

Part II

Entity Retrieval

Krisztian Balog

University of Stavanger



Minnesota Children's Museum Deloitte.

GUIDANT

NORTHWEST AIRLINES. ECOLAB

Valspar

Donaldson
Filtration Solutions



Carlson Companies

AMS
Solutions for Life



Medtronic
Alleviating Pain • Restoring Health • Extending Life

Thrivent Financial for Lutherans

MARVIN
Windows and Doors

3M Worldwide

POLARIS
The Way Out.

GENERAL MILLS



What is an entity?

- Uniquely identifiable “thing” or “object”
- Properties:
 - ID
 - Name(s)
 - Type(s)
 - Attributes
 - Relationships to other entities

Entity retrieval tasks

- Ad-hoc entity retrieval
- List completion
- Question answering
 - Factual questions
 - List questions
 - Related entity finding
- Type-restricted variations
 - People, blogs, products, movies, etc.

What's so special about it?

- Entities are not always directly represented
 - Recognise and disambiguate entities in text
 - Collect and aggregate information about a given entity from multiple documents and even multiple data collections
- More structure
 - Types (from some taxonomy)
 - Attributes (from some ontology)
 - Relationships to other entities (“typed links”)

In this Part

- Focus on the ad-hoc entity retrieval task
- Mainly probabilistic models
 - Specifically, Language Models

Outline for Part II

- Crash course into probability theory
- Ranking with ready-made entity descriptions
- Ranking without explicit entity representations
- Evaluation initiatives
- Future directions

Ad-hoc entity retrieval

- **Input:** unconstrained natural language query
 - “telegraphic” queries (neither well-formed nor grammatically correct sentences or questions)
- **Output:** ranked list of entities
- **Collection:** unstructured and/or semi-structured documents

Ranking with ready-made entity descriptions

This is not unrealistic...

The image displays a collage of overlapping web browser windows, illustrating a realistic scenario of a user's online activity. The windows include:

- Wikipedia:** The main page with navigation links like 'Main page', 'Contents', and 'Featured content'.
- IMDb:** The Internet Movie Database homepage with a search bar and navigation links for 'Movies', 'TV', 'News', etc.
- LinkedIn:** A user profile page for 'Krisztian Balog' with navigation links for 'Home', 'Profile', 'Contacts', etc.
- Amazon.com:** The product page for the book 'Modern Information Retrieval: The Concepts and Technology behind Search (2nd Edition)' by Ricardo Baeza-Yates and Berthier Ribeiro-Neto. The page shows a price of \$62.49, a 'Buy New' button, and a 'FREE TWO-DAY SHIPPING FOR COLLEGE STUDENTS' banner. The book description at the bottom states it is a rigorous textbook for a first course on information retrieval from a computer science perspective.

The Amazon.com window is the most prominent, showing the book's title, authors, price, and a 'Buy New' button. It also includes a 'FREE TWO-DAY SHIPPING FOR COLLEGE STUDENTS' banner and a 'Book Description' section at the bottom.

Book Description

Publication Date: **February 10, 2011** | ISBN-10: **0321416910** | ISBN-13: **978-0321416919** | Edition: **2**

This is a rigorous and complete textbook for a first course on information retrieval from the computer science perspective. It provides an up-to-date student oriented treatment of information retrieval including extensive coverage of new topics such as web retrieval, web crawling, open source search engines and user interfaces.

From parsing to indexing, clustering to classification, retrieval to ranking, and user feedback to retrieval evaluation, all of the most important concepts are carefully introduced and exemplified. The contents and structure of the book have been carefully designed by the two main authors, with individual contributions coming from leading international authorities in the field, including Yoelle Maarek, Senior Director of Yahoo! Research Israel; Dulce Poncele on IBM Research; and Malcolm Slaney, Yahoo Research USA.

Document-based entity representations

- Each entity is described by a document
- Ranking entities much like ranking documents
 - Unstructured
 - Semi-structured

Standard Language Modeling approach

- Rank documents d according to their likelihood of being relevant given a query q : $P(d|q)$

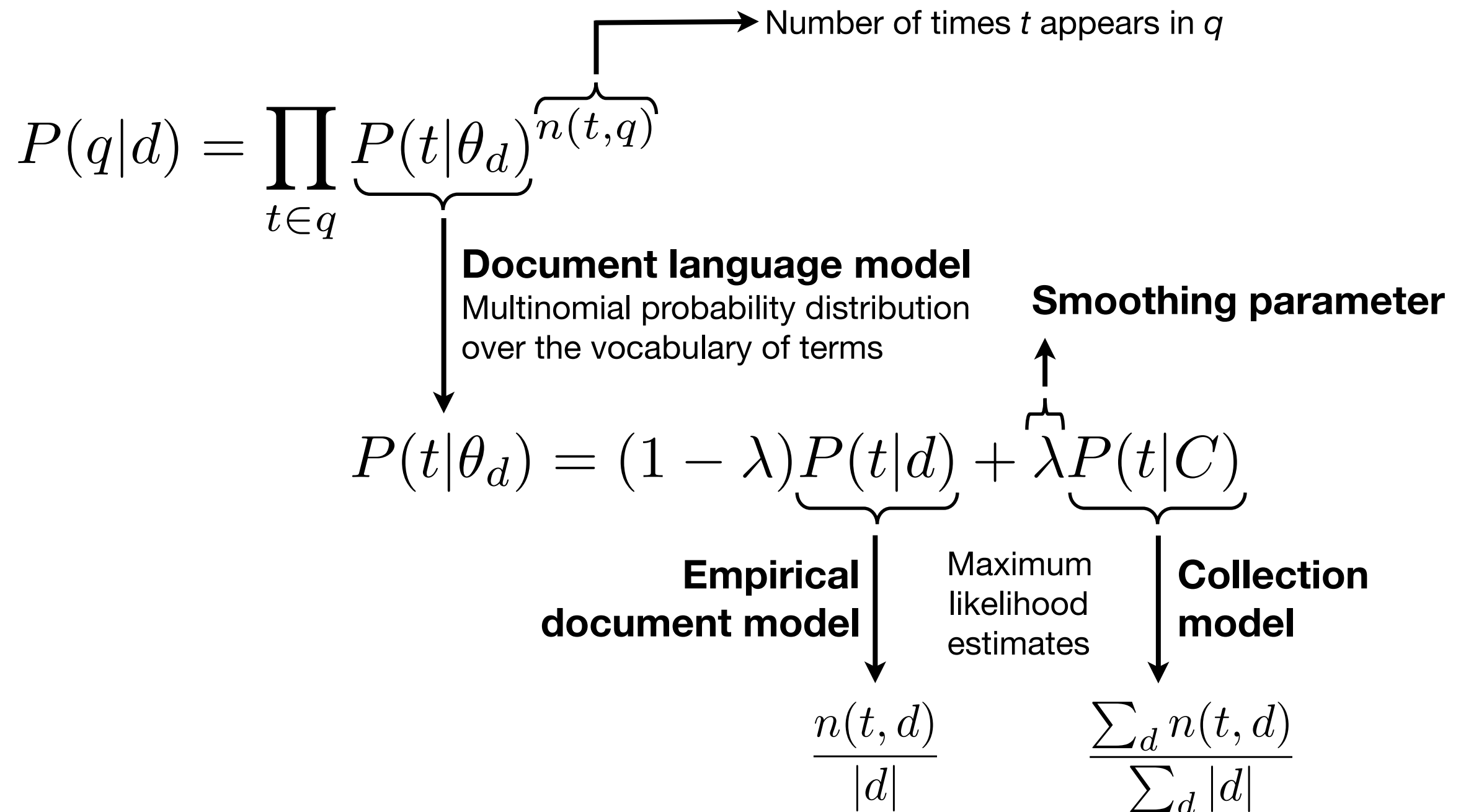
$$P(d|q) = \frac{P(q|d)P(d)}{P(q)} \propto P(q|d)P(d)$$

