

# Part II

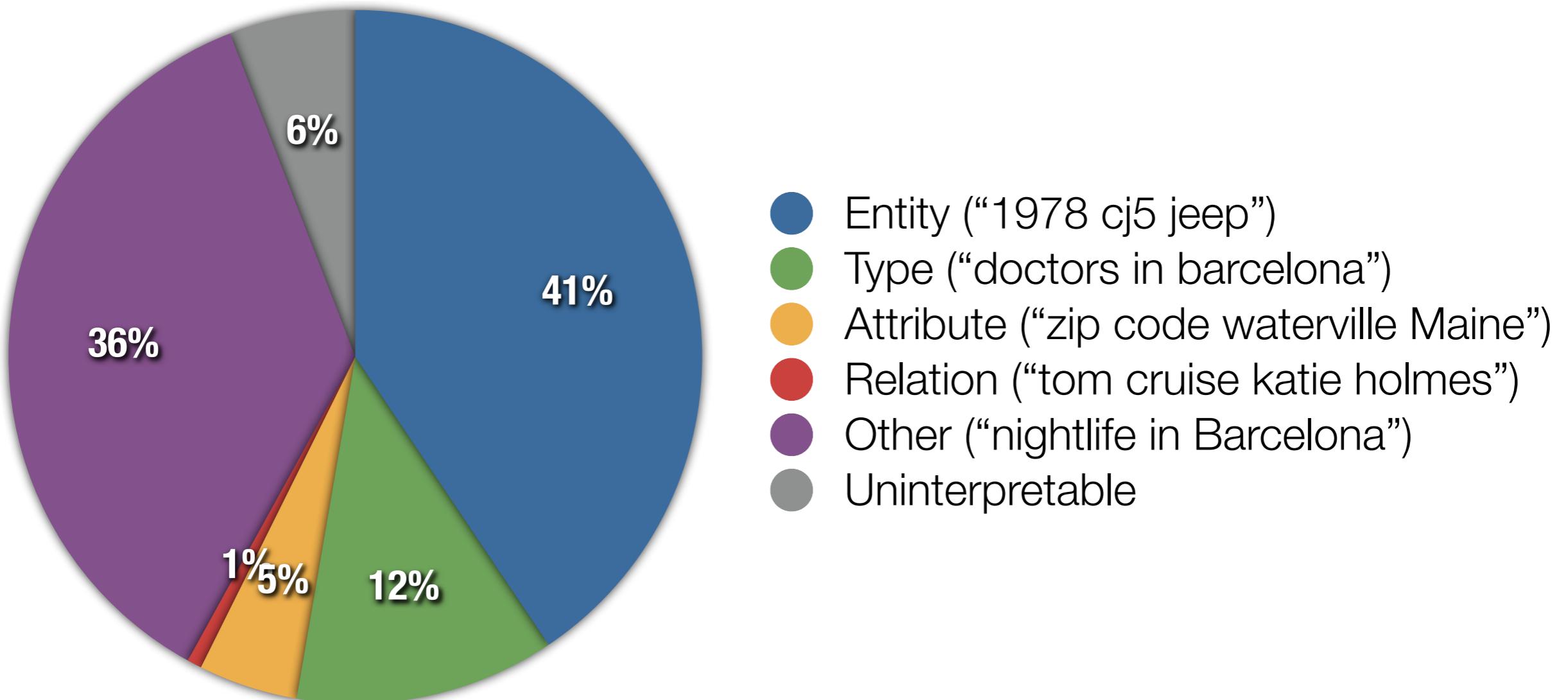
# Entity Retrieval

**Krisztian Balog**  
University of Stavanger

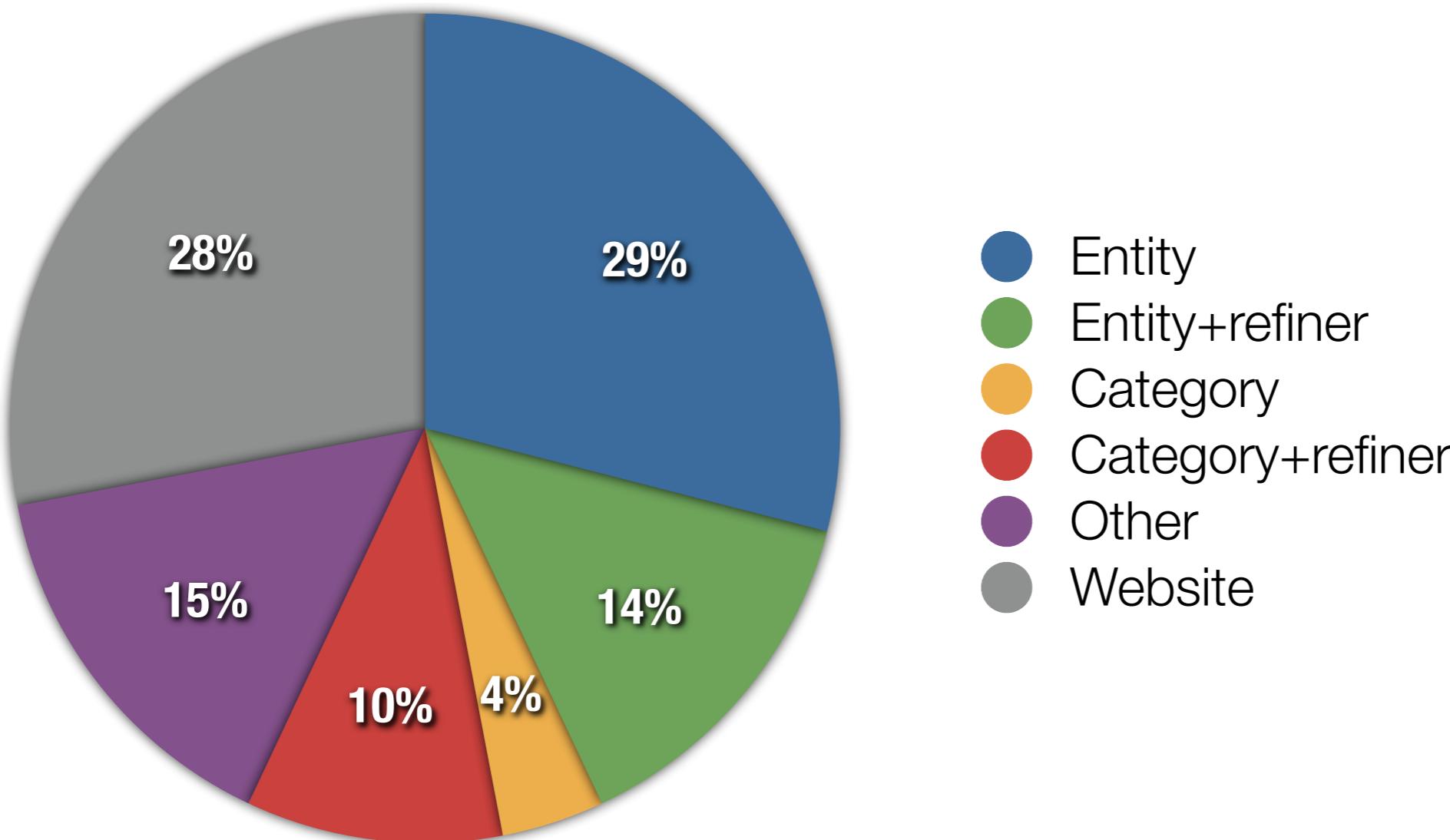
# Entity retrieval

*Addressing information needs that are better answered by returning specific objects instead of just any type of documents.*

# Distribution of web search queries [Pound et al. 2010]



# Distribution of web search queries [Lin et al. 2011]



# What's so special here?

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

# In this part

- Look at a number of entity ranking tasks
  - Motivating real-world use-cases
  - Abstractions at evaluation benchmarking campaigns (TREC, INEX)
  - Methods and approaches
- In all cases
  - Input: (semi-structured) query
  - Output: ranked list of entities
  - Evaluation: standard IR metrics

# Outline

- 1.Ranking based on entity descriptions
- 2.Incorporating entity types
- 3.Entity relationships

Attributes  
(/Descriptions)

Type(s)

Relationships

# **Setting boundaries**

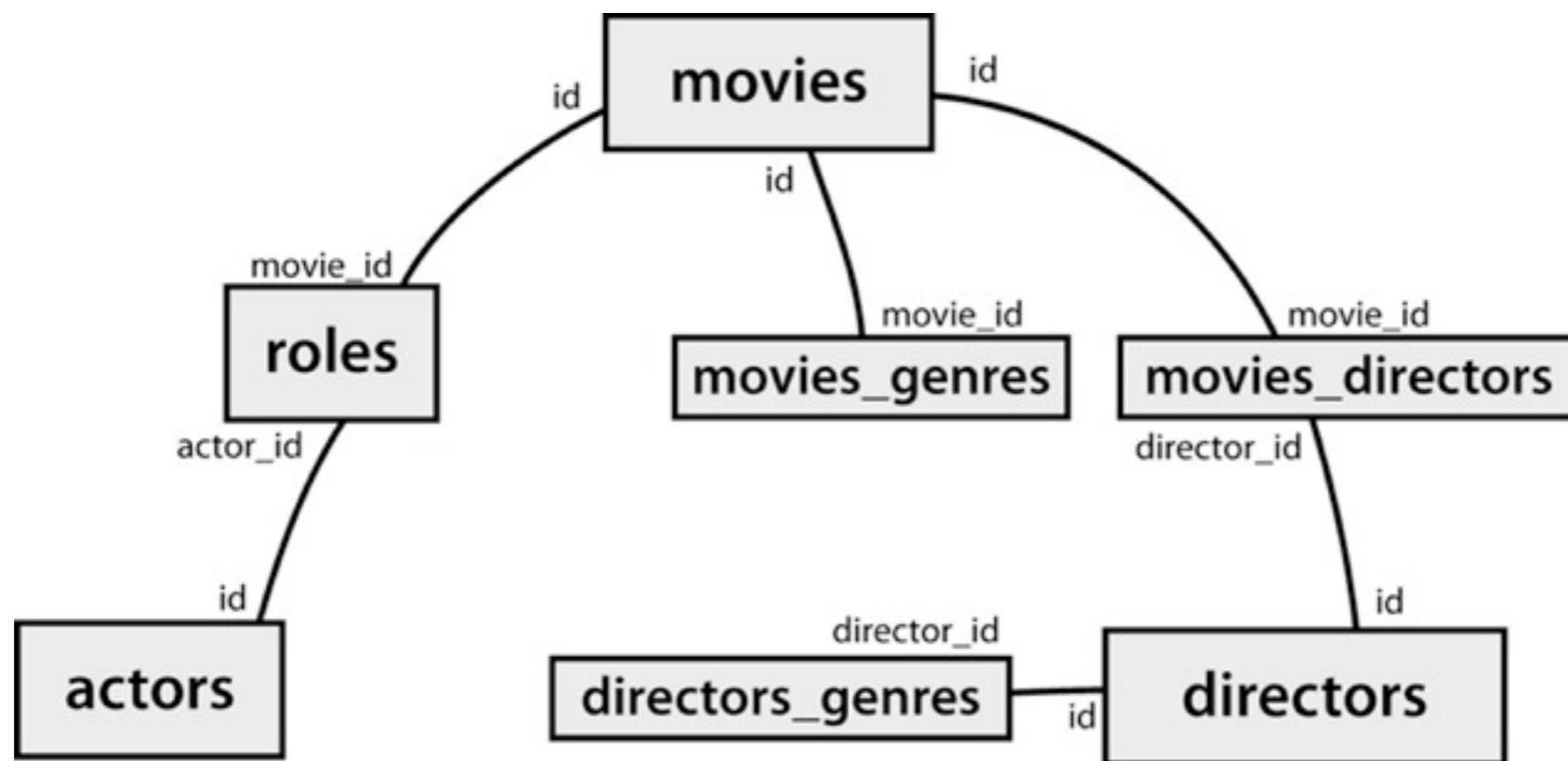


# We are IR people ...

- ... but that doesn't mean that we are the only ones who thought about this
- Entity retrieval is an active research area in neighbouring communities
  - Databases
  - Semantic web
  - Natural language processing

# Databases

- Keyword search in DBs
  - Return tuples with matching keywords, minimally joined through primary-foreign key relationships



# Databases (2)

 **SPARK**  
searching, probing & ranking

IMDB      Results 1 - 10 for "tom cruise nicole kidman"

