NLP & IR (WMM9MO75)

Probabilistic IR - 2023

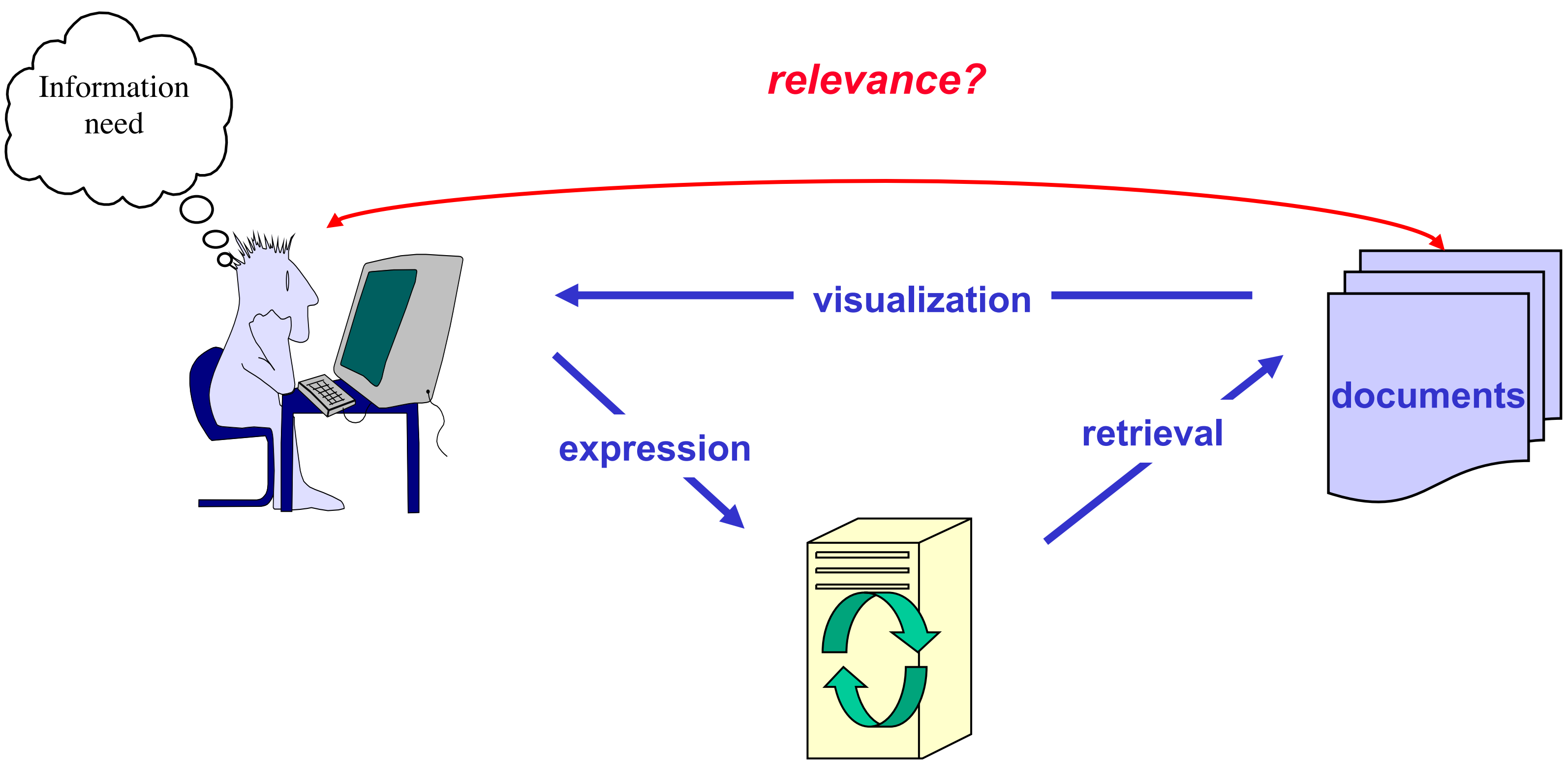
Philippe Mulhem
MRIM-LIG Research group
http://lig-membres.imag.fr/mulhem/