Query likelihood
Probability that query q
was “produced” by document d

Document prior
Probability of the document
being relevant to *any* query

$$P(q|d) = \prod_{t \in q} P(t|\theta_d)^{n(t,q)}$$

Standard Language Modeling approach (2)



Here, documents=entities, so

$$P(e|q) \propto P(e)P(q|\theta_e) = \underbrace{P(e)}_{\substack{\text{Entity prior} \\ \text{Probability of the entity} \\ \text{being relevant to any query}}} \prod_{t \in q} \underbrace{P(t|\theta_e)}_{\substack{\text{Entity language model} \\ \text{Multinomial probability distribution} \\ \text{over the vocabulary of terms}}}^{n(t,q)}$$

Entity prior
Probability of the entity
being relevant to *any* query

Entity language model
Multinomial probability distribution
over the vocabulary of terms

Semi-structured entity representation

- Entity description documents are rarely unstructured
- Representing entities as
 - Fielded documents -- the IR approach
 - Graphs -- the DB/SW approach



Article Talk

Read

Edit

View history

Search



Audi A4

From Wikipedia, the free encyclopedia

The **Audi A4** is a line of compact executive cars produced since late 1994 by the German car manufacturer Audi, a subsidiary of the Volkswagen Group.

The A4 has been built in four generations and is based on Volkswagen's B platform. The first generation A4 succeeded the Audi 80. The automaker's internal numbering treats the A4 as a continuation of the Audi 80 lineage, with the initial A4 designated as the B5-series, followed by the B6, B7, and the current B8. The B8 A4 is built on the Volkswagen Group MLB platform shared with many other Audi models and potentially one Porsche model within Volkswagen Group.^[2]

Audi A4



Manufacturer Audi

dbpedia:Audi_A4

foaf:name	Audi A4
rdfs:label	Audi A4
rdfs:comment	The Audi A4 is a compact executive car produced since late 1994 by the German car manufacturer Audi, a subsidiary of the Volkswagen Group. The A4 has been built [...]
dbpprop:production	1994 2001 2005 2008
rdf:type	dbpedia-owl:MeanOfTransportation dbpedia-owl:Automobile dbpedia:Audi dbpedia:Compact_executive_car freebase:Audi A4 dbpedia:Audi_A5 dbpedia:Cadillac_BLS
dbpedia-owl:manufacturer	
dbpedia-owl:class	
owl:sameAs	
is dbpedia-owl:predecessor of	
is dbpprop:similar of	

Mixture of Language Models

[Ogilvie & Callan, 2003]

- Build a separate language model for each field
- Take a linear combination of them

$$P(t|\theta_d) = \sum_{j=1}^m \underbrace{\mu_j}_{\text{Field weights}} \underbrace{P(t|\theta_{d_j})}_{\text{Field language model}}$$

Smoothed with a collection model built from all document representations of the same type in the collection

Field weights

$$\sum_{j=1}^m \mu_j = 1$$

Setting field weights

- **Heuristically**
 - Proportional to the length of text content in that field, to the field's individual performance, etc.
- **Empirically (using training queries)**
- **Problems**
 - Number of possible fields is huge
 - It is not possible to optimise their weights directly
- **Entities are sparse w.r.t. different fields**
 - Most entities have only a handful of predicates

Predicate folding

- **Idea:** reduce the number of fields by grouping them together
- Grouping based on
 - Type [\[Pérez-Agüera et al. 2010\]](#)
 - Manually determined importance [\[Blanco et al. 2011\]](#)

Hierarchical Entity Model

[Neumayer et al. 2012]

- Organise fields into a 2-level hierarchy
 - Field types (4) on the top level
 - Individual fields of that type on the bottom level
- Estimate field weights
 - Using training data for field types
 - Using heuristics for bottom-level types

Two-level hierarchy

Name	foaf:name	Audi A4
	rdfs:label	Audi A4
	rdfs:comment	The Audi A4 is a compact executive car produced since late 1994 by the German car manufacturer Audi, a subsidiary of the Volkswagen Group. The A4 has been built [...]
	dbpprop:production	1994 2001 2005 2008
Attributes	rdf:type	dbpedia-owl:MeanOfTransportation dbpedia-owl:Automobile
	dbpedia-owl:manufacturer	dbpedia:Audi
	dbpedia-owl:class	dbpedia:Compact_executive_car
	owl:sameAs	freebase:Audi A4
Out-relations	is dbpedia-owl:predecessor of	dbpedia:Audi_A5
	is dbpprop:similar of	dbpedia:Cadillac_BLS
In-relations		

Formally

$$P(t|\theta_d) = \sum_F \underbrace{P(t|F, d)}_{\text{Term importance}} \underbrace{P(F|d)}_{\text{Field type importance}}$$

Term importance

Field type importance

Taken to be the same for all entities

$$P(F|d) = P(F)$$

$$P(t|F, d) = \sum_{d_f \in F} \underbrace{P(t|d_f, F)}_{\text{Term generation}} \underbrace{P(d_f|F, d)}_{\text{Field generation}}$$

Term generation

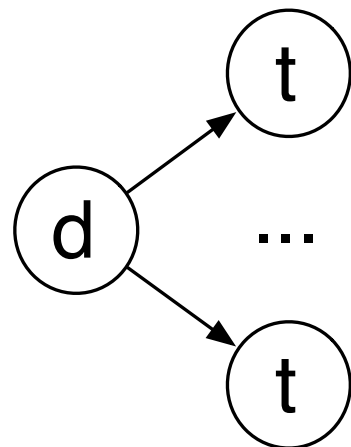
Importance of a term is jointly determined by the field it occurs as well as all fields of that type (smoothed with a coll. level model)