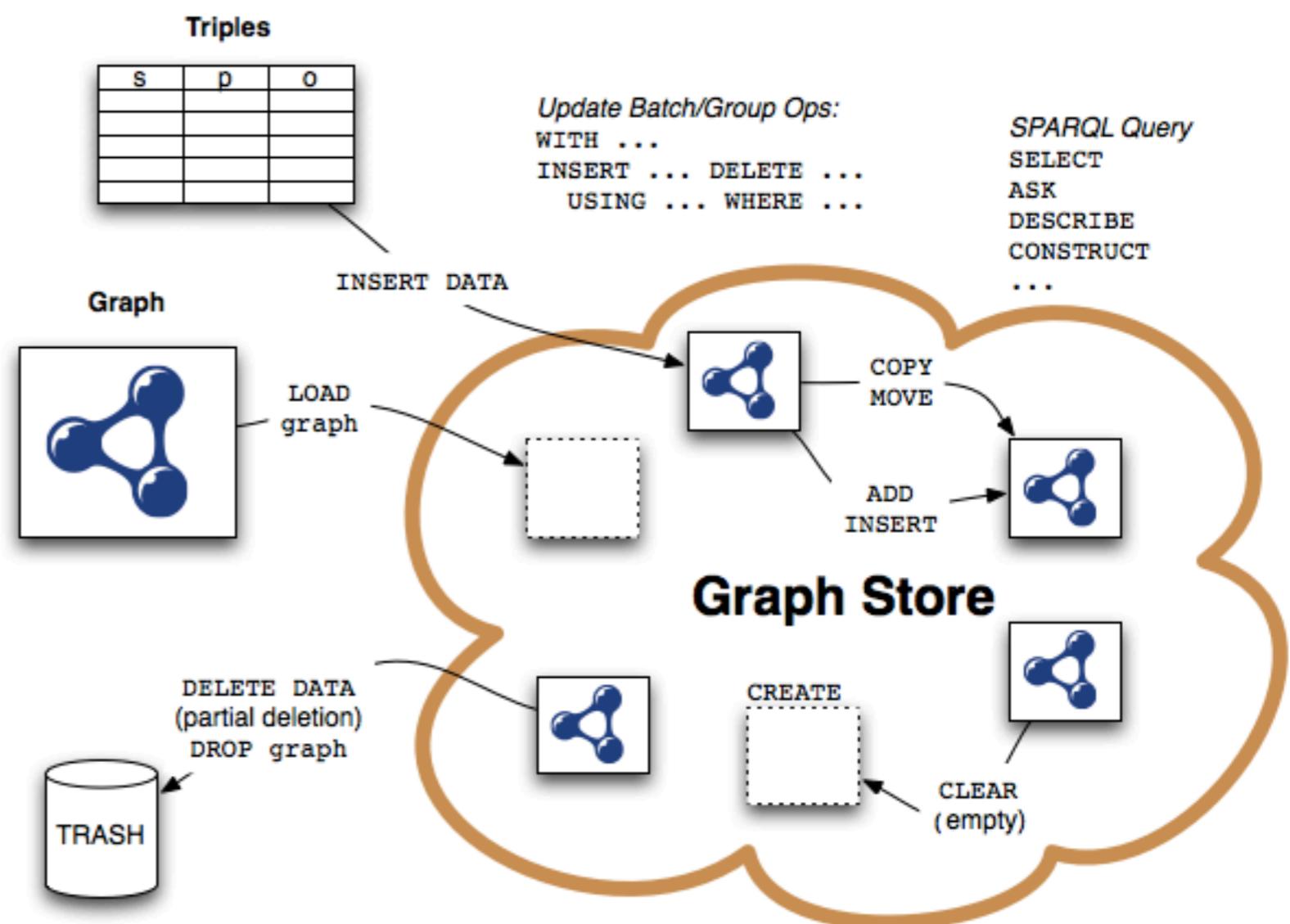
DBLP    IMDb

[Preferences](#)

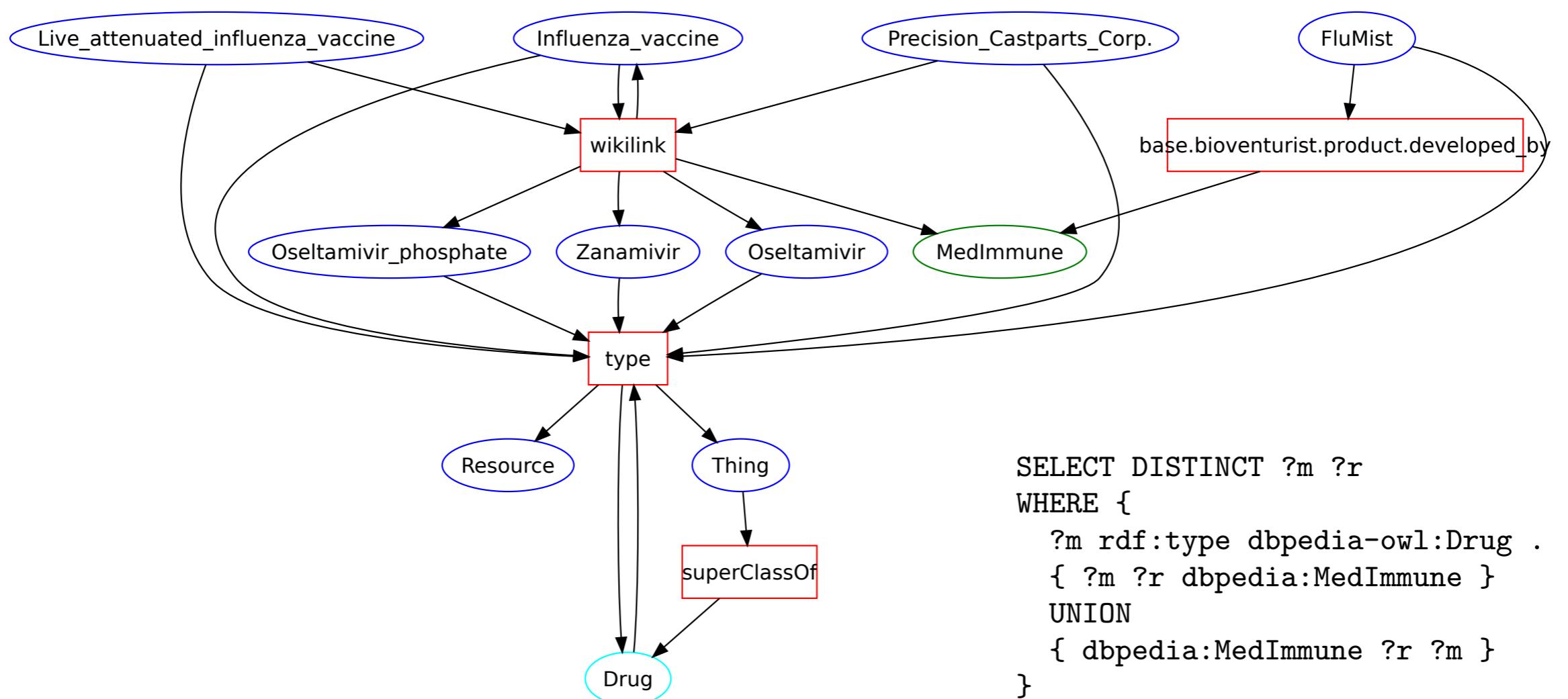
<b>actresses</b> : Name: <b>kidman, nicole</b> Id: 206700	54.08	
<b>actressplay</b> : Charactor: Herself Id: 1053008		
<b>movies</b> : Name: "Love Chain" (2003) { <b>tom cruise</b> (#1.4)} Id: 663077		
<b>actors</b> : Name: <b>cruise, tom</b> Id: 142703	31.02	
<b>actorplay</b> : Charactor: Himself Id: 841769		
<b>movies</b> : Name: 101 Most Shocking Moments in Entertainment (2003) (TV) Id: 1318		
<b>actressplay</b> : Charactor: Herself Id: 1052857		
<b>actresses</b> : Name: <b>kidman, nicole</b> Id: 206700		
<b>actors</b> : Name: <b>cruise, tom</b> Id: 142703	31.02	
<b>actorplay</b> : Charactor: Himself Id: 841840		
<b>movies</b> : Name: Road to the Red Carpet (2002) (TV) Id: 344540		
<b>actressplay</b> : Charactor: Herself Id: 1052938		
<b>actresses</b> : Name: <b>kidman, nicole</b> Id: 206700		
<b>actors</b> : Name: <b>cruise, tom</b> Id: 142703	31.02	
<b>actorplay</b> : Charactor: Himself Id: 841842		
<b>movies</b> : Name: Shirtless: Hollywood's Sexiest Men (2002) (TV) Id: 369194		
<b>actressplay</b> : Charactor: Herself Id: 1052942		
<b>actresses</b> : Name: <b>kidman, nicole</b> Id: 206700		
<b>actors</b> : Name: <b>cruise, tom</b> Id: 142703	31.02	
<b>actorplay</b> : Charactor: Himself (Narrator) Id: 841844		
<b>movies</b> : Name: Stanley Kubrick: A Life in Pictures (2001) Id: 386247		
<b>actressplay</b> : Charactor: Herself Id: 1052946		
<b>actresses</b> : Name: <b>kidman, nicole</b> Id: 206700		

# Semantic web

- Indexing
- Retrieval
- Querying
- Inference



# Semantic web (2)

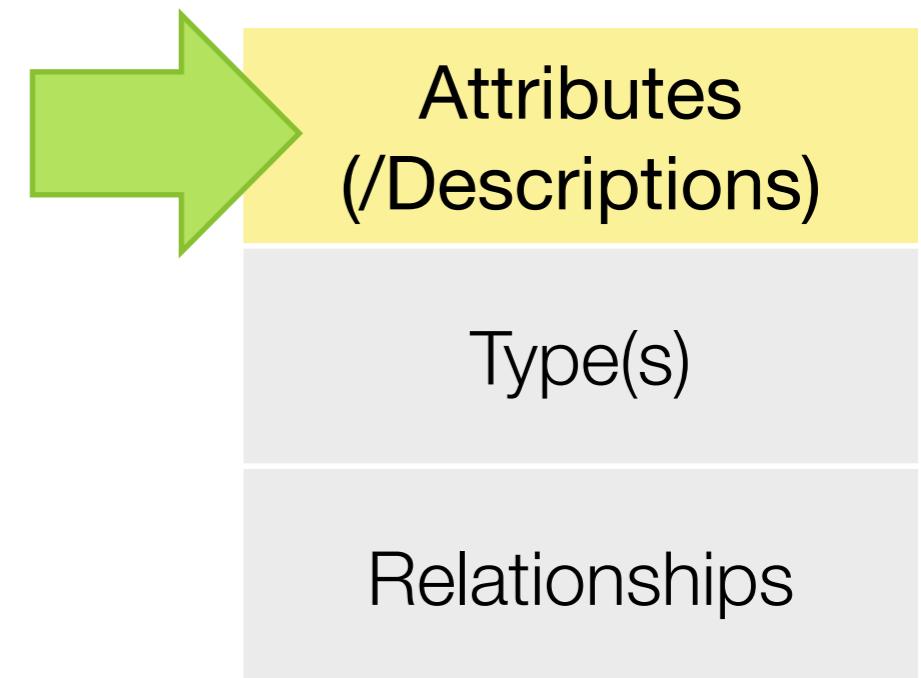


```
SELECT DISTINCT ?m ?r
WHERE {
  ?m rdf:type dbpedia-owl:Drug .
  { ?m ?r dbpedia:MedImmune }
  UNION
  { dbpedia:MedImmune ?r ?m }
}
```

# Natural language processing

- Question answering
  - “Who invented the paper clip?”
  - “What museums have displayed Chanel clothing?”
- Relationship extraction

# **Ranking based on entity descriptions**



# **Task: 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

# Example information needs

 american embassy nairobi

 ben franklin

 Chernobyl

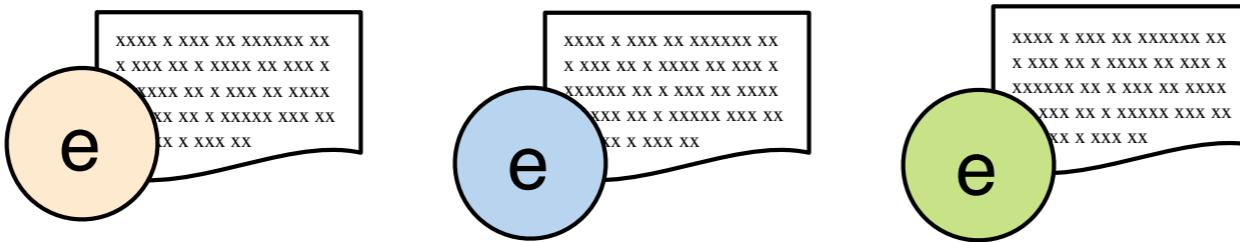
 meg ryan war

 Worst actor century

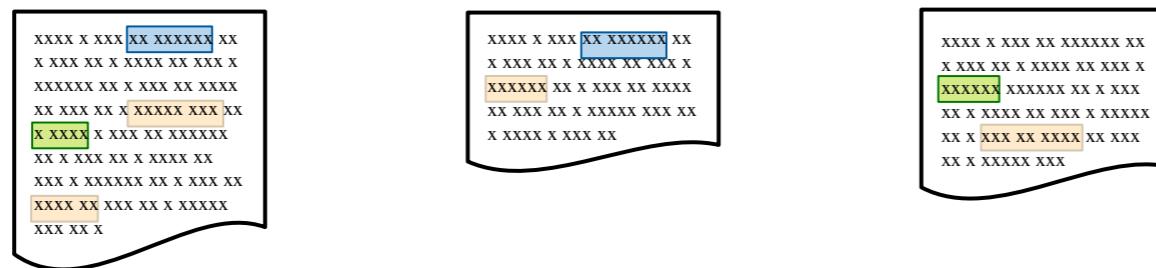
 Sweden Iceland currency

# Two settings

## 1. With ready-made entity descriptions



## 2. Without explicit entity representations



# **Ranking with ready-made entity descriptions**

# This is not unrealistic...

The image shows a composite screenshot of three overlapping web pages:

- Wikipedia (left):** A sidebar with links like Main page, Navigation, Interaction, and Toolbox.
- IMDb (top center):** A search bar and navigation menu for movies, TV, news, etc.
- LinkedIn (center):** A user profile for Krisztian Balog, showing activity and connections.
- Amazon (right):** A product page for "Modern Information Retrieval: The Concepts and Technology behind Search (2nd Edition)" by Ricardo Baeza-Yates and Berthier Ribeiro-Neto. It shows a price of \$62.49, a "Buy New" button, and a "Sell Us Your Item" section.

The LinkedIn and Amazon pages are heavily overlaid, suggesting simultaneous use of multiple online platforms.