Outline

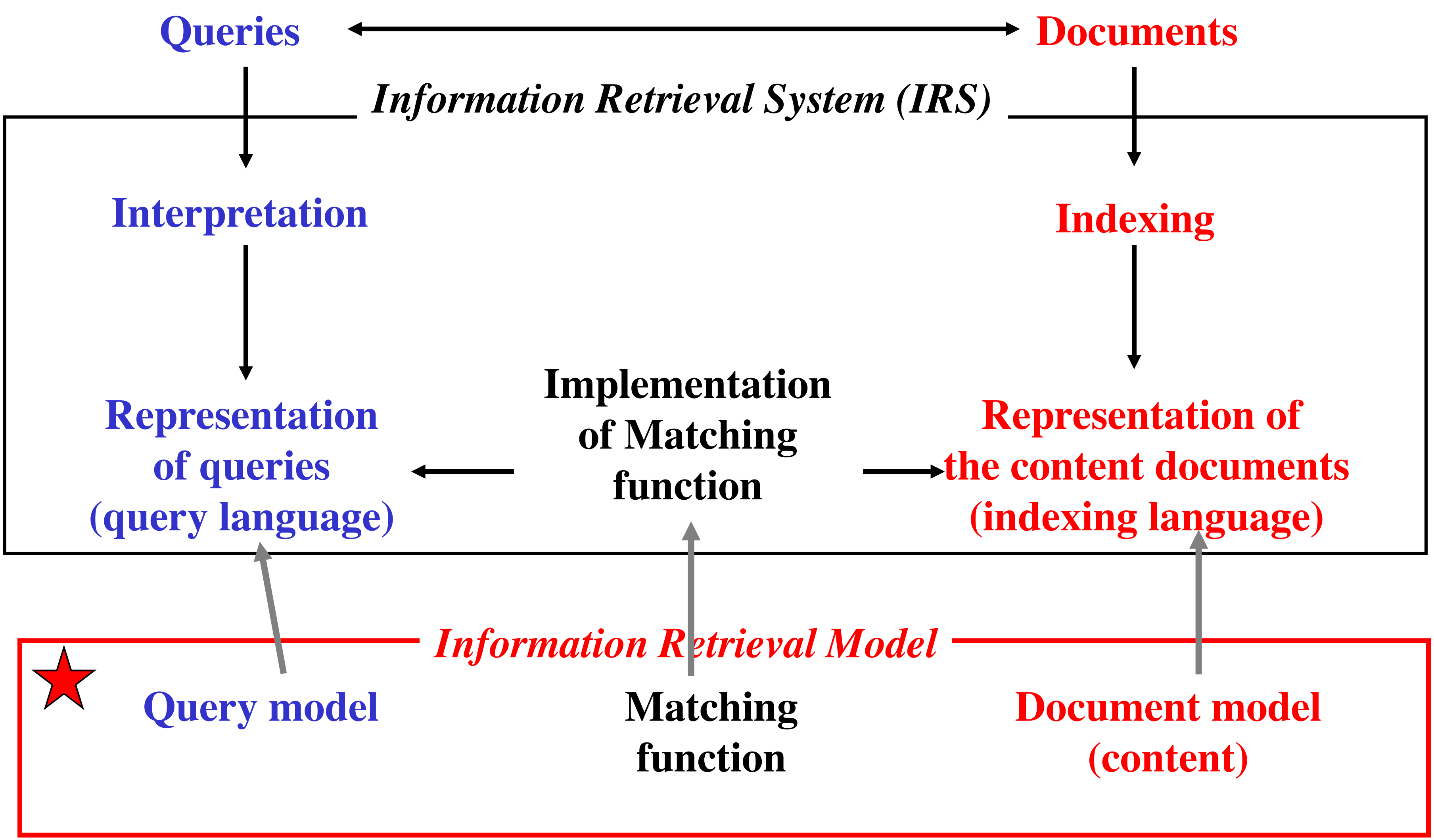
1. Introduction
2. Binary Independent Model
3. Language Models
4. Conclusion

1. Introduction

- Challenge of Information Retrieval:
  - Content base access to documents that satisfy a users information need



1. Introduction



1. Introduction

- Probabilistic IR Models
  - To capture the IR problem in a probabilistic framework
    - First "classical" probabilistic model (Binary Independent Retrieval Model) by Robertson and Spark-Jones in 1976, leading to BM25 [Robertson & Spärk-Jones]
    - Late 80s, Inference Networks [Tutle & Croft]
    - Late 90s, emergence of language models, still hot topic in IR [Croft][Hiemstra][Nie]
  - Question: "what is the probability for a document to be relevant to a query ?"
    - several interpretations explored here