Field generation

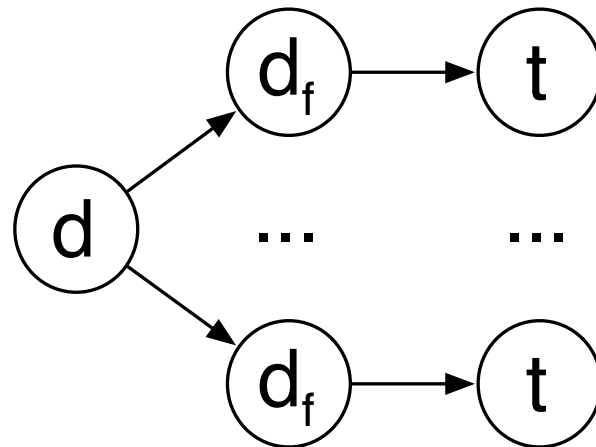
Uniform or estimated heuristically (based on length, popularity, etc)

$$P(t|d_f, F) = (1 - \lambda)P(t|d_f) + \lambda P(t|\theta_{d_F})$$

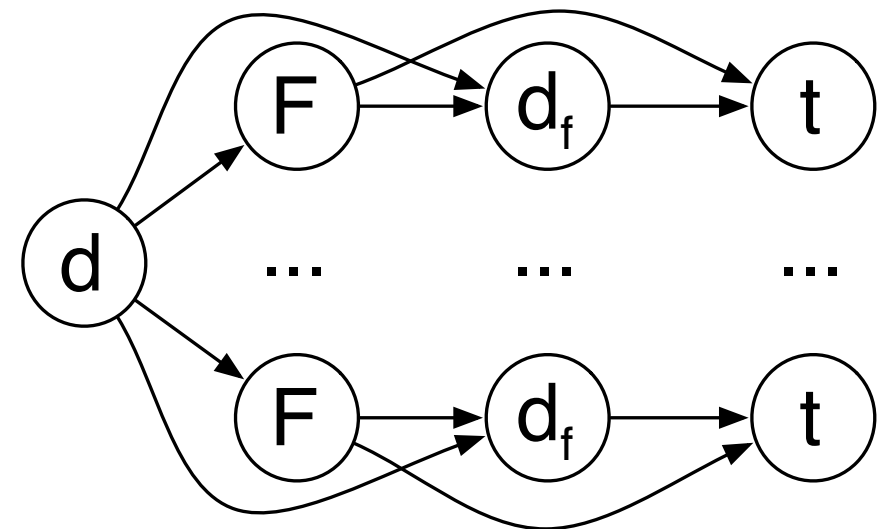
Comparison of models



**Unstructured
document model**



**Fielded
document model**




**Hierarchical
document model**

Probabilistic Retrieval Model for Semistructured data

[Kim et al. 2009]

- Extension to the Mixture of Language Models
- Find which document field each query term may be associated with

$$P(t|\theta_d) = \sum_{j=1}^m \mu_j P(t|\theta_{d_j})$$

**Mapping probability**
Estimated for each query term

$$P(t|\theta_d) = \sum_{j=1}^m \overbrace{P(d_j|t)} P(t|\theta_{d_j})$$

Estimating the mapping probability

$$P(t|C_j) = \frac{\sum_d n(t, d_j)}{\sum_d |d_j|}$$

Term likelihood

Probability of a query term occurring in a given field type

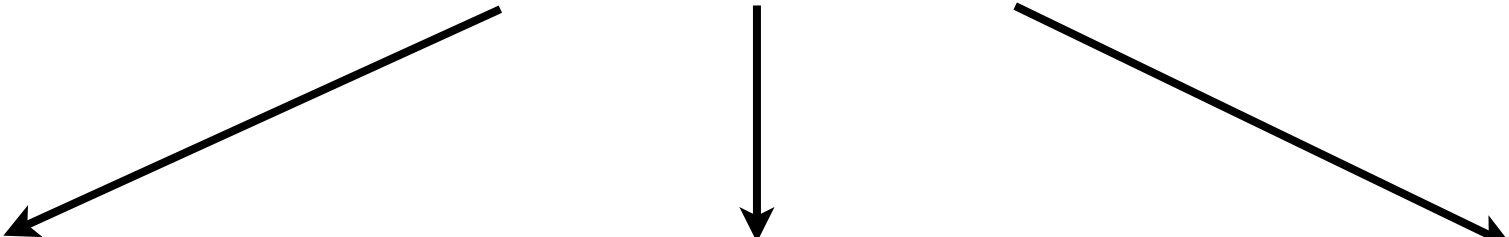
Prior field probability

Probability of mapping the query term to this field before observing collection statistics

$$P(d_j|t) = \frac{\overbrace{P(t|d_j)} \overbrace{P(d_j)}}{\underbrace{P(t)}}$$
$$\sum_{d_k} P(t|d_k) P(d_k)$$

Example

Query: meg ryan war



d_j	$P(t d_j)$
cast	0.407
team	0.382
title	0.187

d_j	$P(t d_j)$
cast	0.601
team	0.381
title	0.017

d_j	$P(t d_j)$
genre	0.927
title	0.070
location	0.002

The usual suspects from document retrieval...

- Priors
 - HITS, PageRank
 - Document link indegree **[Kamps & Koolen 2008]**
- Pseudo relevance feedback
 - Document-centric vs. entity-centric **[Macdonald & Ounis 2007; Serdyukov et al. 2007]**
 - sampling expansion terms from top ranked documents and/or (profiles of) top ranked candidates
 - Field-based **[Kim & Croft 2011]**

J. Kamps and M. Koolen. **The importance of link evidence in Wikipedia.** *ECIR'08*.

C. Macdonald and I. Ounis. **Expertise drift and query expansion in expert search.** *CIKM'07*.

P. Serdyukov, S. Chernov, and W. Nejdl. **Enhancing expert search through query modeling.** *ECIR'07*.

J.Y. Kim and W.B. Croft. **A Field Relevance Model for Structured Document Retrieval.** *ECIR'12*.

So far...


- Ranking (fielded) documents...
- What is special about entities?
 - Type(s)
 - Relationships with other entities


Entity types

`rdf:type`

`dbpedia-owl:MeanOfTransportation`
`dbpedia-owl:Automobile`

Categories: [Audi vehicles](#) | [Compact executive cars](#) | [Euro NCAP large family cars](#) | [Sedans](#) | [Station wagons](#) | [Convertibles](#) | [Vehicles with CVT transmission](#) | [All-wheel-drive vehicles](#) | [Front-wheel-drive vehicles](#) | [Vehicles introduced in 1994](#) | [1990s automobiles](#) | [2000s automobiles](#) | [2010s automobiles](#) | [Hybrid electric cars](#)

 Find... [Browse](#) [Query](#) [Help](#) [Sign In or Sign Up](#) [English](#)



Topic

Audi A4 ^{en}

id: /guid/9202a8c04000641f8000000000305a7c mid: /m/030qmx notable type: /automotive/model notable for: /automotive/model on the web: [Wikipedia](#)

