbesserung	+86%	+47%	+160%	+56%	+88%	+87%
ausgewählte Terme	Precision	0,2479	0,1924	0,2498	0,1976	0,6218	
	Verbesserung	+70%	+63%	+116%	+44%	+86%	+76%
BIR-Modell							
mit allen Termen	Precision	0,2289	0,1436	0,3108	0,1621	0,5972	
	Verbesserung	+57%	+21%	+169%	+19%	+78%	+69%
ausgewählte Terme	Precision	0,2224	0,1634	0,2120	0,1876	0,5643	
	Verbesserung	+52%	+38%	+83%	+37%	+69%	+56%

- Probabilistic model (BIR) in general worse than VSM+rel feedback (IDE)
  - Probabilistic model does not weight terms in documents
  - Probabilistic model does not allow to weight terms in queries