1. Introduction

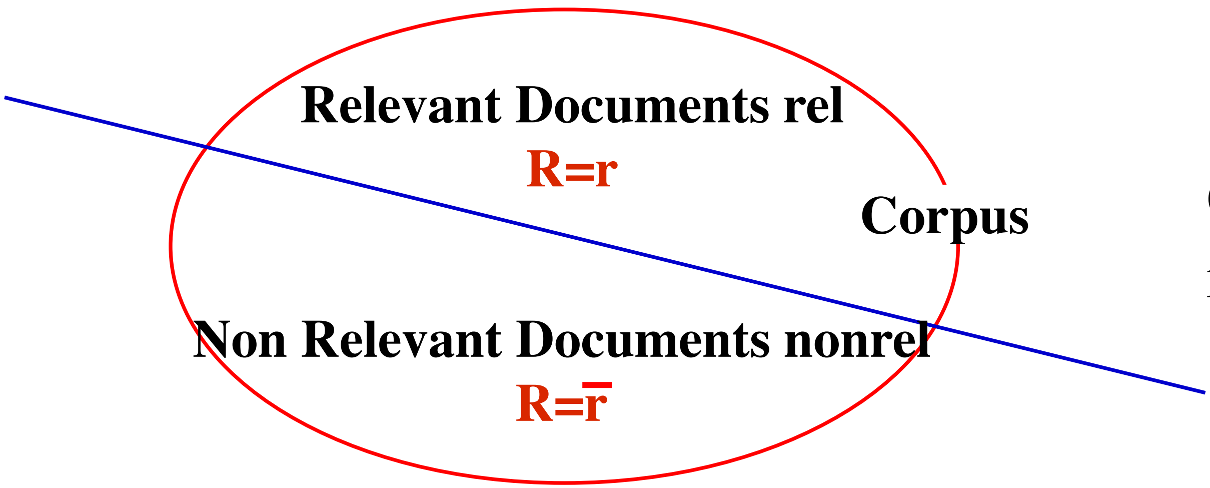
- Probabilistic Models of IR
  - Different approaches of seeing a probabilistic approach for information retrieval
    - Classical approach: probability to have the event *Relevant* knowing one document and one query.
    - Inference Networks approach: probability that the query is true after inference from the content of a document.
    - Language Models approach: probability that a query is generated from a document.

2. Binary Independant Retrieval Model

- [Robertson & Spärk-Jones 1976]
  - Computes the relevance of a document from the relevance known a priori from other documents.
  - Estimated by using the Bayes Theorem and a decision rule
  - Relies on training data

2. BIR

- R: binary random variable
  - R = r : relevant; R = $\bar{r}$: non relevant
  - P(R=r | d, q): probability that R is r for the document d and the query q considered (P(R=r | d, q) is noted *P(r | d, q)*)
    - depends only on document and query



Corpus = rel $\cup$ nonrel
rel $\cap$ nonrel = $\varnothing$

$P(r \mid d, q)$

Probability for the document d belongs to the set of relevant documents (rel) for q

2. BIR

- Matching function :
    - Use of Bayes theorem

Probability to obtain the description d from observed relevance

Relevance probability: the chance of randomly taking one document from the corpus which is relevant for the query q

$$P(r|d,q) = \frac{P(d|r,q).P(r,q)}{P(d,q)}$$

Probability that the document d belongs to the set of relevant documents of the query q.

Probability that the document d is picked for q

2. BIR

Matching function

– Decision rule: document d retrieved if

$$\frac{P(r|d,q)}{P(\bar{r}|d,q)} = \frac{P(d|r,q).P(r,q)}{P(d|\bar{r},q).P(\bar{r},q)} > 1$$

- P(r,q)/P(r,q) constant for a given query (const$\bar{a}$nt): removed for IR
- In IR, it is more convenient to use logs to compute relevance status value *rsv*:

$$rsv(d) =_{rank} \log\left(\frac{P(d|r,q)}{P(d|\bar{r},q)}\right)$$

2. BIR

- Each term t of d is characterized by a a binary variable w^d_t, indicating the presence/absence of the term
  - term weights are binary (d=(11…100…), w^d_t=0 or w^d_t= 1)
  - $P(w^d_t = 1 \mid q, r)$: probability that t occurs in a relevant doc d for q.