The Audi A4 is a line of compact executive cars produced since late 1994 by the German car manufacturer Audi, a subsidiary of the Volkswagen Group. The A4 has been built in four generations and is based on Volkswagen's B platform. The first generation A4 succeeded the Audi 80. The automaker's internal numbering treats the A4 as a continuation of the Audi 80 lineage, with the initial A4 designated as the B5-series, followed by the B6, B7, and the current B8. The B8 A4 is built on the Volkswagen Group MLB platform shared with many other Audi models and potentially one Porsche model within Volkswagen Group. The Audi A4 automobile layout consists of a longitudinally oriented engine at the front, with transaxle-type transmissions mounted at the rear of the engine. The cars are front-wheel drive, or on some models, "quattro" all-wheel drive. The A4 is available as a saloon/sedan and estate/wagon. The second and third generations of the A4 also had a convertible version, but the B8 version of the convertible became a variant of the Audi A5 instead as Audi got back into the compact executive coupé segment. [Wikipedia](#)

Properties [118n](#) [Keys](#) [Links](#)

Filter options: ☐ Show all domains and properties

[Common](#) /common

[Topic](#) /common/topic

[Also known as](#) /common/topic/alias

[Freebase Commons](#)

Types:

[Common](#)

[Topic](#)

[Automotive](#)

[Automobile Model](#)

Using target types

Assuming they have been identified...

- **Constraining results**
 - Soft/hard filtering
 - Different ways to measure type similarity (between target types and the types associated with the entity)
 - Set-based
 - Content-based
 - Lexical similarity of type labels
- **Query expansion**
 - Adding terms from type names to the query
- **Entity expansion**
 - Categories as a separate metadata field

Modeling terms and categories

[Balog et al. 2011]

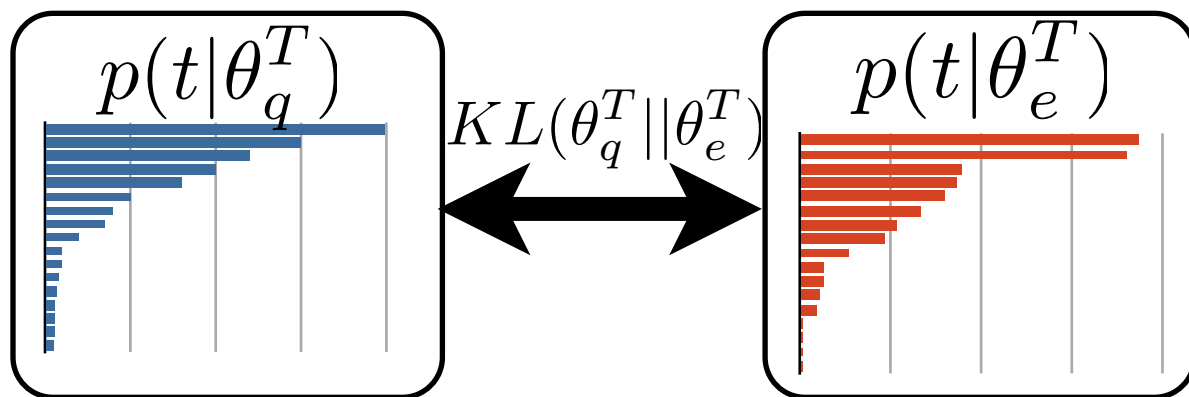
$$P(e|q) \propto P(q|e)P(e)$$

$$P(q|e) = (1 - \lambda)P(\theta_q^T | \theta_e^T) + \lambda P(\theta_q^C | \theta_e^C)$$

Term-based representation

Query model

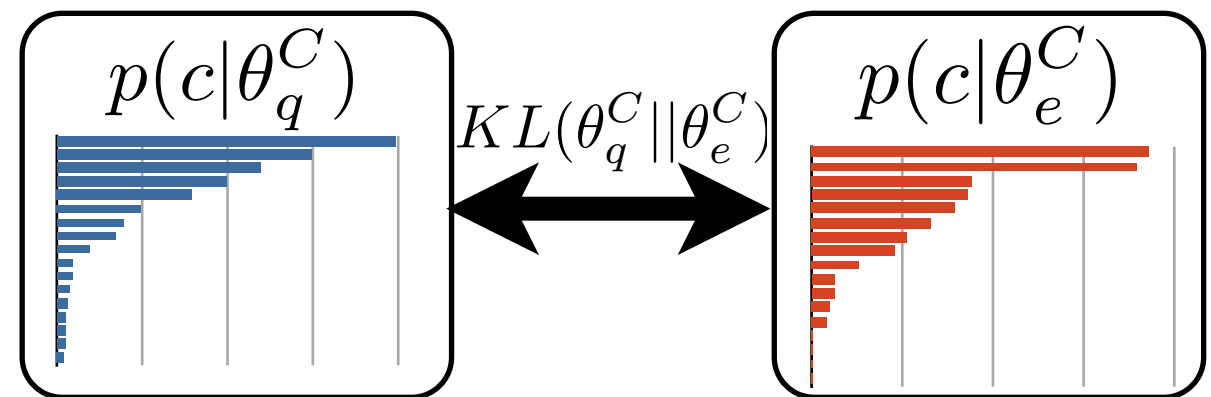
Entity model



Category-based representation

Query model

Entity model



Identifying target types

- Types of top ranked entities [Vallet & Zaragoza 2008]
- Direct term-based vs. indirect entity-based representations [Balog & Neumayer 2012]
- Hierarchical case is difficult...

Expanding target types

- Pseudo relevance feedback
- Based on hierarchical structure
- Using lexical similarity of type labels