# **Document-based entity representations**

- Most entities have a “home page”
- I.e., each entity is described by a document
- In this scenario, ranking entities is much like ranking documents
  - unstructured
  - semi-structured

# Evaluation initiatives

- INEX Entity Ranking track (2007-09)
  - Collection is the (English) Wikipedia
  - Entities are represented by Wikipedia articles
- Semantic Search Challenge (2010-11)
  - Collection is a Semantic Web crawl (BTC2009)
    - ~1 billion RDF triples
  - Entities are represented by URIs
- INEX Linked Data track (2012-13)
  - Wikipedia enriched with RDF properties from DBpedia and YAGO

# Standard Language Modeling approach

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

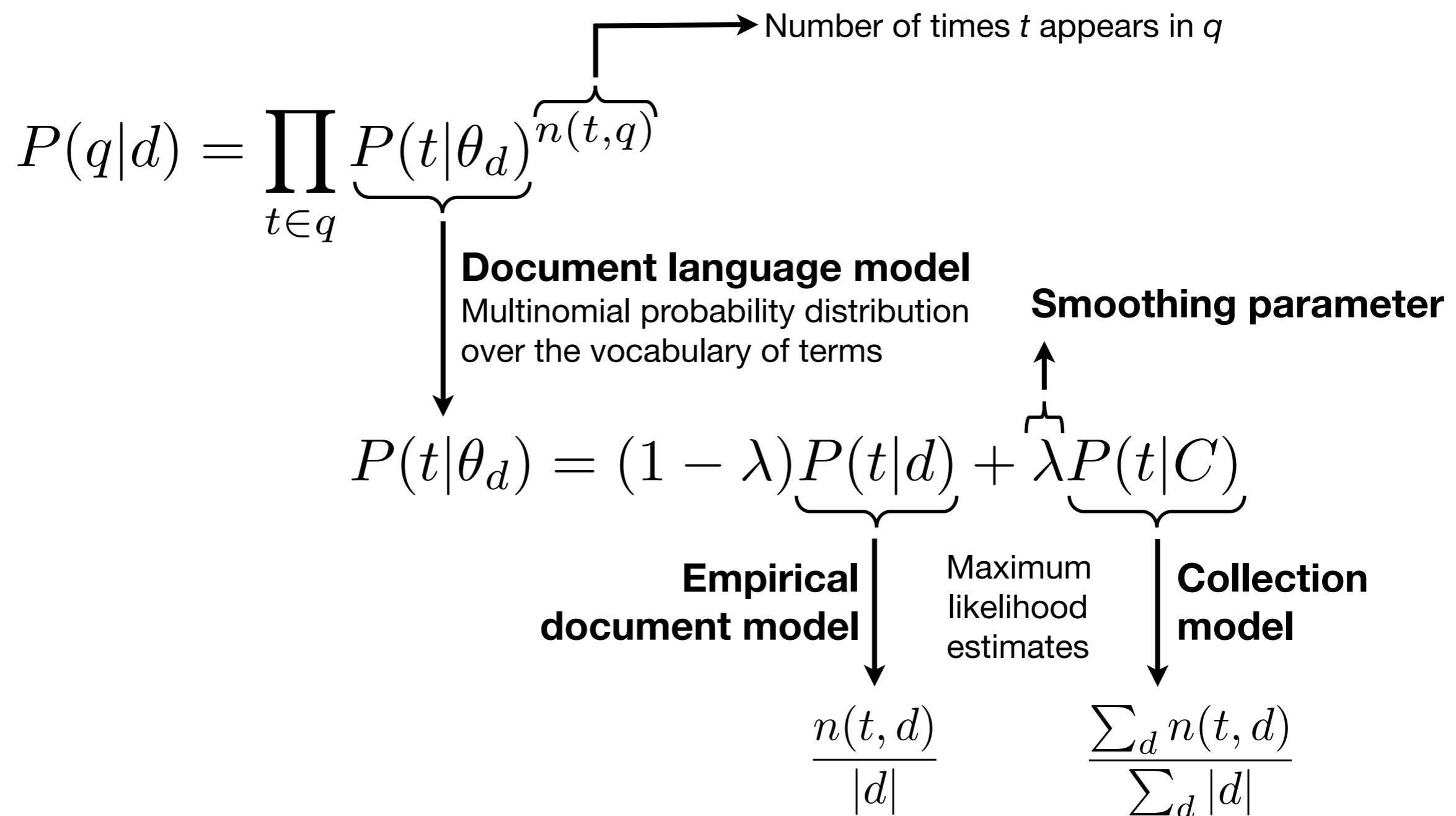
$$P(d|q) = \frac{P(q|d)P(d)}{P(q)} \propto P(q|d)P(d)$$

**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

$$P(e|q) \propto P(e)P(q|\theta_e) = \underbrace{P(e)}_{\text{Entity prior}} \prod_{t \in q} \underbrace{P(t|\theta_e)^{n(t,q)}}_{\text{Entity language model}}$$

**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



# 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.<sup>[2]</sup>

**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 [...]

**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

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

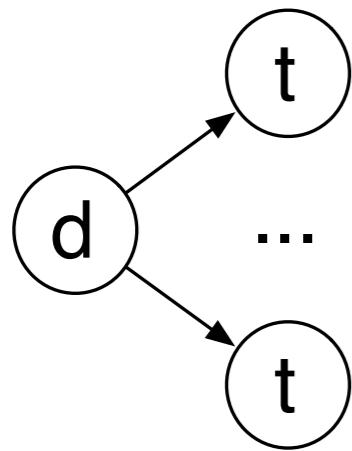
**Field weights**

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

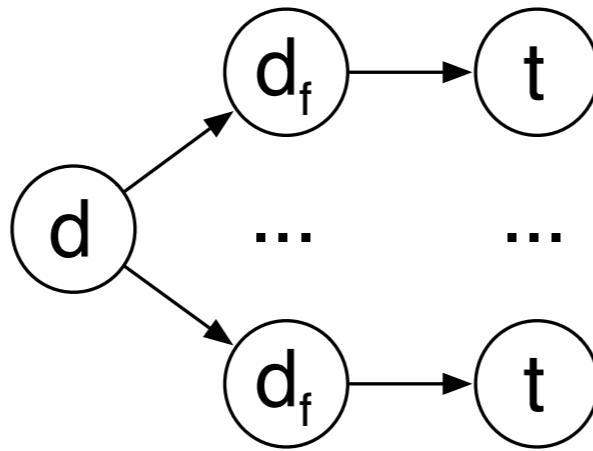
**Field language model**

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

# Comparison of models



**Unstructured  
document model**



**Fielded  
document model**

# 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 (BM25F and)
  - type **[Pérez-Agüera et al. 2010]**
  - manually determined importance **[Blanco et al. 2011]**

# Hierarchical Entity Model

[Neumayer et al. 2012]

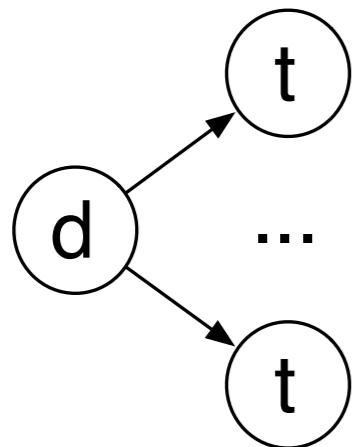
- Organize fields into a 2-level hierarchy
  - Field types (4) on the top level
  - Individual fields of that type on the bottom level
- Estimate field weights
  - Using training data for field types
  - Using heuristics for bottom-level types

# Two-level hierarchy

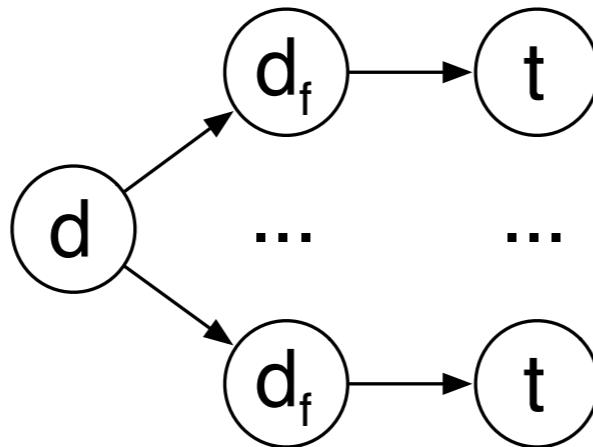
[Neumayer et al. 2012]

<b>Name</b>	{	foaf:name rdfs:label rdfs:comment	Audi A4 Audi A4 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 [...]
<b>Attributes</b>	{	dbpprop:production	1994 2001 2005 2008
<b>Out-relations</b>	{	rdf:type dbpedia-owl:manufacturer dbpedia-owl:class owl:sameAs	dbpedia-owl:MeanOfTransportation dbpedia-owl:Automobile dbpedia:Audi dbpedia:Compact_executive_car freebase:Audi_A4
<b>In-relations</b>	{	is dbpedia-owl:predecessor of is dbpprop:similar of	dbpedia:Audi_A5 dbpedia:Cadillac_BLS

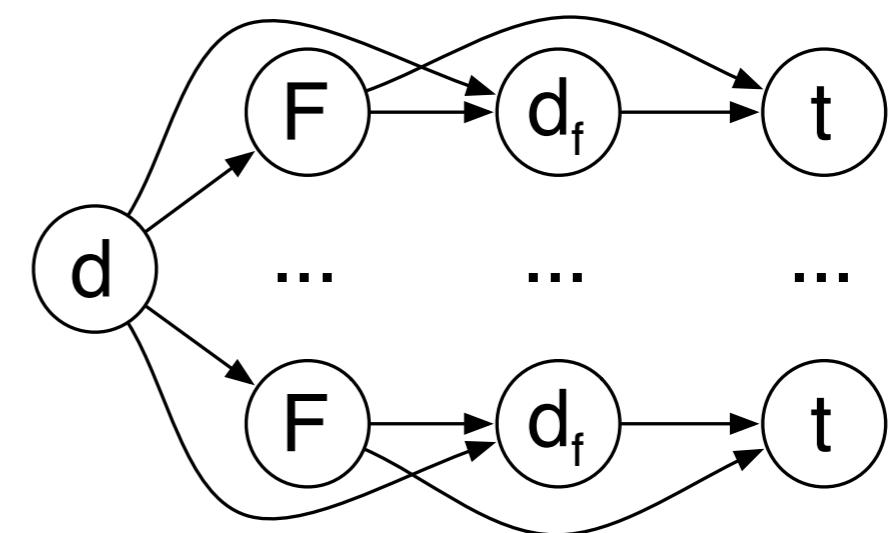
# Comparison of models



**Unstructured  
document model**



**Fielded  
document model**



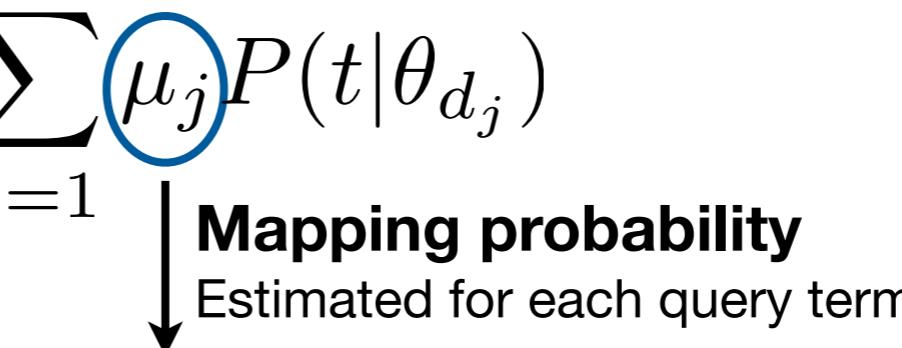
**Hierarchical  
document model**

# 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

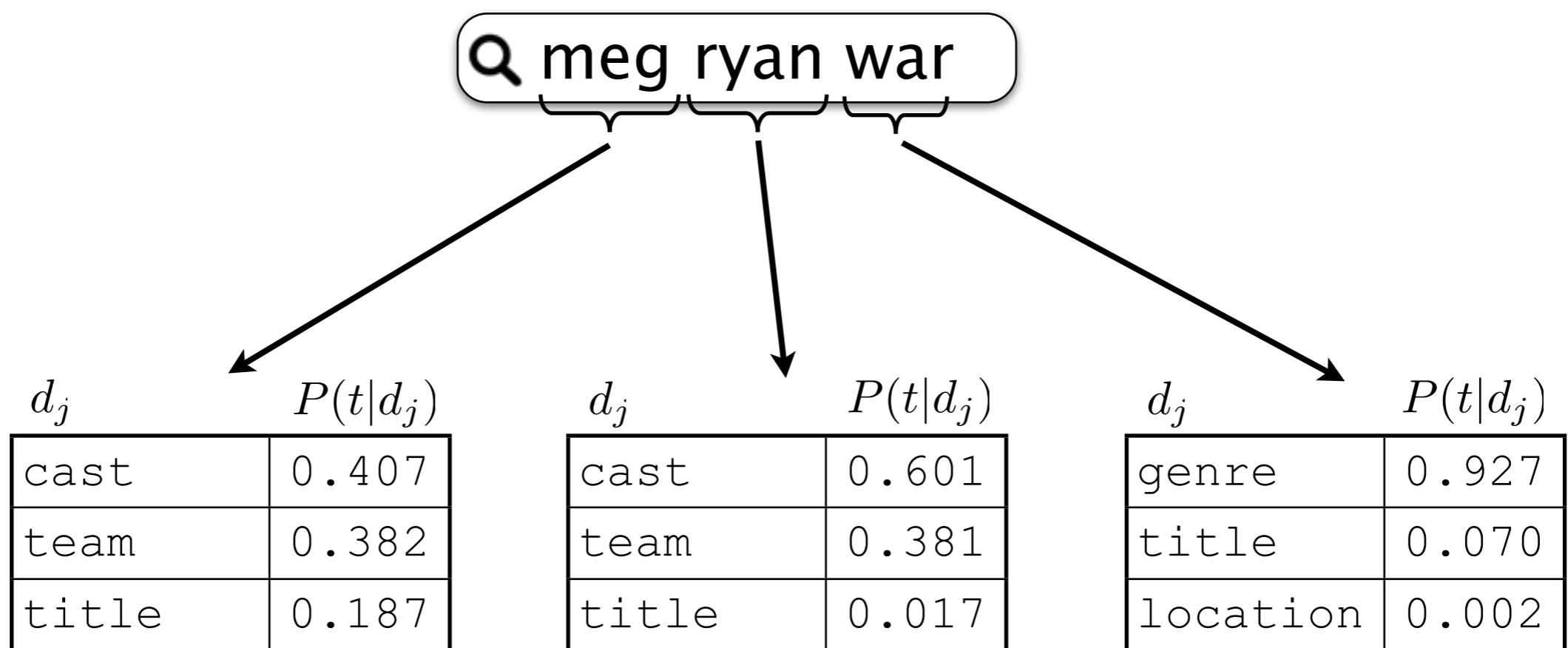
$$P(t|C_j) = \frac{\sum_d n(t, d_j)}{\sum_d |d_j|}$$