    note: $P(w^d_t = 0 \mid q, r) = 1 - P(w_t = 1 \mid q, r)$)
- Hypothesis of conditional independence between terms (Binary Independance) with weight w^d_t for term t in d:

$$P(d \mid r, q) = P(d = (10...110...) \mid r, q) = \prod_{w_t^d = 1} P(w_t^d = 1 \mid r, q) . \prod_{w_t^d = 0} P(w_t^d = 0 \mid r, q)$$

$$P(d \mid \bar{r}, q) = P(d = (10...110...) \mid \bar{r}, q) = \prod_{w_t^d = 1} P(w_t^d = 1 \mid \bar{r}, q) . \prod_{w_t^d = 0} P(w_t^d = 0 \mid \bar{r}, q)$$

2. BIR

– Notations: $p_t = P(w_t = 1 | r, q)$ Prob. of the term t in doc, and doc relevant

$q_t = P(w_t = 1 | \bar{r}, q)$ Prob. of the term t in doc, and doc non-relevant

– Then: $P(w_t = 0 | r, q) = 1 - p_t$ $\quad P(w_t = 0 | \bar{r}, q) = 1 - q_t$

– So

$$rsv(d) =_{rank} \log\left(\frac{P(d|r,q)}{P(d|\bar{r},q)}\right) = \log\left(\frac{\prod_{w_t^d=1} p_t \cdot \prod_{w_t^d=0} 1 - p_t}{\prod_{w_t^d=1} q_t \cdot \prod_{w_t^d=0} 1 - q_t}\right) = \log\left(\prod_{w_t^d=1} \frac{p_t}{q_t} \times \prod_{w_t^d=0} \frac{1 - p_t}{1 - q_t}\right)$$

$$rsv(d|r,q) =_{rank} \log\left(\prod_{w_t^d=1} \frac{p_t}{q_t}\right) + \log\left(\prod_{w_t^d=0} \frac{1 - p_t}{1 - q_t}\right)$$

2. BIR

- We have: $rsv(d|r,q) =_{rank} \log(\prod_{w_t^d=1} \frac{p_t}{q_t}) + \log(\prod_{w_t^d=0} \frac{1-p_t}{1-q_t})$
- Hypothesis: $p_t=q_t$ for all the terms t absent in the query, assuming no impact on the relevance of d for q

$$rsv(d|r,q) =_{rank} \log(\prod_{t \in D \cap Q} \frac{p_t}{q_t}) + \log(\prod_{t \in Q \setminus D} \frac{1-p_t}{1-q_t})$$

2. BIR

– For "inverted files compatibility" (cf. previous lessons):

$$rsv(d|r,q) =_{rank} \log(\prod_{t\in D\cap Q}\frac{p_t}{q_t}) + \log(\prod_{t\in Q\setminus D}\frac{1-p_t}{1-q_t})$$

$$=_{rank} \log(\prod_{t\in D\cap Q}\frac{p_t}{q_t}) \underbrace{- \log(\prod_{t\in D\cap Q}\frac{1-p_t}{1-q_t})} + \log(\prod_{t\in Q\setminus D}\frac{1-p_t}{1-q_t}) \underbrace{+ \log(\prod_{t\in D\cap Q}\frac{1-p_t}{1-q_t})}$$

$$=_{rank} \log(\prod_{t\in D\cap Q}\frac{p_t}{q_t}) + \log(\prod_{t\in D\cap Q}\frac{1-q_t}{1-p_t}) + \log(\prod_{t\in Q\setminus D}\frac{1-p_t}{1-q_t}) + \log(\prod_{t\in D\cap Q}\frac{1-p_t}{1-q_t})$$

$$= \log(\prod_{t\in D\cap Q}\frac{p_t(1-q_t)}{q_t(1-p_t)}) - \log(\prod_{t\in Q}\frac{1-p_t}{1-q_t})$$

constant for a given query Q.

Finally ...

$$rsv(d|r,q) =_{rank} \log(\prod_{t\in D\cap Q}\frac{p_t(1-q_t)}{q_t(1-p_t)})$$

2. BIR

- Or:

$$rsv(d|r,q) =_{rank} \sum_{t \in D \cap Q} \log(\frac{p_t(1-q_t)}{q_t(1-p_t)}) = \sum_{t \in D \cap Q} \log(\frac{p_t}{(1-p_t)} . \frac{(1-q_t)}{q_t}) = \sum_{t \in D \cap Q} \log\left(\frac{\frac{p_t}{1-p_t}}{\frac{q_t}{1-q_t}}\right)$$

- Question: how to estimate p_t and q_t ?