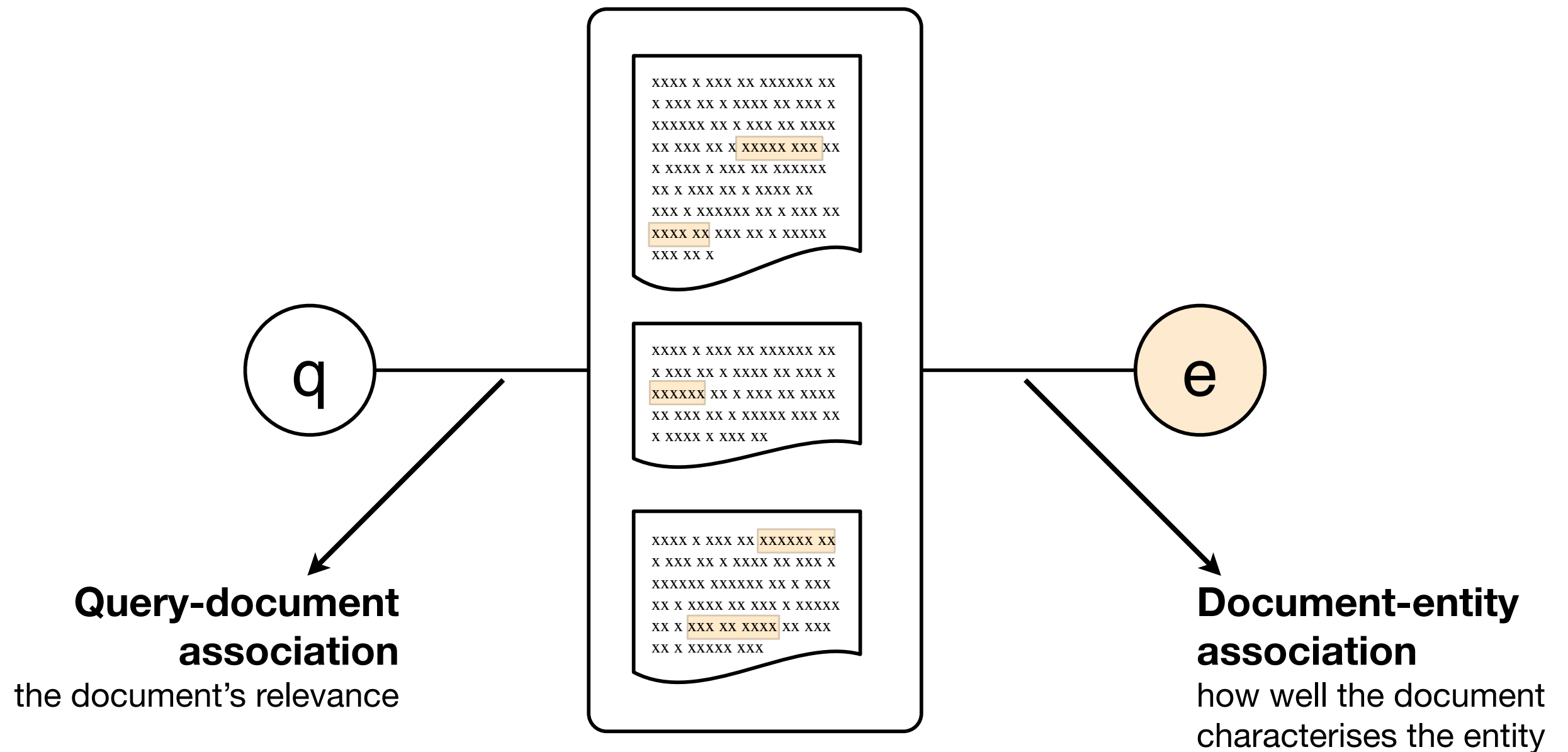
Ranking without explicit entity representations

Scenario

- Entity descriptions are not readily available
- Entity occurrences are annotated

The basic idea

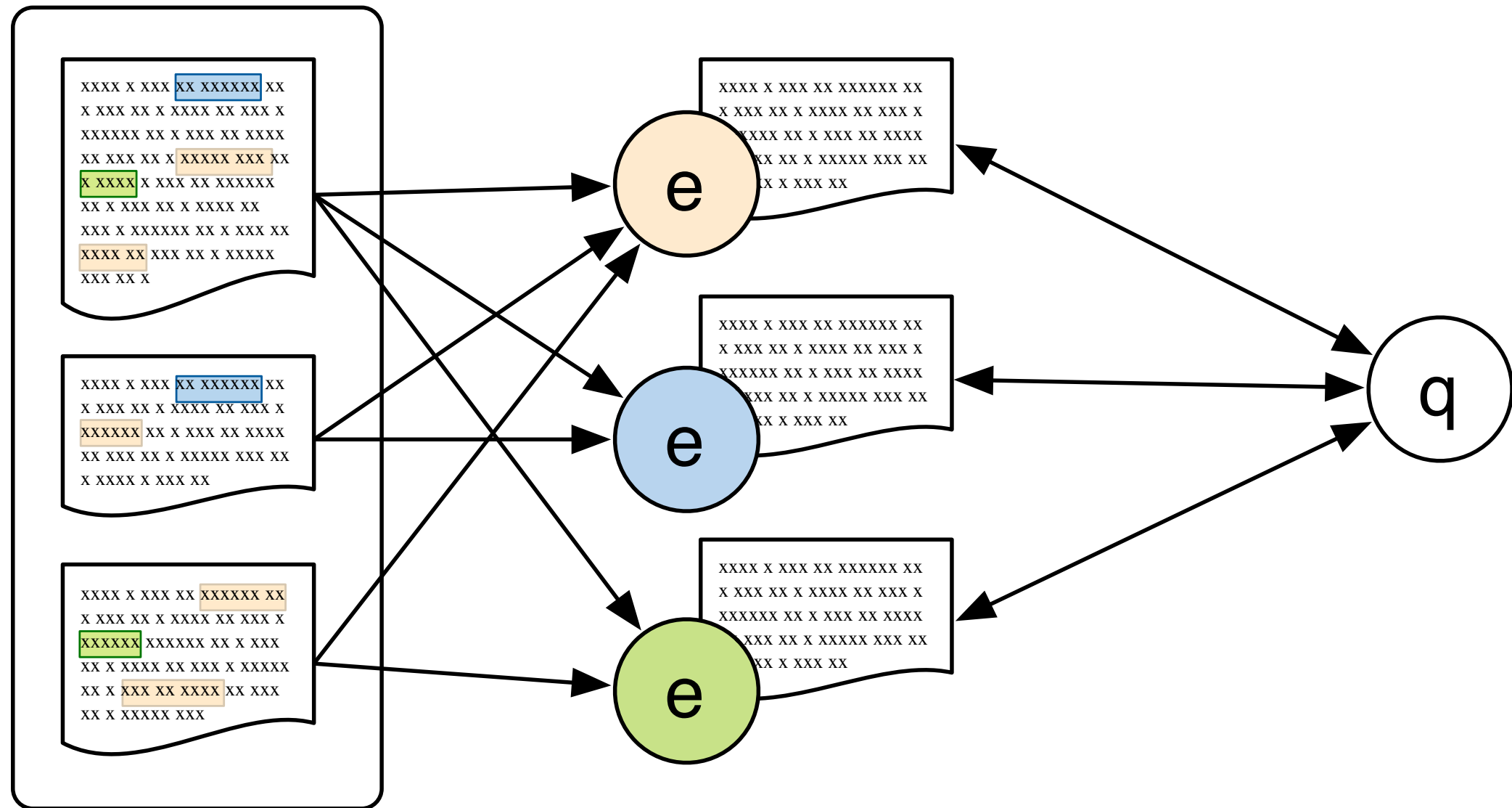
Use documents to get from queries to entities