**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

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

# Example



# **Ranking without explicit entity representations**

# Scenario

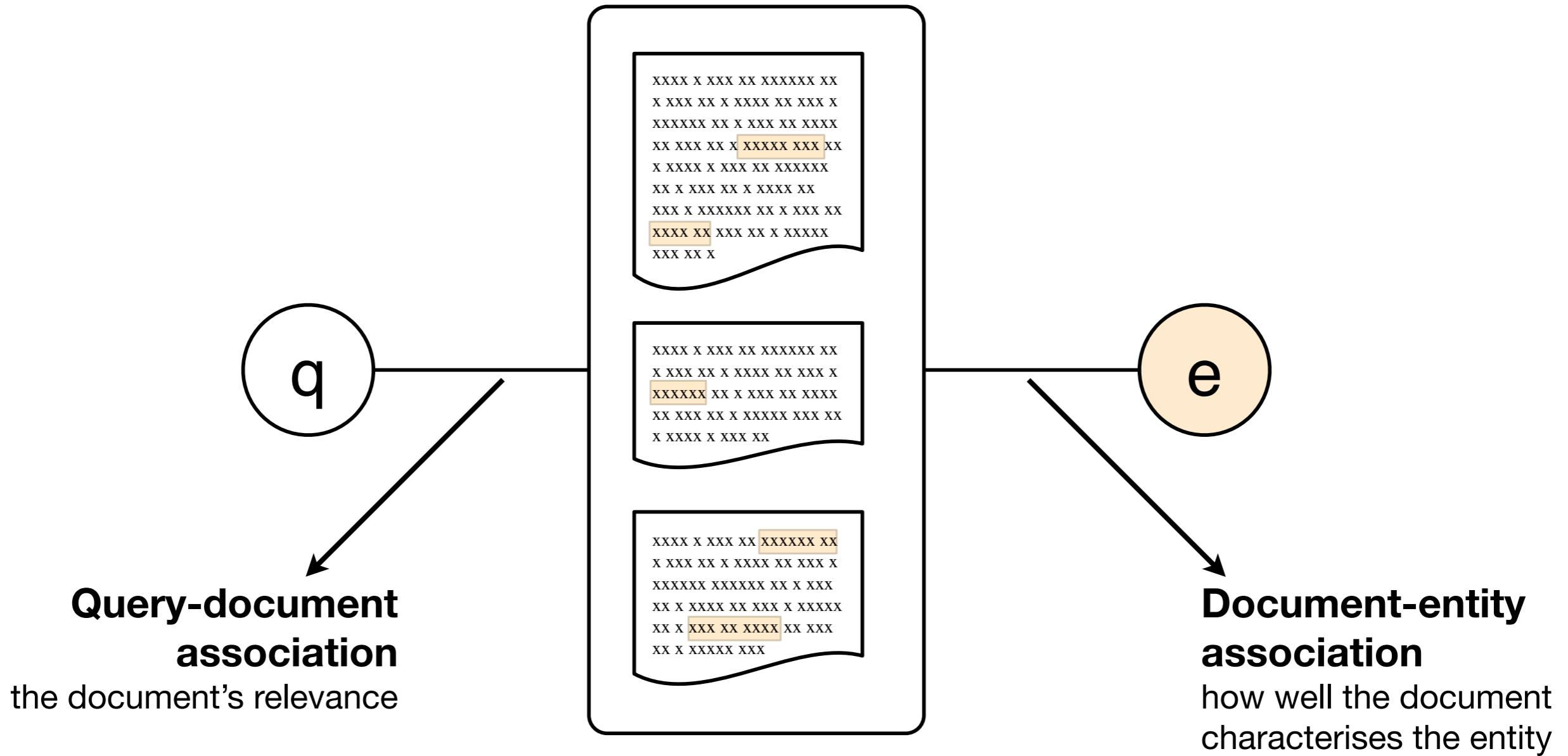
- Entity descriptions are not readily available
- Entity occurrences are annotated
  - manually
  - automatically (~entity linking)

# TREC Enterprise track

- Expert finding task (2005-08)
  - Enterprise setting (intranet of a large organization)
  - Given a query, return people who are experts on the query topic
  - List of potential experts is provided
- We assume that the collection has been annotated with <person>...</person> tokens

# The basic idea

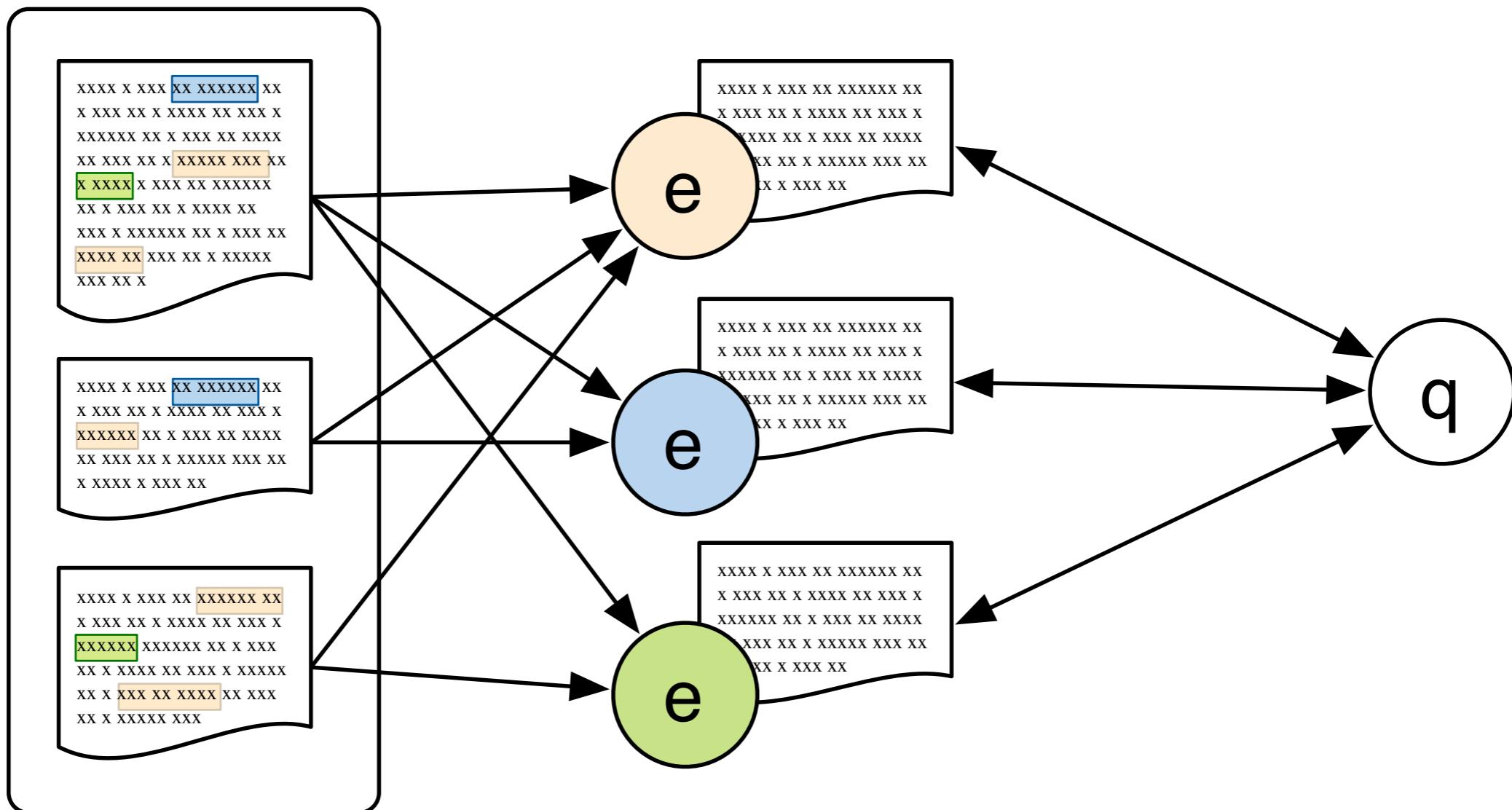
Use documents to go from queries to entities