$p_t = P(w_t = 1|r,q)$: Prob. of the term t in doc, and doc relevant

$q_t = P(w_t = 1|\bar{r},q)$: Prob. of t in doc, and doc non-relevant

$P(w_t = 0|r,q) = 1 - p_t$: Prob. of t not in doc, and doc relevant

$P(w_t = 0|\bar{r},q) = 1 - q_t$: Prob. of t not in doc, and doc non-relevant

2. BIR

- Estimation of p_t and q_t on a set of resolved queries

  (queries for which we know the relevant answers in the corpus of N documents)

	Relevant	Non Relevant	Total
term t present	r_t	$n_t - r_t$	n_t
term t absent	$R_t - r_t$	$N - n_t - (R_t - r_t)$	$N – n_t$
Total	R_t	$N - R_t$	N

– With

  - r_t: number of relevant documents for q containing the term t
  - R_t: number of relevant documents for q that contains t
  - N: number of documents in the corpus
  - $n_t - r_t$: number of non relevant documents containing t

2. BIR

- Estimation of p_t and q_t on a set of resolved queries

	Relevant	Non Relevant	Total
term t present	r_t	$n_t - r_t$	n_t
term t absent	$R_t - r_t$	$N - n_t - (R_t - r_t)$	$N - n_t$
Total	R_t	$N - R_t$	N

$$p_t = \frac{r_t}{R_t} \qquad 1 - p_t = \frac{R_t - r_t}{R_t}$$

$$q_t = \frac{n_t - r_t}{N - R_t} \qquad 1 - q_t = \frac{N - R_t - n_t + r_t}{N - R_t}$$

2. BIR

- Global formula

$$rsv(D) =_{rank} \sum_{t \in D \cap Q} \log \left(\frac{\frac{r_t / R_t}{(R_t - r_t)/R_t}}{\frac{(n_t - r_t)/(N - R_t)}{(N - R_t - n_t + r_t)/(N - R_t)}} \right) = \sum_{t \in D \cap Q} \log \left(\frac{\frac{r_t}{R_t - r_t}}{\frac{n_t - r_t}{N - R_t - n_t + r_t}} \right)$$

- Modified to avoid "problems" with 0s:

$$rsv(D) =_{rank} \sum_{t \in D \cap Q} \log \left(\frac{\frac{r_t + 0.5}{R_t - r_t + 0.5}}{\frac{n_t - r_t + 0.5}{N - R_t - n_t + r_t + 0.5}} \right)$$

2. BIR

- Need set of resolved queries to estimate the probabilities
- Problem of initial probabilities
  - For terms not in the resolved queries ?
- Limited to binary events (term present/absent)

=> Basic model binary and independent

2. BIR => BM25

- Best Match [Robertson 1994]: BM25
  – Weighted terms (queries and docs) without resoved queries
  – Lenght of documents

$$rsv_{BM25}(d|r,q) =_{rank} \sum_{t \in d \cap q} \underbrace{\log\left(\frac{N - n_t + 0.5}{n_t + 0.5}\right)}_{\sim \text{idf}} . \underbrace{\frac{(k_1 + 1) w_t^d}{k_1((1-b) + b.\frac{dl}{avdl}) + w_t^d}}_{\sim \text{tf}_d} . \underbrace{\frac{(k_3 + 1).w_t^q}{k_3 + w_t^q}}_{\sim \text{tf}_q}$$

Common values :
k_1 in [1, 2]
b=0.75
k_3 in [0, 1000]

dl: document length
avdl: average document lenght

Pre-Neural State of the art results

4. Language Models of IR

- Probability that a document generates the query
- Consider two dices d1 and d2 so that :
  - for d1 $P(1) = P(3) = P(5) = \frac{1}{3} - \varepsilon \quad P(2) = P(4) = P(6) = \varepsilon$
  - for d2 $P(1) = P(3) = P(5) = \varepsilon \quad P(2) = P(4) = P(6) = \frac{1}{3} - \varepsilon$

- Suppose we observe the sequence Q={1,3,3,2}.
- What dice, d1 or d2, is likely to have generated this sequence ?