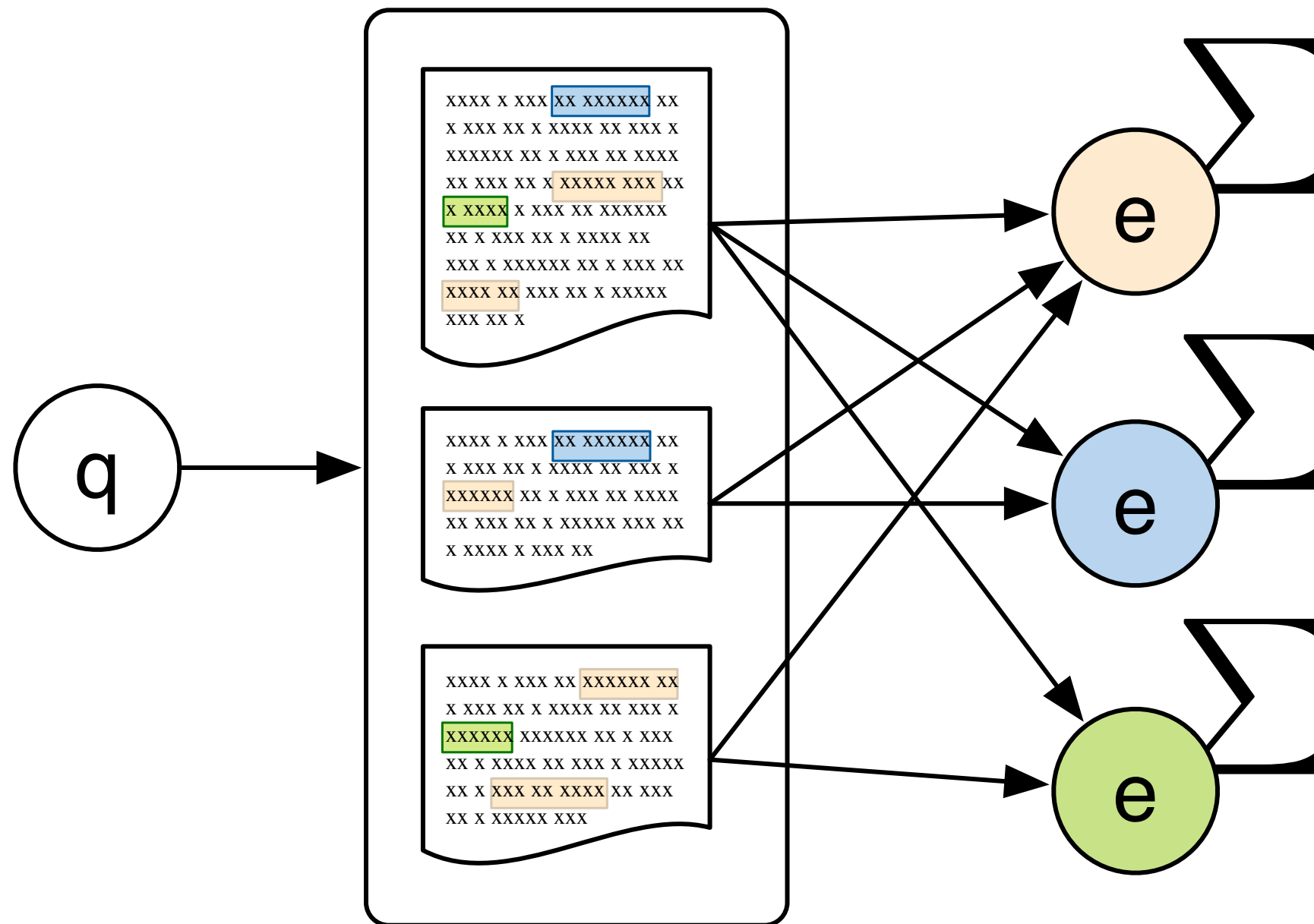
Two principal approaches

- **Profile-based methods**
 - Create a textual profile for entities, then rank them (by adapting document retrieval techniques)
- **Document-based methods**
 - Indirect representation based on mentions identified in documents
 - First ranking documents (or snippets) and then aggregating evidence for associated entities

Profile-based methods



Document-based methods

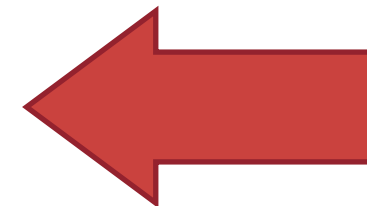


Many possibilities in terms of modeling

- Generative probabilistic models
- Discriminative probabilistic models
- Voting models
- Graph-based models

Generative probabilistic models

- Candidate generation models ($P(e|q)$)
 - Two-stage language model
- Topic generation models ($P(q|e)$)
 - Candidate model, a.k.a. Model 1
 - Document model, a.k.a. Model 2
 - Proximity-based variations
- Both families of models can be derived from the Probability Ranking Principle **[Fang & Zhai 2007]**



Candidate models (“Model 1”)

[Balog et al. 2006]

$$P(q|\theta_e) = \prod_{t \in q} \underbrace{P(t|\theta_e)}_{\text{Smoothing}}$$

Smoothing

With collection-wide background model

$$(1 - \lambda) \underbrace{P(t|e)}_{\text{Term-candidate co-occurrence}} + \lambda P(t)$$

$$\sum_d \underbrace{P(t|d, e)}_{\text{Term-candidate co-occurrence}} \underbrace{P(d|e)}_{\text{Document-entity association}}$$

**Term-candidate
co-occurrence**

In a particular document.
In the simplest case: $P(t|d)$

**Document-entity
association**

Document models (“Model 2”)

[Balog et al. 2006]

$$P(q|e) = \sum_d \underbrace{P(q|d, e)}_{\text{Document relevance}} \underbrace{P(d|e)}_{\text{Document-entity association}}$$

Document relevance

How well document d supports the claim that e is relevant to q

Document-entity association

$$\prod_{t \in q} \underbrace{P(t|d, e)}_{\text{Simplifying assumption}}$$

(t and e are conditionally independent given d)

$$P(t|\theta_d)$$

Document-entity associations

- Boolean (or set-based) approach
- Weighted by the confidence in entity linking
- Consider other entities mentioned in the document

Proximity-based variations

- So far, conditional independence assumption between candidates and terms when computing the probability $P(t|d,e)$
- Relationship between terms and entities that in the same document is ignored
 - Entity is equally strongly associated with everything discussed in that document
- Let's capture the dependence between entities and terms
 - Use their distance in the document

Using proximity kernels

[Petkova & Croft 2007]

$$P(t|d, e) = \underbrace{\frac{1}{Z}}_{\text{Normalising constant}} \sum_{i=1}^N \underbrace{\delta_d(i, t)}_{\text{Indicator function}} \underbrace{k(t, e)}_{\text{Proximity-based kernel}}$$