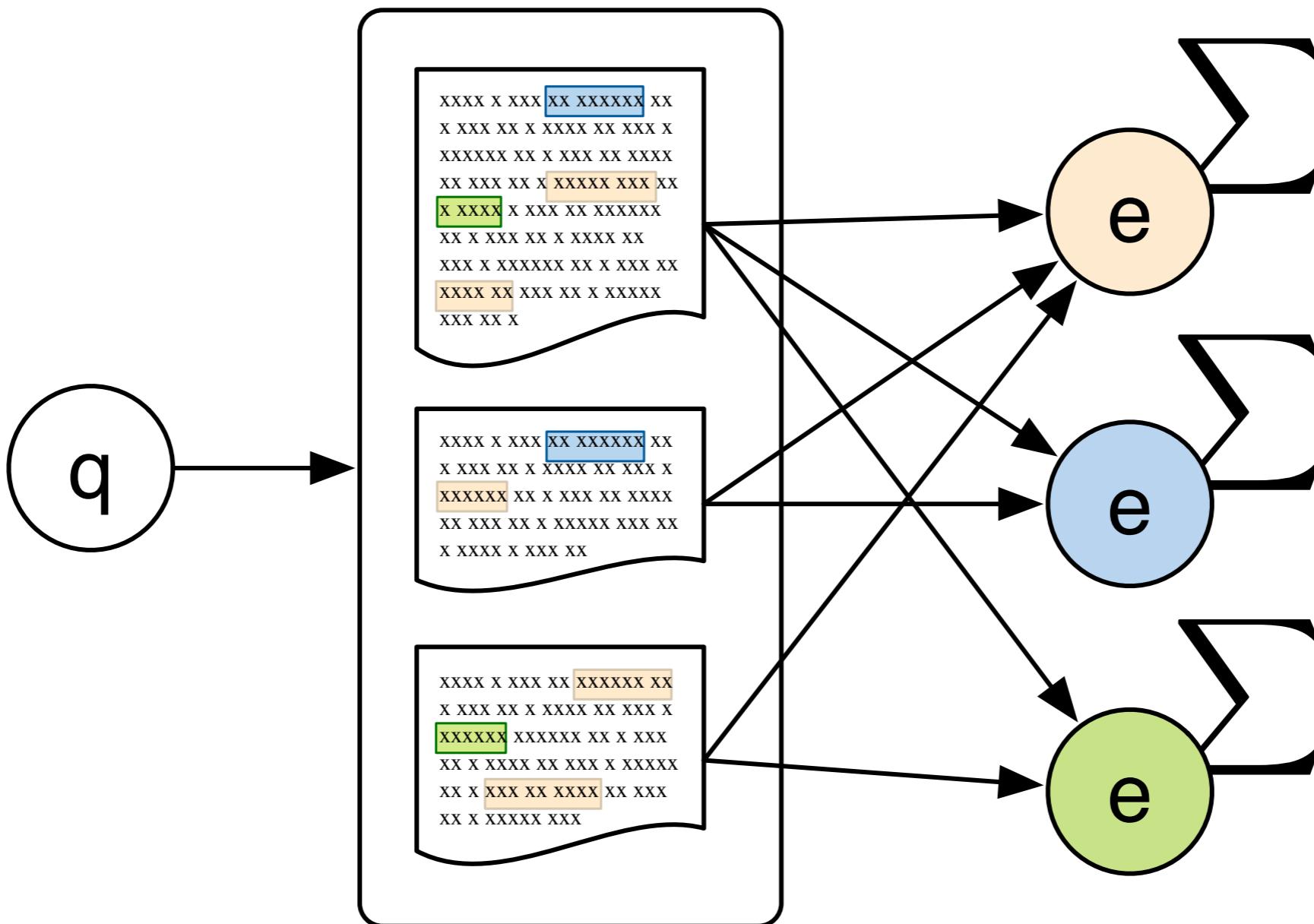
# Two principal approaches

- **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

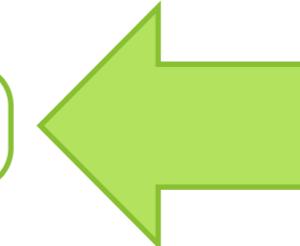


# **Many possibilities in terms of modeling**

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

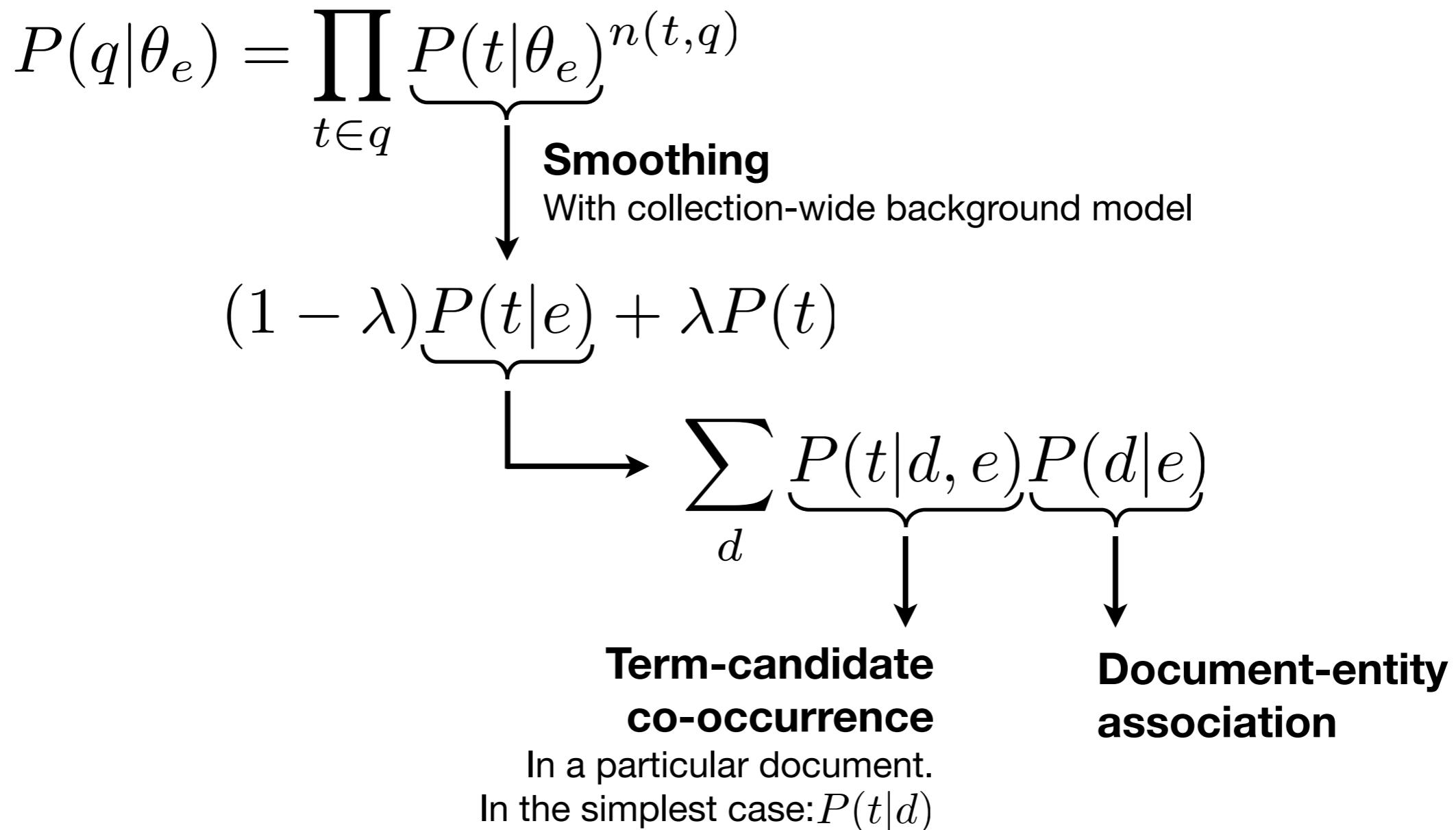
# Generative probabilistic models

- 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]



# Document models (“Model 2”)

[Balog et al. 2006]

$$P(q|e) = \sum_d P(q|d, e) P(d|e)$$

**Document relevance**  
How well document  $d$  supports the claim that  $e$  is relevant to  $q$

**Document-entity association**

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

( $t$  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]

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

**Normalizing  
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

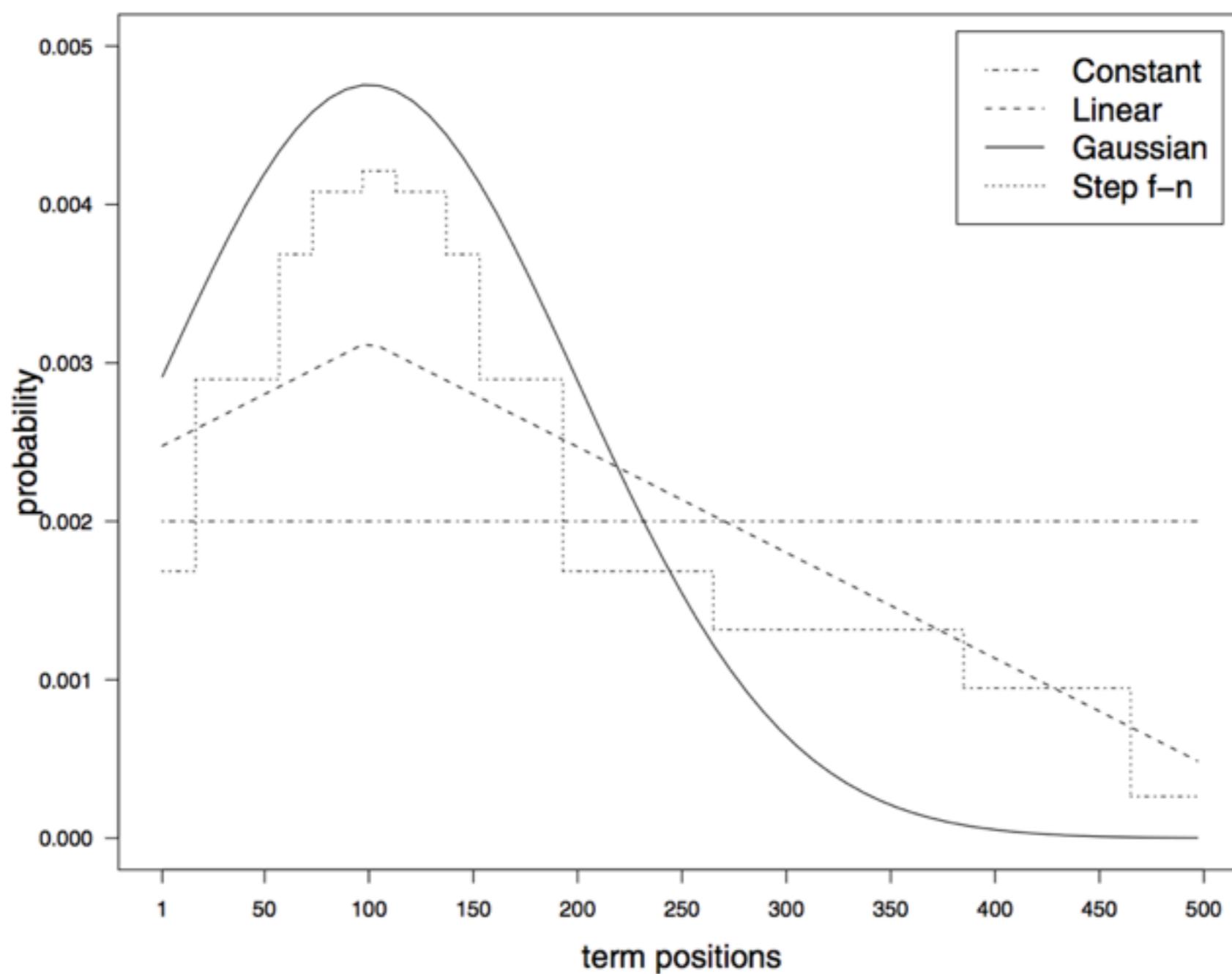


Figure taken from D. Petkova and W.B. Croft. **Proximity-based document representation for named entity retrieval.** CIKM'07.

# **Many possibilities in terms of modeling**

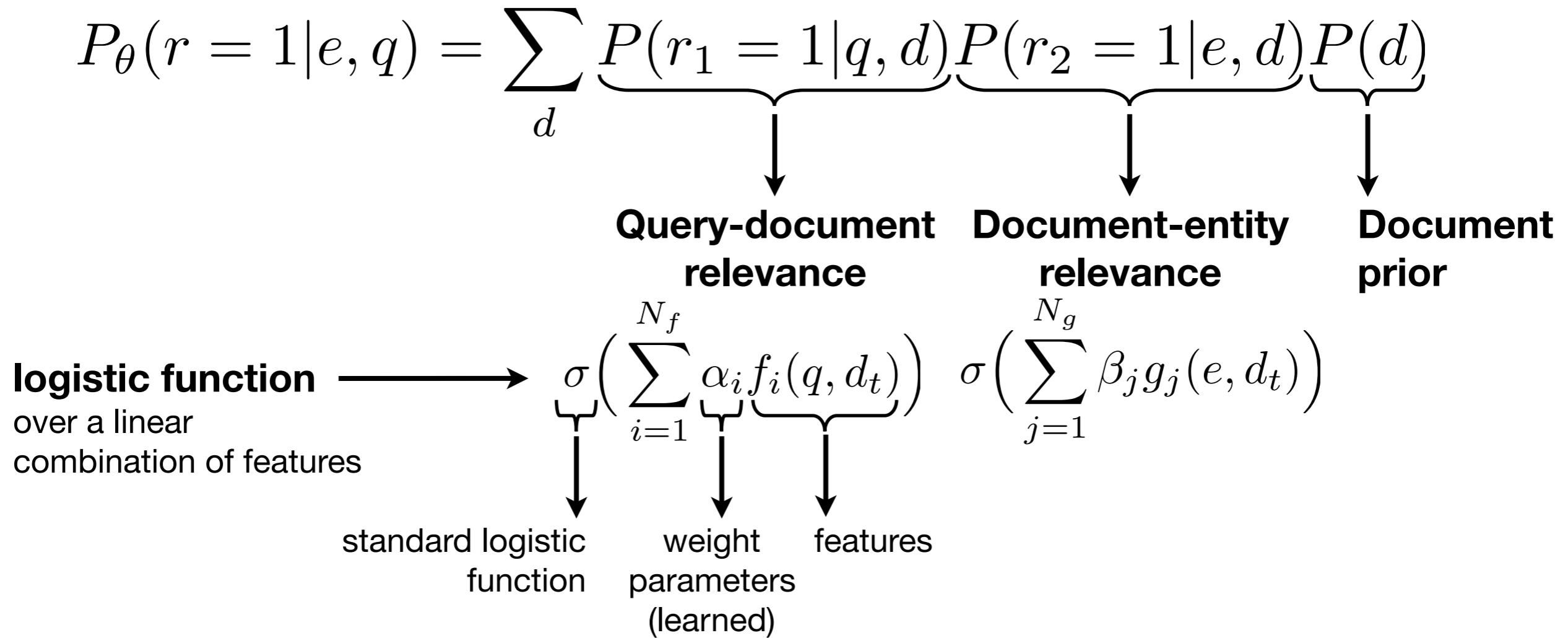
- Generative probabilistic models
- Discriminative probabilistic models
- Voting models
- Graph-based models

# 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  $P(e,q|r=1)$ )
  - Optimization can get trapped in a local maximum/minimum

# Arithmetic Mean Discriminative (AMD) model

[Yang et al. 2010]



# Learning to rank && entity retrieval

- 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]**

# 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

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

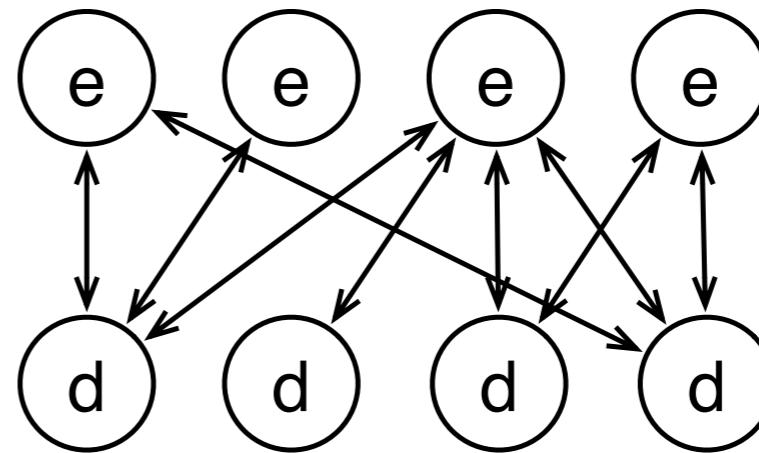
# Graph-based models

[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

[Serdyukov et al. 2008]

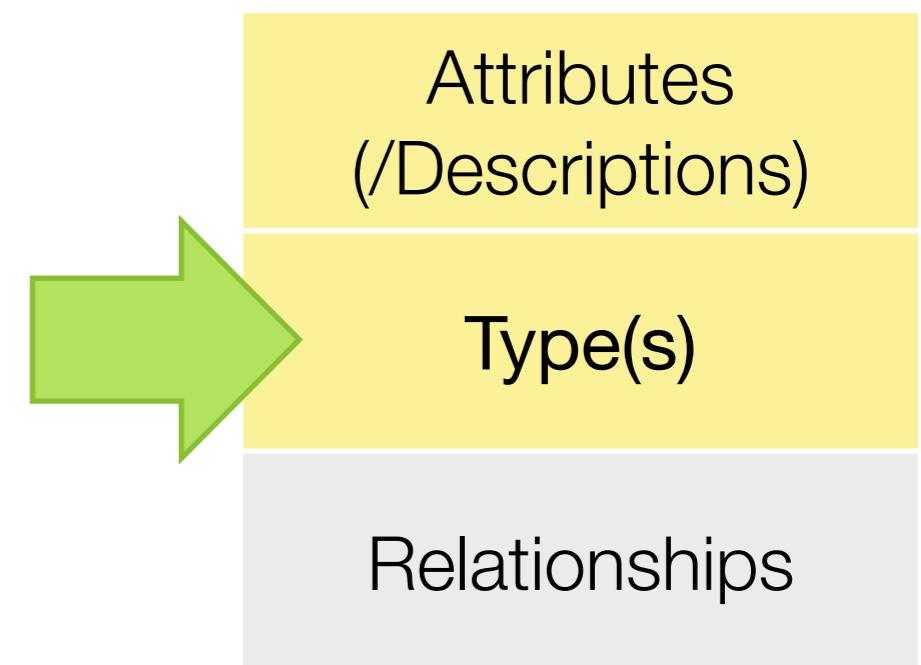


$$P_i(d) = \lambda P_J(d) + (1 - \lambda) \sum_{e \rightarrow d} P(d|e) P_{i-1}(e),$$

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

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

# Incorporating entity types



# Entities are typed...

The image displays two web pages side-by-side, illustrating the concept of entity typing.

