UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Ranking Models

Structural Models

Rodrygo L. T. Santos
rodrygo@dcc.ufmg.br

The ranking problem

$$f(q,d)$$

Simplifying assumptions

Query terms occur independently of one another
- [information retrieval] = [information + retrieval]

All query term occurrences worth the same
- information (title) = information (anchor text) = …
- retrieval (title) = retrieval (anchor text) = …

Structural models

Exploiting query structure
- Term dependence models

Exploiting document structure
- Field-based models

Exploiting query structure

Term independence widely assumed
- Make modeling simpler
- Make estimation tractable

Key limitation
- Assumption does not hold in practice
- $P(\text{retrieval} \mid \text{information}) \neq P(\text{retrieval})$

Term dependence models

Full dependence model
- $P(t_1 \dots t_k) = P(t_1)P(t_2|t_1) \dots P(t_k|t_1 \dots t_{k-1})$
- Infeasible in practice (expensive and unreliable)

Tunable dependence (e.g., bigram model)
- $P(t_1 \dots t_k) = P(t_1)P(t_2|t_1) \dots P(t_k|t_{k-1})$
- Considers only dependences wrt previous terms

Term dependence models

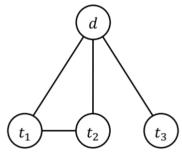
Different dependence types
- [hubble telescope achievements]

Short-range dependences
- hubble-telescope, telescope-achievements

Long-range dependences
- hubble-achievements, achievements-hubble

Markov Random Field (MRF) model

Undirected graphical model representing the joint probability $P(q, d)$ over $q = t_1 \ldots t_k$ and d



Markov Random Field (MRF) model

Undirected graphical model representing the joint probability $P(q, d)$ over $q = t_1 \ldots t_k$ and d
- $P(q, d) = \frac{1}{Z_\Lambda} \prod_{c \in C(G)} \psi(c; \Lambda)$

where
- $c \in C(G)$ is a clique on G
- $\psi(c; \Lambda)$ is a potential function over c

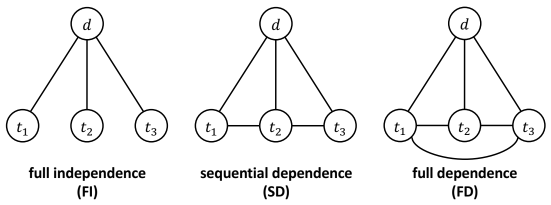
Dependence types

Full independence (FI)
- All terms are independent

Sequential dependence (SD)
- Terms dependent on neighbor terms

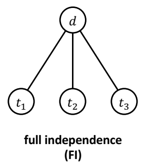
Full dependence (FD)
- Terms dependent on all other terms

Dependence types



Potential functions

Linear models defined over cliques
- $\psi(c; \Lambda) \propto \lambda_x f_x(c_x)$

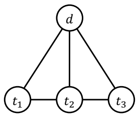


$c_1 \in \{\{t_1, d\}, \{t_2, d\}, \{t_3, d\}\}$

Potential functions

Linear models defined over cliques

◦ $\psi(c;\Lambda) \propto \lambda_x f_x(c_x)$



sequential dependence (SD)

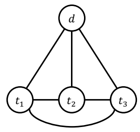
$c_1 \in \{\{t_1, d\}, \{t_2, d\}, \{t_3, d\}\}$

$c_2 \in \{\{t_1, t_2, d\}, \{t_2, t_3, d\}\}$

Potential functions

Linear models defined over cliques

◦ $\psi(c;\Lambda) \propto \lambda_x f_x(c_x)$



full dependence (FD)

$c_1 \in \{\{t_1, d\}, \{t_2, d\}, \{t_3, d\}\}$

$c_2 \in \{\{t_1, t_2, d\}, \{t_2, t_3, d\}, \{t_1, t_2, t_3, d\}\}$

$c_3 \in \{\{t_1, t_3, d\}\}$

Potential functions

Linear models defined over cliques

◦ $\psi(c;\Lambda) \propto \lambda_x f_x(c_x)$

Different measures of compatibility

◦ $f_1(c_1) \equiv P(t_i|d)$

◦ $f_2(c_2) \equiv P(\langle t_i, t_{i+1}\rangle_w|d)$

◦ $f_3(c_3) \equiv P(\langle t_i, t_{k\neq i}\rangle_w|d)$

Markov Random Field (MRF) model

$$\begin{aligned} f(q,d) &= P(d|q) \\ &= P(q,d)/P(q) \\ &\propto P(q,d) \\ &= \tfrac{1}{Z_\Lambda} \textstyle\prod_{c\in C(G)} \psi(c;\Lambda) \\ &\propto \textstyle\sum_{c\in C(G)} \log \lambda_x f_x(c_x) \end{aligned}$$

Markov Random Field (MRF) model

$$\begin{aligned} f(q,d) = \lambda_1 &\sum_{t_i} \log P(t_i|d) \\ + \lambda_2 &\sum_{t_i} \sum_{t_{k=i+1}} \log P(\langle t_i, t_k\rangle_w|d) \\ + \lambda_3 &\sum_{t_i} \sum_{t_{k\neq i}} \log P(\langle t_i, t_k\rangle_w|d) \end{aligned}$$

Proximity matches

How to estimate $P(\langle t_i, t_k\rangle_w|d)$?