4. Language Models of IR

$$P(Q|d1) = (\frac{1}{3} - \varepsilon)^3 . \varepsilon \qquad P(Q|d2) = (\frac{1}{3} - \varepsilon) . \varepsilon^3$$

$$if\ \varepsilon = 0.01$$

$$P(Q|d1) = 3.38E - 4 \qquad P(Q|d2) = 2.99E - 6$$

4. Language Models of IR

- Link with IR
  - the documents are the dices
    - we represent documents as "documents models"
  - the query is the observed sequence

4. Language Models of IR

Inspired from speech understanding theory

- Idea : Use of statistical techniques to estimate both document models and the matching score of document for a query
  - Document model ?
    - A document is a « bag of terms »
    - A language model of a document is a probability function of its terms. The terms being part of the indexing vocabulary.

4. Language Models of IR

– Models

- Probability P of occurrence of a word or a word sequence in one language
  - Consider a sequence *s* composed of words : $m_1, m_2, \ldots, m_l$.
  - The probability *P(s)* may be computed by

$$P(s) = \prod_{i=1}^{l} P(m_i | m_1 \ldots m_{i-1} m_1 \ldots m_{i-1})$$

  - For complexity reasons, consider only the n-1 preceding words of a word (*ngram* model)

$$P(m_i | m_1 \ldots \underbrace{m_{i-1}}) = P(m_i | \underbrace{m_{i-n+1} \ldots m_{i-1}})$$

4. Language Models of IR

– Models

- Unigram $$P(s) = \prod_{i=1}^{l} P(m_i)$$

- Bigram $$P(s) = \prod_{i=1}^{l} P(m_i | m_{i-1}) = \prod_{i=1}^{l} \frac{P(m_{i-1}\, m_i)}{P(m_{i-1})}$$

- Trigram $$P(s) = \prod_{i=1}^{l} P(m_i | m_{i-2}\, m_{i-1}) = \prod_{i=1}^{l} \frac{P(m_{i-2}\, m_{i-1}\, m_i)}{P(m_{i-2}\, m_{i-1})}$$

In IR, most approaches use unigrams

4. Language Models of IR

- Basic idea :

$$P(R=r|d,q)=P(q|\theta_d,R=r) \quad noted \quad P(q|\theta_d)$$

meaning:

what is the probability that a user, who finds the document d relevant, should use the query q (to retrieve d) ?

Question: how to estimate θ_d ?

4. Language Models of IR

- Estimation of θ_d
  - Ex.: Multinomial distribution
    - example : one urn with several marbles of *c* colors, several marbles of each color may appear. A sequence of colors (marble picked and put back) is modelled by a multinomial law of probability:

      ex.: p([c1, c2, c2])=p(c1)*p(c2)*p(c2) with $\sum_c p(c) = 1$
    - For documents [Song and Croft 1999]:
      - the probability that the query terms get selected from the document
      - with the vocabulary V (i.e. the set of all words):
      - each word occurrence is independant

$$P(q|\theta_d) = \frac{|q|!}{\prod_{t\in V}\left(|w_t^q|!\right)} \prod_{t\in V} p(t|\theta_d)^{w_t^q} \propto \prod_{t\in V} p(t|\theta_d)^{w_t^q}$$

  - Note: $\sum_{t\in V} p(t|\theta_d) = 1$

4. Language Models of IR

- How to estimate the parameters of the model?
  - A simple solution: use the Maximum Likelihood Estimate (MLE) to fit the statistical model to the data: We look for the $p(t|\theta_d)$ that maximize the probability to observe the document.

$$P_{ML}(t|\theta_d) = \frac{w_d^t}{\sum_{t \in V} w_d^t} = \frac{w_d^t}{|d|}$$ with w^t_d the count of t in d

Note, we have: $$\sum_{t \in V} P_{ML}(t|\theta_d) = \frac{\sum_{t \in V} w_d^t}{|d|} = \frac{|d|}{|d|} = 1$$

4. Language Models of IR

- Is it done, so? Not really... consider
  - a vocabulary V={"day", "night", "sky"}
  - a document d so that θ_d={p_{ML}(day| θ_d)=0.67, p_{ML}(night| θ_d)=0.33, p_{ML}(sky| θ_d)=0}
  - a query q="day sky"
  - then: $p(q|\theta_d) \propto p_{ML}(day|\theta_d)^1 * p_{ML}(sky|\theta_d)^1$

$$= 0.67 * 0$$

$$= 0 \quad \ldots!$$

But d matches partially the query !!!