**Left Entity (Amazon.com):** A screenshot of the Amazon.com homepage. The top navigation bar includes links for "Krisztian's Amazon.com", "Today's Deals", "Gift Cards", "Sell", and "Help". A banner for "FREE TWO-DAY SHIPPING FOR COLLEGE STUDENTS" is visible. The main content area features a "Search" bar with dropdown options like "Shop by Department", "All", and "Go". Below the search bar, a section titled "EARTH'S BIGGEST SELECTION" lists categories such as "Unlimited Instant Videos", "MP3s & Cloud Player", "Amazon Cloud Drive", and "Kindle". On the right, there's a sidebar for "Appstore" and "Digital Content".

**Right Entity (Wikipedia):** A screenshot of the Wikipedia page "Category:Main topic classifications". The page title is "Category:Main topic classifications" and it is described as "From Wikipedia, the free encyclopedia". The page content lists major topic classifications, such as Agriculture, Arts, Business, etc., each with a count of categories and pages. A sidebar on the right contains a "Wikimedia Commons has media related to: Topics" link. At the bottom, there are sections for "Subcategories" and "Categories: Articles".

# If target type is not provided

## Rely on techniques from...

- Federated search
  - Obtain a separate ranking for each type of entity, then merge **[Kim & Croft 2010]**
- Aggregated search
  - Return top ranked entities from each type **[Lalmas 2011]**

# Example (1)

## Mixing all types together (“federated search”)

Freebase Find... Browse Query Help Sign In or Sign Up English ▾

Search **magnum**

any /common/topic Constrain results by type... Options: Scoring entity |  Prefixed

### Search Results

 **Magnum, P.I.** /m/01pj8d  
Action/Adventure TV Program, TV Program, Award-Nominated Work, Netflix Title, Award-Winning Work  
program creator: Donald Bellisario, Glen A. Larson  
network: CBS  
episode running time: 60, 45  
number of seasons: 8  
number of episodes: 157  
alias: Magnum

 **Magnum Photos** /m/01szqd  
Photography Organization, Literature Subject, Organization, Employer  
founders: Robert Capa, Henri Cartier-Bresson  
legal structure: Cooperative

 **.357 Magnum** /m/02mc3x

 **Magnum** /m/06rfd4  
Hard rock Artist, Musical Group, Musical Artist, Person or entity appearing in film, Social network user  
origin: Birmingham  
genre: Rock music, Hard rock, Progressive rock, Melodic Rock

# Example (2)

## Grouping results by entity type (“aggregated search”)

The screenshot shows a LinkedIn search results page for the query "best". The search bar at the top contains "best". The results section displays 4,183,729 results for "best". The results are grouped into several categories:

- Christoph Best** (2nd): Computational Scientist at Google, London, United Kingdom - Information Technology and Services. Includes a "Connect" button.
- Angelina Best** (2nd): Enterprise Sales Manager at Microsoft, Amsterdam Area, Netherlands - Information Technology and Services. Includes a "Connect" button.
- Clive Best** (2nd): Director OSVISION, Varese Area, Italy - Internet. Includes a "Connect" button.
- Companies for best**:
  - Best Advisors Network: Accounting - 1-10 employees
  - mCentric: Telecommunications - 11-50 employees
  - Event Industry Awards: Events Services - 1-10 employees
- Hubert Best** (2nd): Owner, ENN Advokatbyrå, Stockholm, Sweden - Law Practice. Includes a "Connect" button.
- Eric de Best** (2nd): Owner, cockpits.nl, The Hague Area, Netherlands - Arts and Crafts. Includes a "Connect" button.
- Jobs for best**:
  - Administrative Information Management Advisor

A sidebar on the right, titled "Spotlight expert", lists results categorized by type:

- Top Hit**: Makefile\_expertApp — expSearch
- Applications**: Makefile\_expertApp — expSearch, Makefile\_expertApp — lm5
- Documents**: expert — spider-url, expert — filtertest-url, expert\_product.tpl
- Folders**: expert, site-expert
- Messages**: [SIG-IRList] ECIR 2014: Second Call for Papers, Your connection Gyula Berke has endorse...
- Events**: II, lecture on Expert Search, Expert Search Skype w/ Doug Oard & Fab..., TrendLight
- Images**: expert.jpg — 2006-06-28 18\_23, expert.jpg — 2006-02-01 06\_45
- PDF Documents**: ir-evaluation-usefulness-Alonso-brixen-..., Expert Finding Entity Search [ECIR2012].pdf, expert\_survey.pdf
- Presentations**: www2013-entityretrieval, workshop\_welcome, uva\_er\_meeting.ppt

# Often, users provide target types explicitly

The image displays two search interfaces side-by-side. On the left is the eBay search interface, featuring a sidebar with categories like Fashion, Parts & accessories, Electronics, Collectibles & art, Home & garden, Women's Clothing, Jewelry & watches, and Daily deals. Below the sidebar is a search bar with the word "magnum" typed in. Underneath the search bar are dropdown menus for "any" and "/common/topic", and a checkbox for "Constrain results by type...". At the bottom, there are options for "Scoring" and "Prefixed". On the right is the OS X Spotlight search interface, showing a list of search results for "Apress.Pro". A dropdown menu titled "Kind" is open, listing various file types: Any, Application, Document, Executable, Folder, Image, Movie, Music, PDF, Presentation, Text, and Other. To the right of the Spotlight interface is a sidebar titled "Departments" which lists various product categories.

**Departments**

- Grocery & Gourmet Food
- Energy Drinks

**Clothing & Accessories**

- Men's Keyrings & Keychains
- Novelty T-Shirts
- Novelty & Special Use Clothing
- Men's Fashion Hoodies & Sweatshirts

**Automotive**

- Racing Apparel
- Motorcycle Protective Coats & Vests
- Decals
- Motorcycle & ATV Helmets
- Motorcycle & ATV Graphics
- Towing Winches
- Key Chains

**Tools & Home Improvement**

- Wall Stickers & Murals
- Diversion Safes

**Sports & Outdoors**

- Sports Fan Clothing
- + See more...

**Computers & Accessories**

- USB Flash Drives

+ See All 33 Departments

# Type-aware entity ranking

- Assume that the user provides target type(s)
- Challenges
  - Target type information is imperfect
    - Users are not familiar with the classification system
  - Categorisation of entities is imperfect
    - Entity might belong to multiple categories
      - E.g. is King Arthur “British royalty”, “fictional character”, or “military person”?
  - Types can be hierarchically organised
    - Although it may not be a strict “is-a” hierarchy

# INEX Entity Ranking track

- Entities are represented by Wikipedia articles
- Topic definition includes target categories



Movies with eight or more Academy Awards  
best picture oscar british films american films

# Titanic (1997 film)

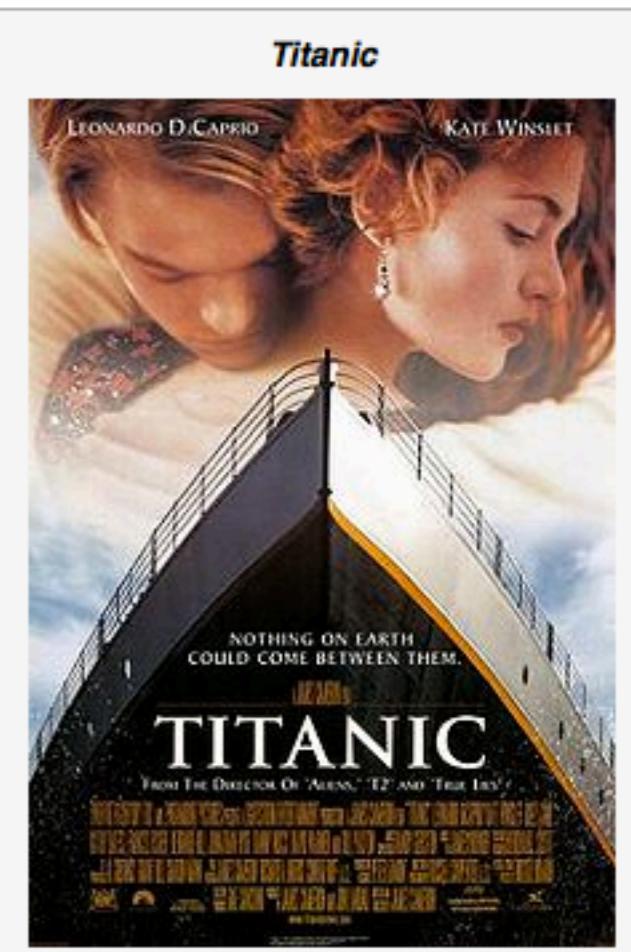


From Wikipedia, the free encyclopedia

**Titanic** is a 1997 American epic romance and disaster film directed, written, co-produced, and co-edited by James Cameron. A fictionalized account of the sinking of the RMS *Titanic*, it stars Leonardo DiCaprio as Jack Dawson and Kate Winslet as Rose DeWitt Bukater, members of different social classes who fall in love aboard the ship during its ill-fated maiden voyage. Although the central roles and love story are fictitious, some characters are based on genuine historical figures. Gloria Stuart portrays the elderly Rose, who narrates the film in a modern-day framing device, and Billy Zane plays Cal Hockley, the overbearing fiancé of the younger Rose. Cameron saw the love story as a way to engage the audience with the real-life tragedy.

Production on the film began in 1995, when Cameron shot footage of the actual *Titanic* wreck. The modern scenes were shot on board the *Akademik Mstislav Keldysh*, which Cameron had used as a base when filming the actual wreck. A reconstruction of the *Titanic* was built at Playas de Rosarito, Baja California, and scale models and computer-generated imagery were also used to recreate the sinking. The film was partially funded by Paramount Pictures and 20th Century Fox – respectively, its American and international distributor – and at the time, it was the most expensive film ever made, with an estimated budget of US\$200 million.<sup>[3][4][5][6]</sup>

The film was originally scheduled to open on July 2, 1997, however, post-production delays pushed back its release to December 19 instead.<sup>[7]</sup> *Titanic* was an enormous critical and commercial success. It was nominated for fourteen Academy Awards, eventually winning eleven, including Best Picture and Best Director.<sup>[8]</sup> It became the highest-grossing film of all time, with a worldwide gross of over \$1.8 billion, and remained so for twelve years until Cameron's next directorial effort, *Avatar*, surpassed it in 2010.<sup>[9][10]</sup> *Titanic* also has been ranked as the sixth best epic film of all time in AFI's 10 Top 10 by the American Film Institute.<sup>[11]</sup> The film is due for theatrical re-release in 2012 after Cameron completes its conversion into 3-D.<sup>[12]</sup>



Categories: 1997 films | American films | English-language films | American disaster films | Best Drama Picture Golden Globe winners | Best Picture Academy Award winners | Best Song Academy Award winners | Films directed by James Cameron | Films set in 1912 | Films that won the Best Sound Mixing Academy Award | Films that won the Best Visual Effects Academy Award | Films whose art director won the Best Art Direction Academy Award | Films whose cinematographer won the Best Cinematography Academy Award | Films whose director won the Best Director Academy Award | Films whose director won the Best Director Golden Globe | Films whose editor won the Best Film Editing Academy Award | Epic films | RMS Titanic | Romantic epic films | Romantic period films | Seafaring films based on actual events | Films shot in Nova Scotia | Films shot in Vancouver | Paramount films | 20th Century Fox films | Lightstorm Entertainment films | 2-D films converted to 3-D

# **Using target type information**

- 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]

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

$$P(q|e) = (1 - \lambda) \underbrace{P(\theta_q^T | \theta_e^T)}_{\text{Term-based representation}} + \lambda \underbrace{P(\theta_q^C | \theta_e^C)}_{\text{Category-based representation}}$$