Normalising constant

Indicator function
1 if the term at position i is t ,
0 otherwise

Proximity-based kernel

- constant function
- triangle kernel
- Gaussian kernel
- step function

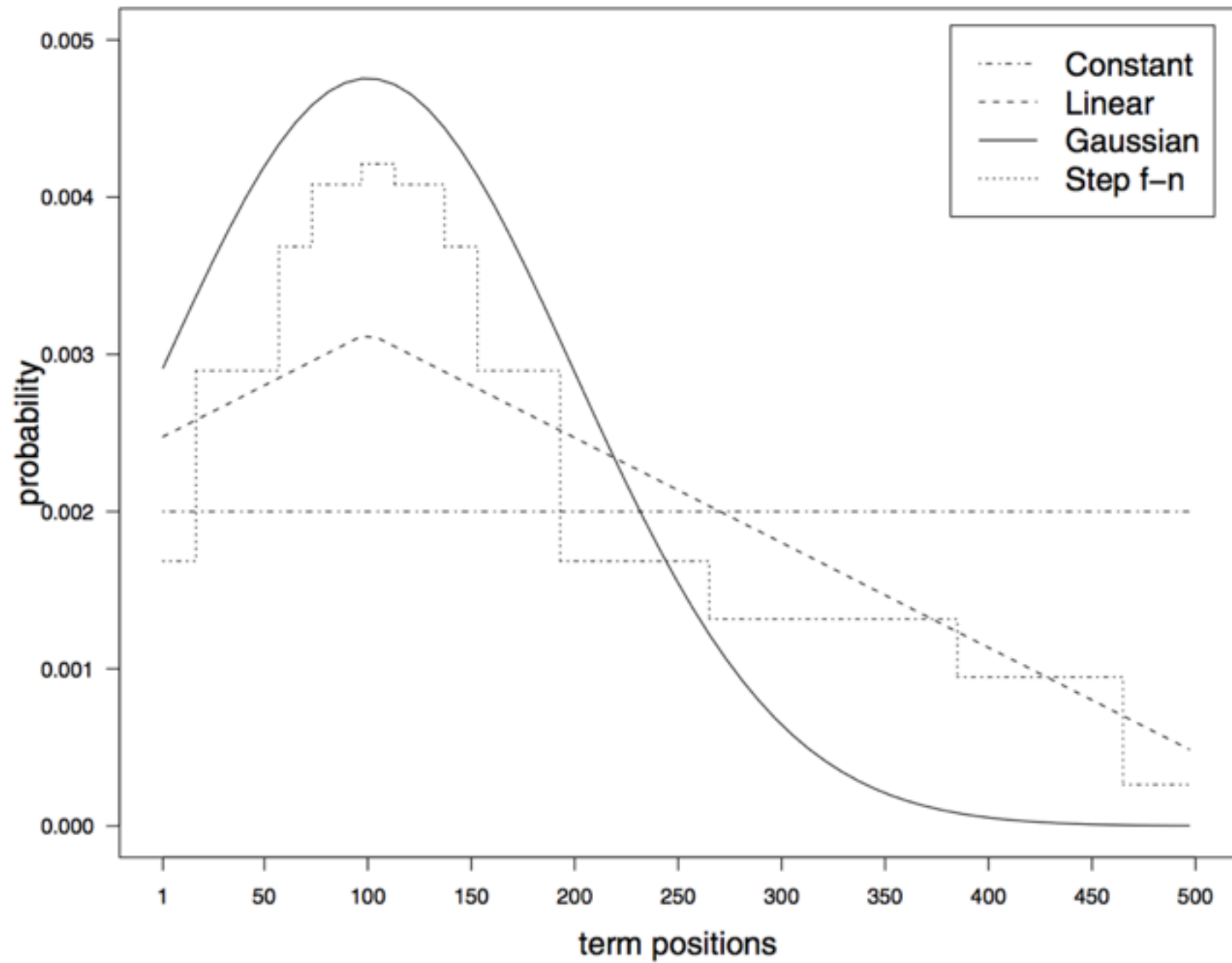


Figure taken from D. Petkova and W.B. Croft. **Proximity-based document representation for named entity retrieval.** *CIKM'07*.

Many possibilities in terms of modeling

- ~~Generative probabilistic models~~
- Discriminative probabilistic models
- Voting models
- Graph-based models

Discriminative models

- Vs. generative models:
 - Fewer assumptions (e.g., term independence)
 - “Let the data speak”
 - Sufficient amounts of training data required
 - Incorporating more document features, multiple signals for document-entity associations
 - Estimating $P(r=1|e,q)$ directly (instead of $P(e,q|r=1)$)
 - Optimisation can get trapped in a local optimum

Arithmetic Mean Discriminative (AMD) model

[Yang et al. 2010]

$$P_{\theta}(r = 1|e, q) = \sum_d \underbrace{P(r_1 = 1|q, d)}_{\text{Query-document relevance}} \underbrace{P(r_2 = 1|e, d)}_{\text{Document-entity relevance}} \underbrace{P(d)}_{\text{Document prior}}$$

logistic function over a linear combination of features \longrightarrow $\underbrace{\sigma}_{\text{standard logistic function}} \left(\sum_{i=1}^{N_f} \underbrace{\alpha_i}_{\text{weight parameters (learned)}} \underbrace{f_i(q, d_t)}_{\text{features}} \right) \sigma \left(\sum_{j=1}^{N_g} \beta_j g_j(e, d_t) \right)$

Learning to rank

- Pointwise
 - AMD, GMD [Yang et al. 2010]
 - Multilayer perceptrons, logistic regression [Sorg & Cimiano 2011]
 - Additive Groves [Moreira et al. 2011]
- Pairwise
 - Ranking SVM [Yang et al. 2009]
 - RankBoost, RankNet [Moreira et al. 2011]
- Listwise
 - AdaRank, Coordinate Ascent [Moreira et al. 2011]

P. Sorg and P. Cimiano. **Finding the right expert: Discriminative models for expert retrieval.** *KDIR'11*.
C. Moreira, P. Calado, and B. Martins. **Learning to rank for expert search in digital libraries of academic publications.** *PAI'11*.
Z. Yang, J. Tang, B. Wang, J. Guo, J. Li, and S. Chen. **Expert2bole: From expert finding to bole search.** *KDD'09*.

Voting models

[Macdonald & Ounis 2006]

- Inspired by techniques from data fusion
 - Combining evidence from different sources
- Documents ranked w.r.t. the query are seen as “votes” for the entity

Voting models

Many different variants, including...