→ not good for IR !

4. Language Models of IR

- Problem:
  - we use only the document source to model the probability distribution,
  - the document is not large enough to estimate accurately the probabilities

➔ p_{ML} alone is not sufficient for the language model of documents.

- Solution: to integrate data from a larger set
  - What do we have ? => The collection of documents

4. Language Models of IR

- *Probability smoothing*
  - we *smooth* p_{ML} by a probability coming from the corpus
  - the probability coming from the corpus is defined as

$$P(t|C) = \frac{\sum_{d \in C} w_d^t}{\sum_{d \in C} \sum_{t \in V} w_d^t} = \frac{\sum_{d \in C} w_d^t}{\sum_{d \in C} |d|}$$

- Several smoothings exist, corresponding to several ways to manage the integration between the data from the documents and the corpus

4. Language Models of IR

- Jelinek-Mercer smoothing
  - fixed coefficient interpolation

$$P_\lambda(t|\hat{\theta}_d) = (1-\lambda).P_{ML}(t|\theta_d) + \lambda.P(t|C)$$

  - one λ in [0, 1] for all the documents
  - when λ=0, $P_\lambda = P_{ML}$ (useless for IR, see before)
  - when λ=1, $P_\lambda = \lambda.P(t|C)$: all document models are the same as the collection model. (useless)
  - Optimization of λ on one test collection (λ≈0.15)
  - simple to compute, good results

4. Language Models of IR

- Implementation formula for one query q:

$$\log(P_\lambda(q|\hat{\theta}_d)) \propto \sum_{t \in q \cap d} \frac{w_q^t}{|q|} . \log(\frac{(1-\lambda)}{\lambda} . \frac{w_d^t}{|d|} . \frac{\sum_{d \epsilon C} w_d^t}{\sum_{d \epsilon C} |d|} + 1)$$

compatible with inverted files

4. Language Models of IR

- Dirichlet smoothing
  - interpolation dependant of each document, with one parameter μ (supposingly better as it takes into account a specificity of each document)
  - considers that the corpus adds pseudo occurrences of terms (non integer), the same pseudo-occurrences for one term for all documents:

$$P_{\mu}(t|\hat{\theta}_d) = \frac{w_d^t + \mu P(t|C)}{|d| + \mu}$$

4. Language Models of IR

- Dirichlet smoothing follows multinomial distribution

$$\sum_{t \in V} P_\mu(t|\hat{\theta}_d) = \frac{1}{\sum_{t \in V} w_d^t + \mu} . \sum_{t \in V} (w_d^t + \mu P(t|C))$$

$$= \frac{1}{\sum_{t \in V} w_d^t + \mu} . (\sum_{t \in V} w_d^t + \mu \sum_{t \in V} P(t|C))$$

$$= \frac{1}{\sum_{t \in V} w_d^t + \mu} . (\sum_{t \in V} w_d^t + \mu) = 1$$

4. Language Models of IR

- Dirichlet smoothing
  - link with Jelinek-Mercer smoothing

$$P_\mu(t|\hat{\theta}_d) = \frac{w_d^t + \mu P(t|C)}{|d| + \mu} = \frac{|d|}{|d| + \mu} . \frac{w_d^t}{|d|} + \frac{\mu}{|d| + \mu} P(t|C)$$

$$= \frac{|d|}{|d| + \mu} . P_{ML}(t|\theta_d) + \underbrace{\frac{\mu}{|d| + \mu}}_{\approx\lambda} P(t|C)$$

  - long documents have less smoothing (because more data)
  - Dirichlet smoothing: very good results (values around 1500 or greater)

4. Language Models of IR

- Smoothing is linked to inverse document frequency (IDF) [Lafferty & Zhai 2001]
  - consider that a general smoothing is of the form

$$P_\mu(t|\hat{\theta}_d) = \begin{cases} p_s(t|\theta_d) & \text{if t in document d} \\ \alpha_d \text{p(t|C)} & \text{otherwise} \end{cases}$$

method	$p_s(t\|\theta_d)$	α_d	Parameter
Jelinek-Mercer	$(1-\lambda).P_{ML}(t\|\theta_d)+\lambda.P(t\|C)$	λ	λ
Dirichlet	$\frac{w_d^t + \mu P(t\|C)}{\sum_{t \in V} w_d^t + \mu}$	$\frac{\mu}{\sum_{t \in V} w_d^t + \mu}$	μ