Term-based representation

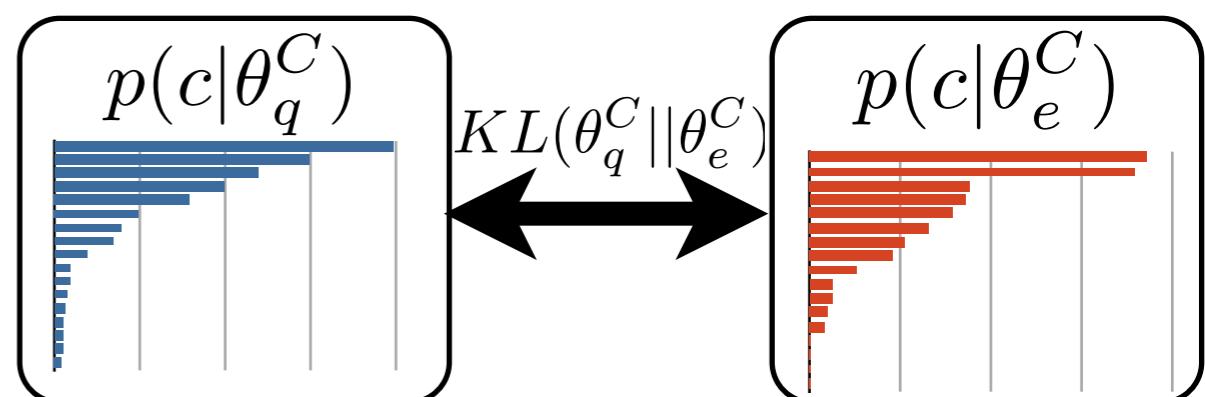
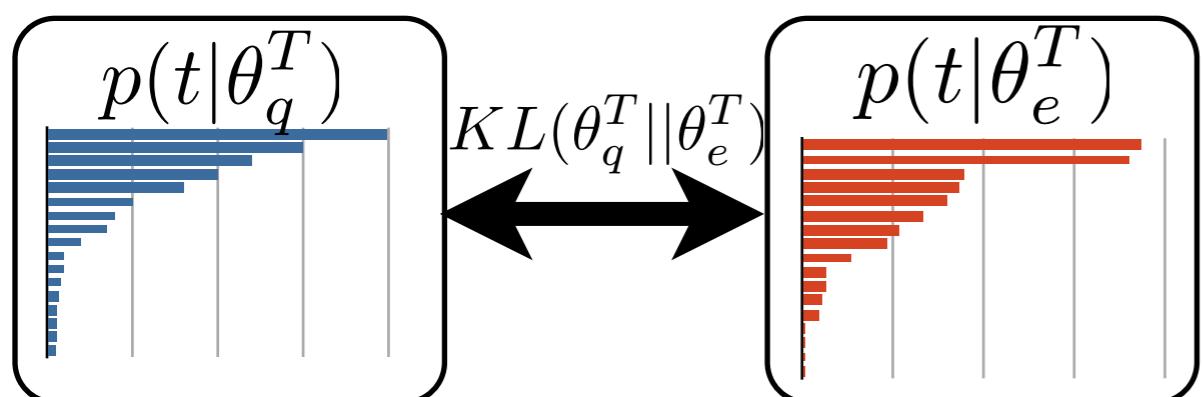
Query model

Entity model

Category-based representation

Query model

Entity model



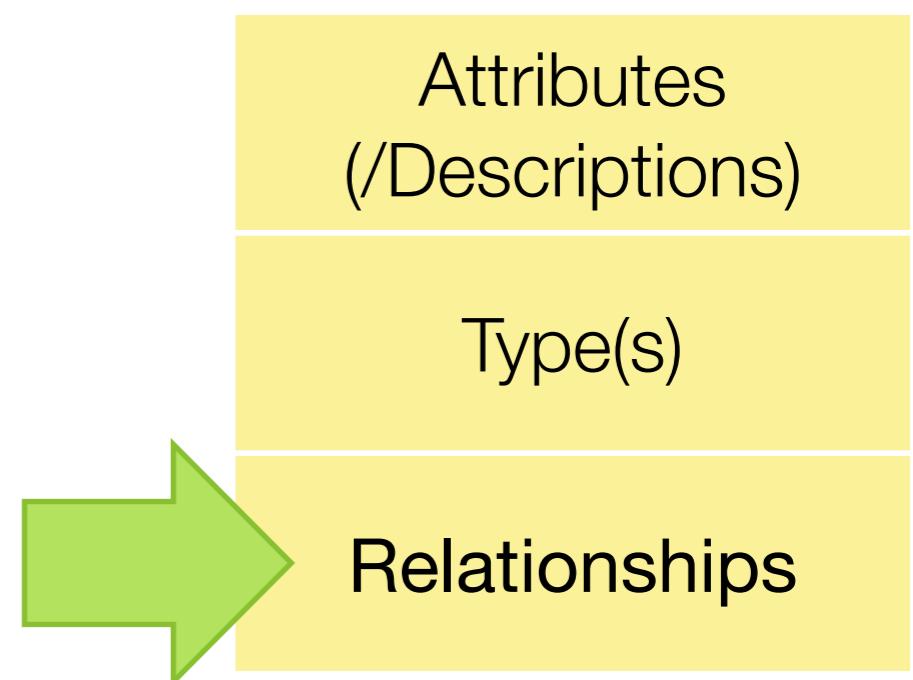
# Identifying target types for queries

- Types of top ranked entities **[Vallet & Zaragoza 2008]**
- Direct term-based vs. indirect entity-based representations **[Balog & Neumayer 2012]**
- Hierarchical case is difficult... **[Sawant & Chakrabarti 2013]**

# Expanding target types

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

# Entity relationships



# Related entities

Google  Search Krisztian Balog Profile

Web Images Maps Shopping News More Search tools

About 8,700,000 results (0.41 seconds)

**Kimi Raikkonen - Lotus**  
Flag 3rd in Formula One World Championship - 116 points - 1 wins - 9 starts

**Recent races**

		Place	Points	Time
Jun 30	British Grand Prix	5	10	01:33:10
Jul 7	German Grand Prix	2	18	01:41:15
Jul 28	Hungarian Grand Prix			today 8:00 AM (EST)

**News for kimi raikkonen**

 [Kimi Raikkonen leaves future to fate and gut instinct](#)  
[The Guardian](#) - 1 day ago  
Kimi Raikkonen, favourite to replace Mark Webber at Red Bull, has said he will decide his team for next season on what feels right for him.

[DECISION TIME ... Raikkonen insists he has no idea what will happen](#)  
[The Sun](#) - 1 day ago  
[Kimi Räikkönen's manager says driver still in running for Red Bull, Lotus F1 rid...](#)  
[AutoWeek](#) - 15 hours ago

**Kimi Räikkönen - Wikipedia, the free encyclopedia**  
[https://en.wikipedia.org/wiki/Kimi\\_Räikkönen](https://en.wikipedia.org/wiki/Kimi_Räikkönen)  
Kimi-Matias Räikkönen (Finnish pronunciation: [ˈkimi ˈmotiəs ˈræikːonen]; born 17 October 1979) is a Finnish racing driver. After nine seasons racing in ...  
[Jenni Dahlman - List of largest sports contracts - List of Finns - Flying Finn](#)

**KIMI RÄIKKÖNEN Official Web Site | Lotus Formula 1 Driver**  
[www.kimiraikkonen.com/](http://www.kimiraikkonen.com/)  
Official site features news, biography, pictures, videos, fan club and chat.

**Kimi Räikkönen - Formula 1® - The Official F1® Website**  
[www.formula1.com/teams\\_and\\_drivers/drivers/12/](http://www.formula1.com/teams_and_drivers/drivers/12/)  
Kimi Raikkonen (FIN) Lotus F1. Formula One World Championship, Rd7, Canadian. 2013. Emerges as an early championship contender after brilliantly winning ...

**Kimi Räikkönen Space**  
[kimi.raikkonen.com/](http://kimi.raikkonen.com/)



More images

## Kimi Räikkönen

Race car driver

Kimi-Matias Räikkönen is a Finnish racing driver. After nine seasons racing in Formula One, in which he won the 2007 Formula One World Drivers' Championship, he competed in the World Rally Championship in 2010 and 2011. [Wikipedia](#)

Born: October 17, 1979 (age 33), [Espoo, Finland](#)  
Height: 5' 9" (1.75 m)  
Full name: Kimi-Matias Räikkönen  
Spouse: [Jenni Dahlman](#) (m. 2004–2013)  
Parents: [Matti Räikkönen](#)  
Siblings: [Rami Räikkönen](#)

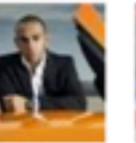
People also search for



Fernando Alonso



Sebastian Vettel



Lewis Hamilton



Felipe Massa



Mark Webber



tom cruise a|



tom cruise and katie holmes  
tom cruise age  
tom cruise and cameron diaz  
tom cruise and nicole kidman



tom cruise wives



Krisztian Balog



+ Share



Web Images Maps Shopping More Search tools

About 2,650,000 results (0.23 seconds)

### Tom Cruise Spouse



Katie Holmes  
(m. 2006–2012)



Nicole Kidman  
(m. 1990–2001)



Mimi Rogers  
(m. 1987–1990)

Feedback / More info

[Each of Tom Cruise's wives](#) has been 11 years younger than the ...  
[www.omg-facts.com](#) > Celebrity Facts

Mimi Rogers was born in 1956, Nicole Kidman was born in 1967, and Katie Holmes was born in 1978. Tom himself was born in 1962, meaning that he was six ...

## Tom Cruise

Actor

Follow



Thomas Cruise Mapother IV, widely known as Tom Cruise, is an American film actor and producer. He has been nominated for three Academy Awards and has won three Golden Globe Awards. He started his career at age 19 in the 1981 film *Taps*.  
[Wikipedia](#)

**Born:** July 3, 1962 (age 51), Syracuse, New York, United States

**Height:** 5' 7" (1.70 m)

**Upcoming movies:** [All You Need Is Kill](#), [Mission: Impossible 5](#)

**Spouse:** [Katie Holmes](#) (m. 2006–2012), [Nicole Kidman](#) (m. 1990–2001), [Mimi Rogers](#) (m. 1987–1990)

**Children:** [Suri Cruise](#), [Isabella Jane Cruise](#), [Connor Cruise](#)

# TREC Entity track

- Related Entity Finding task
- Given
  - Input entity (defined by name and homepage)
  - Type of the target entity (PER/ORG/LOC)
  - Narrative (describing the nature of the relation in free text)
- Return (homepages of) related entities

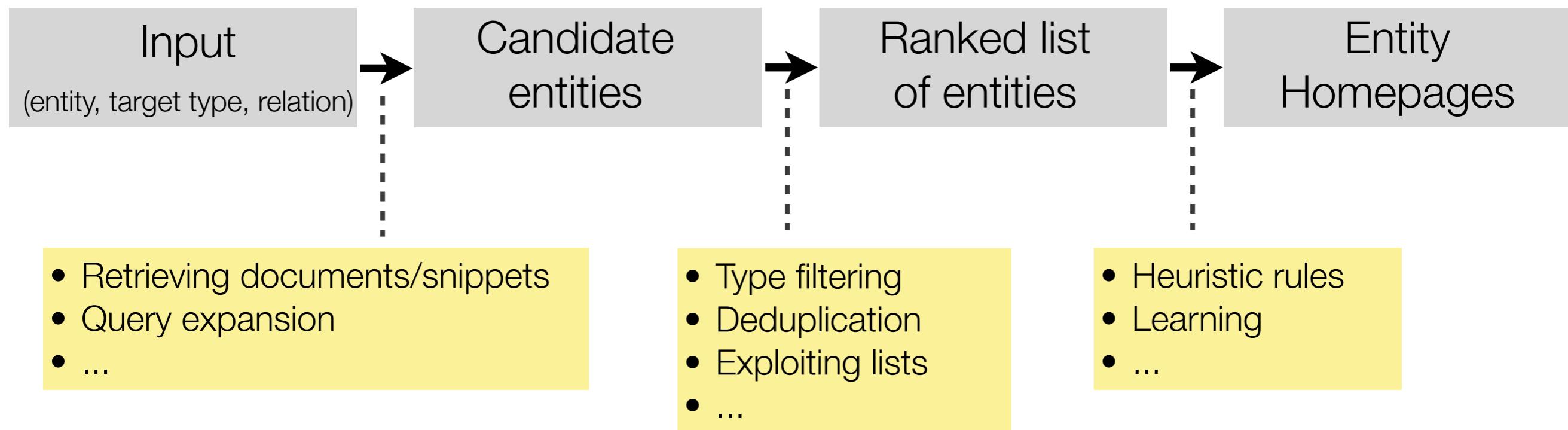
# Example information needs

🔍 airlines that currently use Boeing 747 planes  
ORG Boeing 747

🔍 Members of The Beaux Arts Trio  
PER The Beaux Arts Trio

🔍 What countries does Eurail operate in?  
LOC Eurail

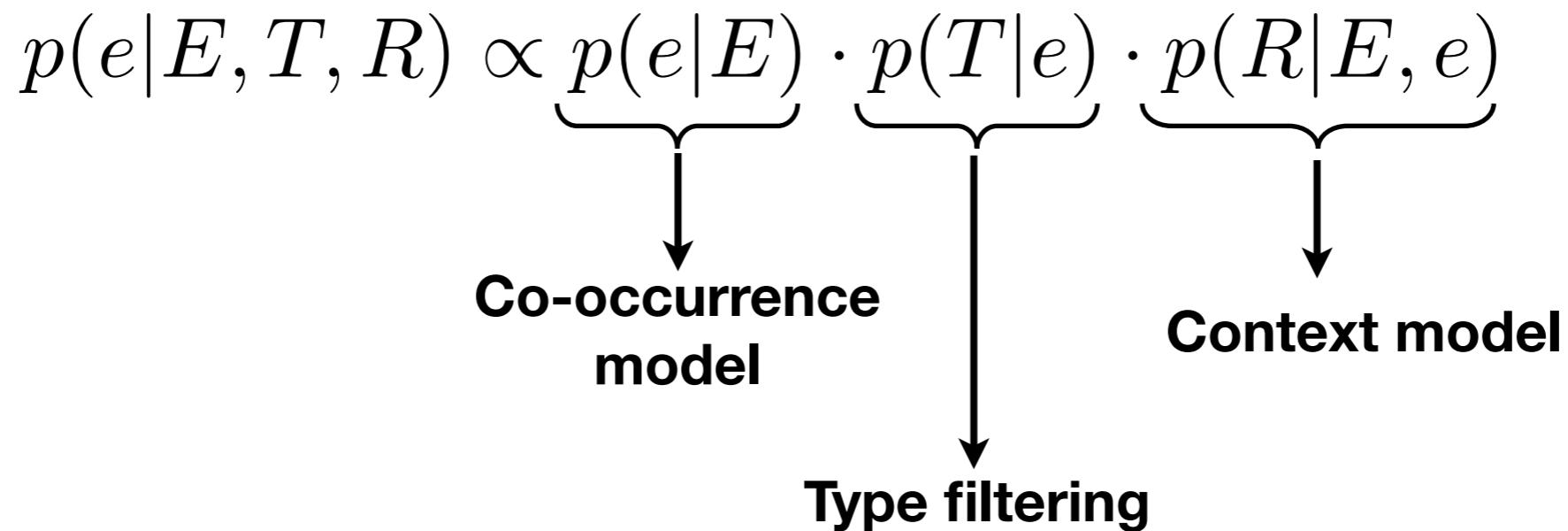
# A typical pipeline



# Modeling related entity finding

[Bron et al. 2010]

- Three-component model



**Anything else?**

# 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]**

# Query understanding

- Structuring and segmentation [**Bendersky et al. 2010, Bendersky et al. 2011**]
- Aiding the user with context-sensitive suggestions [**Bast et al. 2012**]
- Query interpretation with the help of knowledge bases [**Pound et al. 2012**]

# Specialized interfaces

## [Bast et al. 2012]

type here to extend your query ...