- Votes

- Number of documents mentioning the entity

$$Score(e, q) = |M(e) \cap R(q)|$$

- Reciprocal Rank

- Sum of inverse ranks of documents

$$Score(e, q) = \sum_{\{M(e) \cap R(q)\}} \frac{1}{rank(d, q)}$$

- CombSUM

- Sum of scores of documents

$$Score(e, q) = |\{M(e) \cap R(q)\}| \sum_{\{M(e) \cap R(q)\}} s(d, q)$$

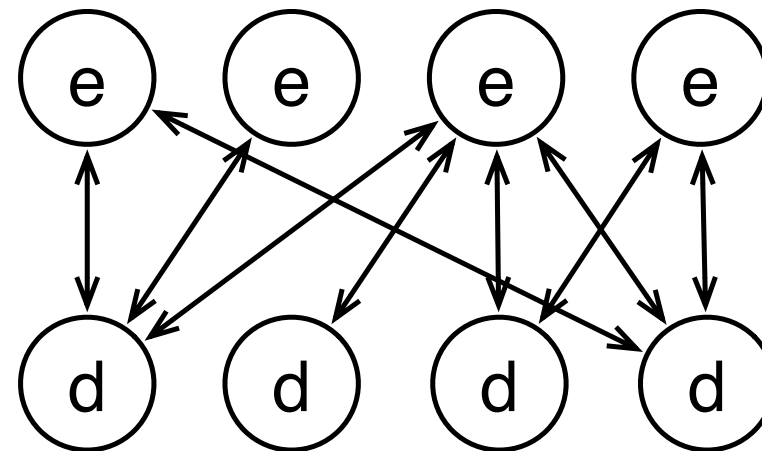
Graph-based models

[Serdyukov et al. 2008]

- One particular way of constructing graphs
 - Vertices are documents and entities
 - Only document-entity edges
- Search can be approached as a random walk on this graph
 - Pick a random document or entity
 - Follow links to entities or other documents
 - Repeat it a number of times

Infinite random walk model

[Serdyukov et al. 2008]

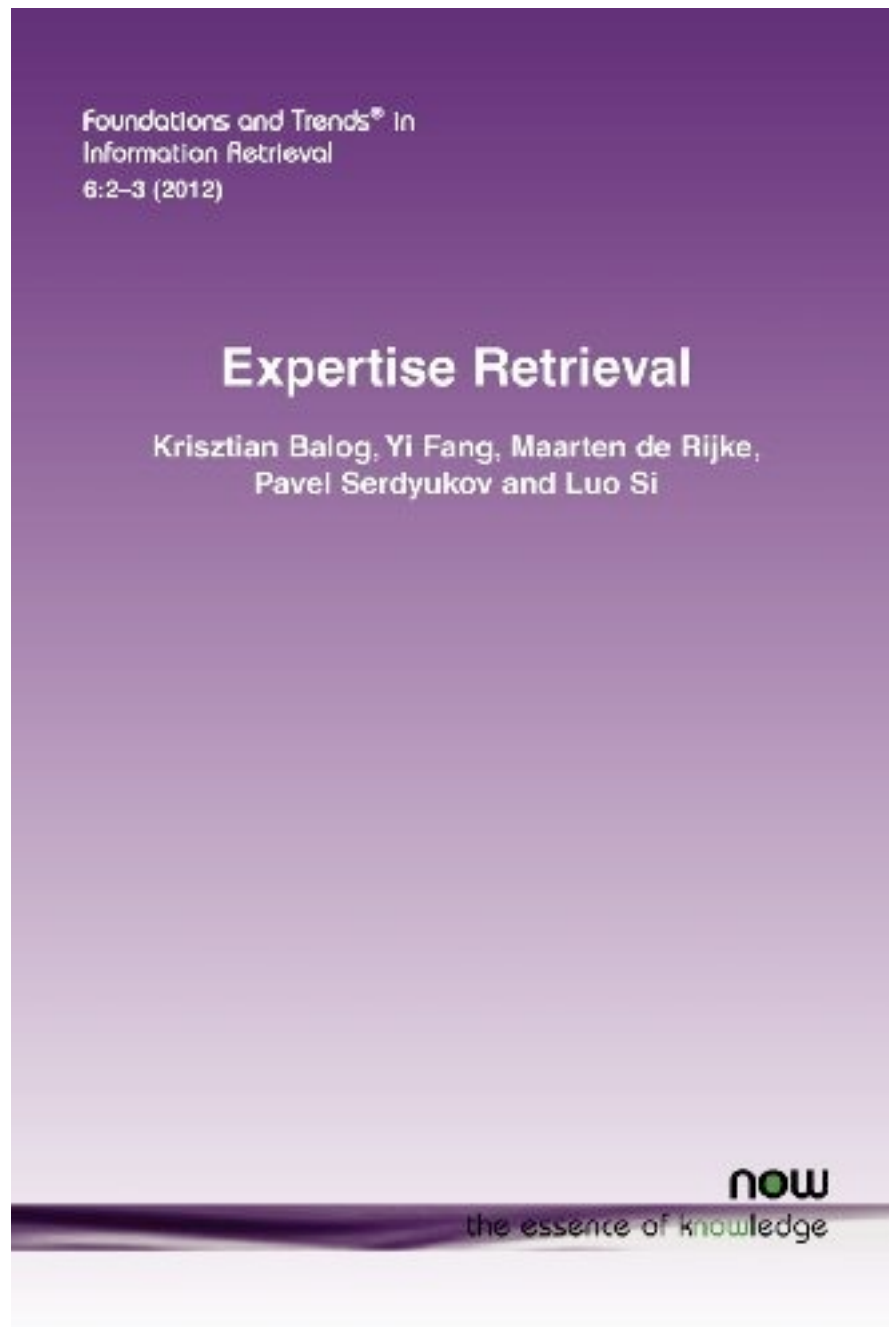


$$P_i(d) = \lambda P_J(d) + (1 - \lambda) \sum_{e \rightarrow d} P(d|e) P_{i-1}(e),$$

$$P_i(e) = \sum_{d \rightarrow e} P(e|d) P_{i-1}(d),$$

$$P_J(d) = P(d|q),$$

Further reading



K. Balog, Y. Fang, M. de Rijke, P. Serdyukov, and L. Si.
Expertise Retrieval. *FnTIR'12*.

Evaluation initiatives

Test collections

Campaign	Task	Collection	Entity repr.	#Topics
TREC Enterprise (2005-08)	Expert finding	Enterprise intranets (W3C, CSIRO)	Indirect	99 (W3C) 127 (CSIRO)
TREC Entity (2009-11)	Rel. entity finding	Web crawl (ClueWeb09)	Indirect	120
	List completion			70
INEX Entity Ranking (2007-09)	Entity search	Wikipedia	Direct	55
	List completion			
SemSearch Chall. (2010-11)	Entity search	Semantic Web crawl (BTC2009)	Direct	142
	List search			50
INEX Linked Data (2012-13)	Ad-hoc search	Wikipedia + RDF (Wikipedia-LOD)	Direct	100 ('12) 144 ('13)

Test collections (2)

- Entity search as Question Answering
 - TREC QA track
 - QALD-2 challenge
 - INEX-LD Jeopardy task

Entity search in DBpedia

[Balog & Neumayer 2013]

- Synthesising queries and relevance assessments from previous eval. campaigns
- From short keyword queries to natural language questions
- 485 queries in total
- Results are mapped to DBpedia

Open challenges

- Combining text and structure
 - Knowledge bases and unstructured Web documents
- Query understanding and modeling
 - See [\[Sawant & Chakrabarti 2013\]](#) at the main conference
- Result presentation
 - How to interact with entities

Resources

- Complete tutorial material
<http://ejmeij.github.io/entity-linking-and-retrieval-tutorial/>
- Referred papers
<http://www.mendeley.com/groups/3339761/entity-linking-and-retrieval-tutorial-at-www-2013-and-sigir-2013/papers/>