4. Language Models of IR

- Smoothing linked to IDF

$$\log P(q|\hat{\theta}_d) =_{rank} \sum_{t\in q} \log P(t|\hat{\theta}_d)$$

$$=_{rank} \sum_{t\in q\cap d} \log p_s(t|\hat{\theta}_d) + \sum_{t\in q\setminus d} \log \alpha_d\, p(t|C)$$

trick → $$- \sum_{t\in q\cap d} \log \alpha_d\, p(t|C) + \sum_{t\in q\cap d} \log \alpha_d\, p(t|C)$$

$$=_{rank} \sum_{t\in q\cap d} \log \frac{p_s(t|\hat{\theta}_d)}{\alpha_d\, p(t|C)} + \sum_{t\in q} \log \alpha_d\, p(t|C)$$

$$=_{rank} \sum_{t\in q\cap d} \log\left(p_s(t|\hat{\theta}_d) * \frac{1}{\alpha_d\, p(t|C)}\right) + |q|.\log \alpha_d + \sum_{t\in q} \log p(t|C)$$

"similar" to TF.IDF

Constant per doc.

Constant per query

4. Language Models of IR

- Generalization of the original matching function, negative Kullback-Leibler divergence:

$$-KL(\theta_q \| \hat{\theta}_d) = -\sum_{t \in V} P(t|\theta_q) \log \frac{P(t|\theta_q)}{P(t|\hat{\theta}_d)}$$

- KL divergence compares two probabilities distributions
  - how to code one distribution with another one

4. Language Models of IR

- KL divergence on multinomial distributions of query and document and MLE similar to original matching:

$$-KL(\theta_q \| \hat{\theta}_d) = -\sum_{t \in V} P(t|\theta_q) \log \frac{P(t|\theta_q)}{P(t|\hat{\theta}_d)}$$

$$= -\sum_{t \in V} \frac{w_t^q}{|q|} \log P(t|\theta_q) + \sum_{t \in V} \frac{w_t^q}{|q|} \log P(t|\hat{\theta}_d)$$

$$=_{rank} \sum_{t \in V} w_t^q \log P(t|\hat{\theta}_d)$$

$$=_{rank} \log \prod_{t \in V} P(t|\hat{\theta}_d)^{w_t^q}$$

$$=_{rank} P(q|\hat{\theta}_d)$$

4. Language Models of IR

- The KL divergence considers by definition comparison of distributions, closer to the usual meaning of matching in IR.

- KL is implemented as Language Model matching in Terrier and Lemur.

5. Conclusion

- Language models are pre-neural state of the art IR
  - Multinomial
  - Dirichlet smoothing
  - Strong fundamentals, links to heuristics in IR (TF, IDF)
- Many extentions
  - cluster-based smoothing
  - other probability models (Poisson)
  - other smoothings
- LM state of the art word-based, competing with BM25

Bibliography

- C. Zhai, Statistical Language Models for Information Retrieval, Morgan&Claypool, 2009
- Zhai&Lafferty, A Study of Smoothing Methods for Language Models Appplied to Ad Hoc Information Retrieval, ACM SIGIR 2001, pp334-342
- F. Song and W.B. Croft. A general language model for information retrieval. In Proceedings of Eighth International Conference on Information and Knowledge Management (CIKM'99), 1999.
- H. Turtle , W. B. Croft, Inference networks for document retrieval, Proceedings of the 13th annual international ACM SIGIR, p.1-24, September 05-07, 1990.
- J. Ponte and W.B. Croft, A Language Modeling Approach to Information Retrieval, ACM SIGIR 1998.

Bibliography

- S. Robertson and K. Spark Jones (1976), Relevance weighting of search terms. Journal of the American Society for Information Science. n°27. pp. 129-146.
- S. Robertson et al., Okapi at TREC-3, TREC-3 conference, 1994.
- A. Singhal, Modern Information Retrieval: A Brief Overview, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 2001.
- M. Boughanem, W. Kraaj an J.Y. Nie, Modèles de langue pour la recherche d'information, *in* les systèmes de recherche d'information : modèles conceptuels, Hermes 2004.