◦ Must compute proximity matches

Counting pairs of words within a window of size w

◦ Efficiently computed with positional indexes

Inverted index: positions

and	1,15				
aquarium	3,5				
are	3,3	4,14			
around	1,9				
as	2,21				
both	1,13				
bright	3,11				
coloration	3,12	4,5			
derives	4,7				
due	3,7				
environments	1,8				
fish	1,2	1,4	2,7	2,18	2,23
			3,2	3,6	4,3
			4,13		

marine	2,22	
often	2,2	3,10
only	2,10	
pigmented	4,16	
popular	3,4	
refer	2,9	
referred	2,19	
requiring	2,12	
salt	1,16	4,11
saltwater	2,16	
species	1,18	
term	2,5	
the	1,10	2,4
their	3,9	

Proximity matches

How to estimate $P(\langle t_i, t_k \rangle_w | d)$?

- Must compute proximity matches

Counting pairs of words within a window of size w

- Efficiently computed with positional indexes

How to smooth with collection statistics?

- Infeasible to compute for all pairs – assumed constant

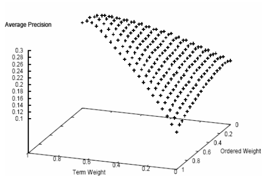
Metric-based parameter tuning

Directly maximize MAP

- Feasible because of the small number of parameters

  $\lambda_1 + \lambda_2 + \lambda_3 = 1$

- Simple hill climbing



Exploiting document structure

Documents often have structure

- HTML tags (e.g., h1, h2, p, a)

Not all parts are equally important

- Document title, URL, metadata, body sections

Can record the scope of word occurrences

- Enable scoped filtering and field-based ranking

Filtering with fields

Attributes in structured domains

[black michael kors dress]

↳ [black:color michael kors:brand dress:category]

Semantic annotations in open domains

[microsoft ceo]

↳ [microsoft:company-3467 ceo:occupation-7234]

Ranking with fields

Straightforward idea

- Apply your favorite ranking function (e.g., BM25, LM) to each document field separately
- Combine field-level scores into a document-level score using a weighted linear combination

$$f(q, d) = \sum_{t \in q} \sum_{a \in d} w_a \, g(\mathrm{tf}_{t,a})$$

What's wrong with per-field scores?

Ranking with fields

Scores across fields are incompatible
- Summing saturated (non-linear) tf over fields may inflate the overall document score
- Sparse background statistics for some fields (e.g., title) may lead to a poor estimation of idf
- Document length semantics varies across fields (e.g., a long anchor text should not be penalized)

Ranking with fields

Rather than combining per-field scores

$$f(q,d) = \sum_{t \in q} \sum_{a \in d} w_a \, g(\mathrm{tf}_{t,a})$$

Combine per-field frequencies

$$f(q,d) = \sum_{t \in q} g\left(\sum_{a \in d} w_a \, \mathrm{tf}_{t,a}\right)$$

Cross-field statistics

Calculate weighted variants of statistics
- $\widetilde{\mathrm{tf}}_{t,d} = \sum_{a \in d} w_a \, \mathrm{tf}_{t,a}$
- $\widetilde{\mathrm{dl}}_d = \sum_{a \in d} w_a \, \mathrm{dl}_a$

Example: BM25F

BM25 [Robertson and Walker, 1994]
- $f(q,d) = \sum_{t \in q} \mathrm{tf}_{t,q} \frac{(k_1+1)\mathrm{tf}_{t,d}}{\mathrm{tf}_{t,d}+k_1\left((1-b)+b\frac{\mathrm{dl}_d}{\mathrm{avdl}}\right)} \log \frac{n+1}{n_t}$

Example: BM25F

BM25F [Robertson and Zaragoza, 2004]
- $f(q,d) = \sum_{t \in q} \mathrm{tf}_{t,q} \frac{(k_1+1)\widetilde{\mathrm{tf}}_{t,d}}{\widetilde{\mathrm{tf}}_{t,d}+k_1\left((1-b)+b\frac{\widetilde{\mathrm{dl}}_d}{\mathrm{avdl}}\right)} \log \frac{n+1}{n_t}$

Empirically, field-specific document length normalization (i.e., b_a) has shown benefits

Summary

Structural properties of queries and documents can help improve ranking effectiveness

Term dependence models
- Both short- and long-range dependences

Field-based models
- Field-adjusted term-document statistics

References

Search Engines: Information Retrieval in Practice, Ch. 11
Croft et al., 2009

Dependence language model for information retrieval
Gao et al., SIGIR 2004

A Markov random field model for term dependencies
Metzler and Croft, SIGIR 2005

References

Modeling higher-order term dependencies in information retrieval using query hypergraphs
Bendersky and Croft, SIGIR 2012

A comparison of retrieval models using term dependencies
Huston and Croft, CIKM 2014

References

Introduction to Information Retrieval, Ch. 6
Manning et al., 2008

Simple BM25 extension to multiple weighted fields
Robertson et al., CIKM 2004

Parameterized fielded term dependence models for ad-hoc entity retrieval from knowledge graph
Nikolaev et al., SIGIR 2016

UFMG UNIVERSIDADE FEDERAL DE MINAS GERAIS

Coming next…

Quality Models

Rodrygo L. T. Santos
rodrygo@dcc.ufmg.br