► Words

▼ Classes:

Garden plant	(24)
House plant	(17)
Crop	(16)
1 - 3 of 28	

▼ Instances:

Broccoli	(58)
Cabbage	(34)
Lettuce	(23)
1 - 3 of 421	

▼ Relations:

occurs-with	<Anything>	
cultivated-in	<Location>	(67)
belongs-to	<Plant family>	(58)
1 - 3 of 7		

Your Query:

```
Plant ✖
  |
  +-- occurs-with ✖
  |    +-- edible leaves ✖
  |
  +-- native-to ✖
      +-- Europe ✖
```

Hits: 1 - 2 of 421

**Broccoli**

Ontology: Broccoli  
Broccoli: is a **plant**; native to **Europe**.  
Document: Edible plant stems  
The **edible** portions of **Broccoli** are the stem tissue, the flower buds, as well as the **leaves**.



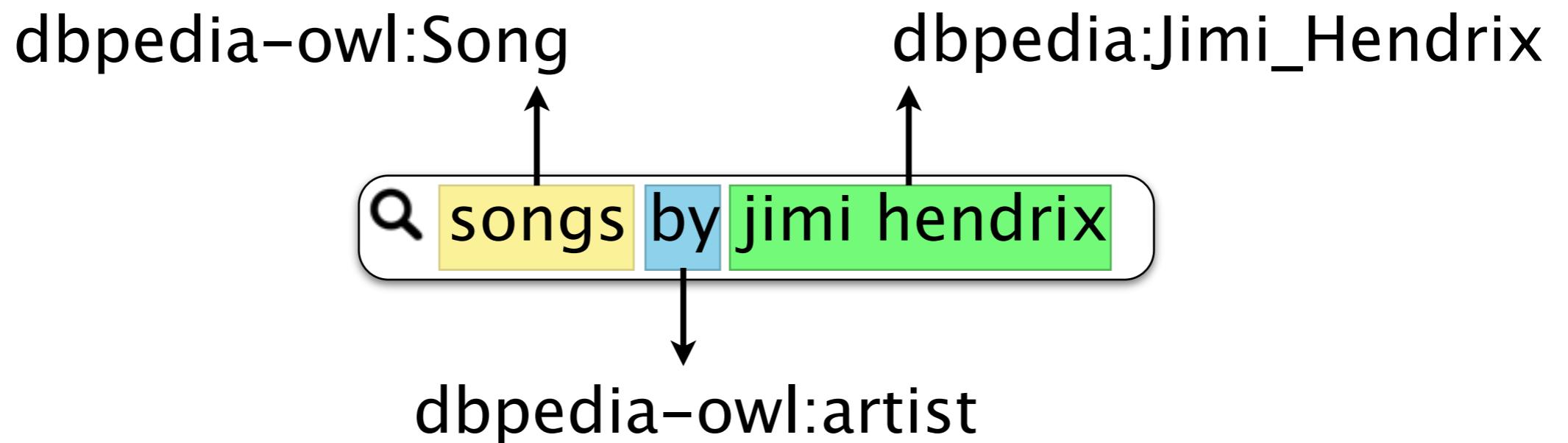
**Cabbage**

Ontology: Cabbage  
Cabbage: is a **plant**; native to **Europe**.  
Document: Cabbage  
The only part of the **plant** that is normally **eaten** is the **leafy** head.



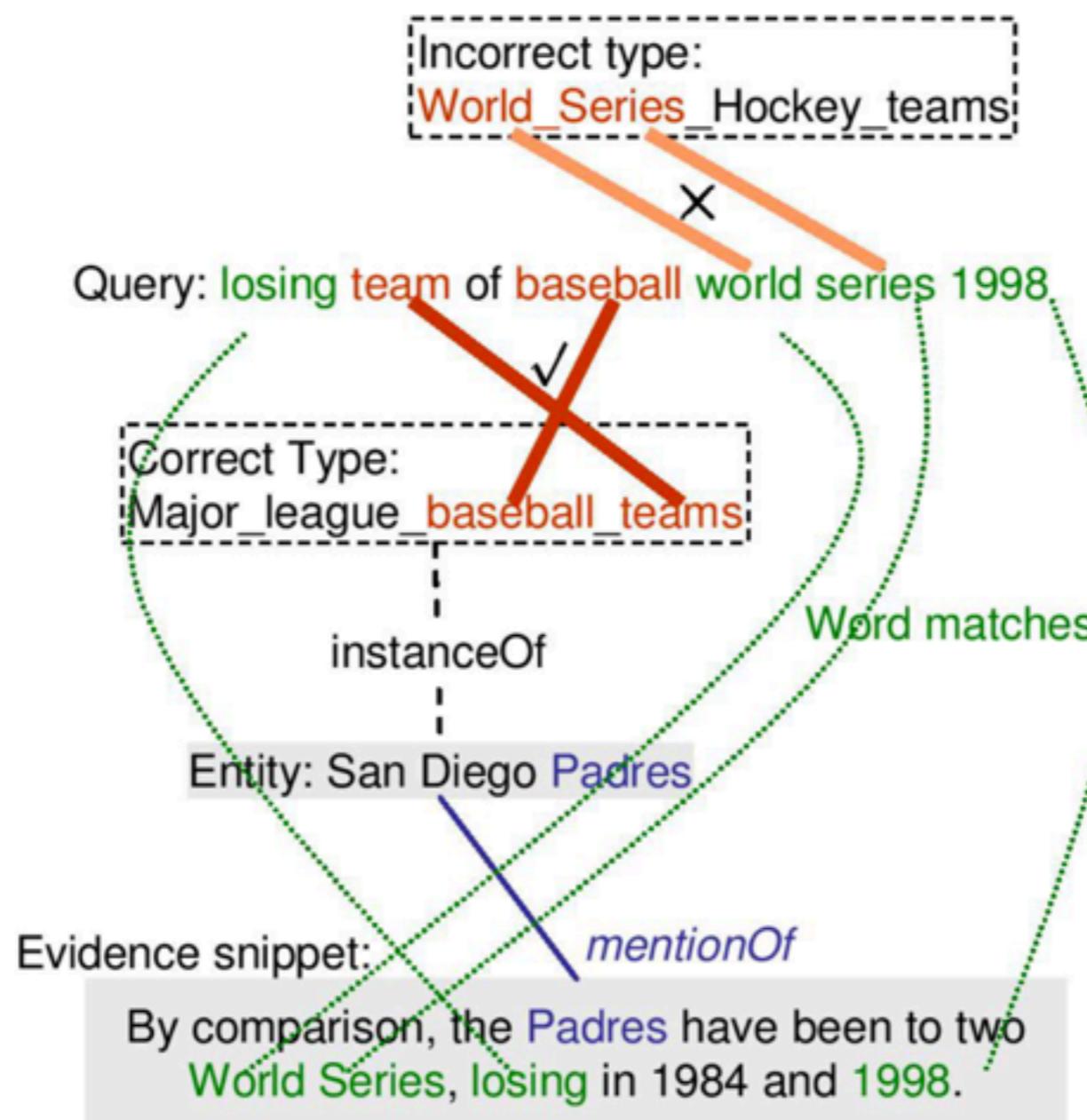
Figure taken from Bast et al. **Broccoli: Semantic Full-Text Search at your Fingertips.**  
<http://arxiv.org/abs/1207.2615/>

# Semantic query understanding



# Joint query interpretation and response ranking

[Sawant & Chakrabarti 2013]



# **Tools & services**



# **Public Toolkits and Web Services for Entity Retrieval**

- EARS
- Sindice & SIREn
- Sig.ma

# EARS



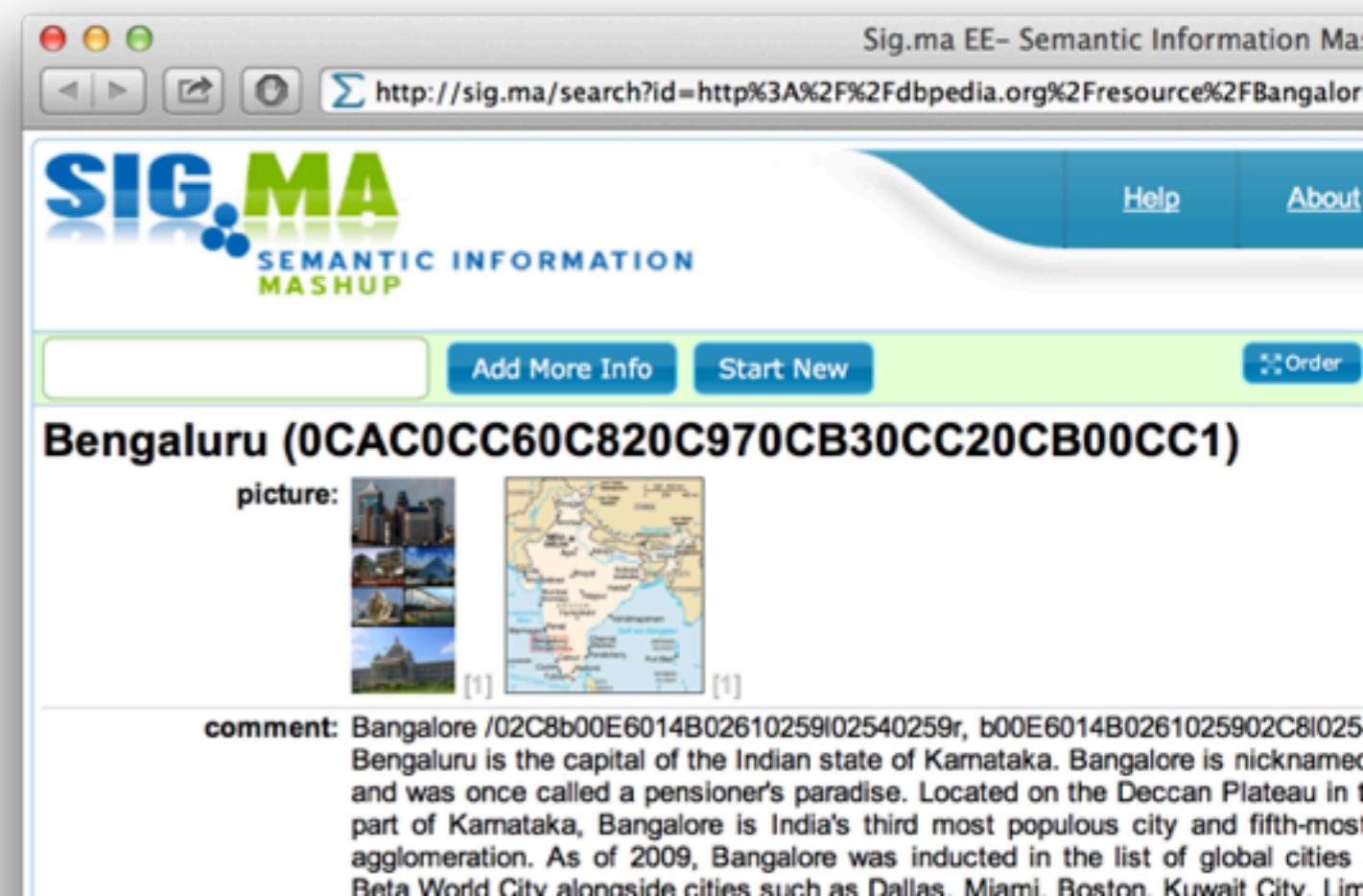
- Entity and Association Retrieval System
  - open source, built on top of Lemur in C++
    - not actively maintained anymore (but still works)
- Entity-topic association finding models
  - suited for other tasks, e.g. blog distillation
  - focuses on two entity-related tasks:
    - finding entities:
      - "Which entities are associated with topic X?"
    - profiling entities:
      - "What topics is an entity associated with?"
- See <https://code.google.com/p/ears/>

# Sindice/SIREn

- Handling of semi-structured data
  - efficient, large scale
  - typically based on DBMS backends
  - uses Lucene for semi-structured search
- Open source
- Online demo, local install
- See <http://siren.sindice.com/>

# Sig.ma

- Search, aggregate, and visualize LOD data
- Powered by Sindice
- See <http://sig.ma/>



# **Test collections**

# 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
- DBpedia entity search **[Balog & Neumayer 2013]**
  - synthesized queries and assessments, distilled from previous campaigns
  - from short keyword queries to natural language questions
  - 485 queries in total; mapped to DBpedia

# **Open challenges**

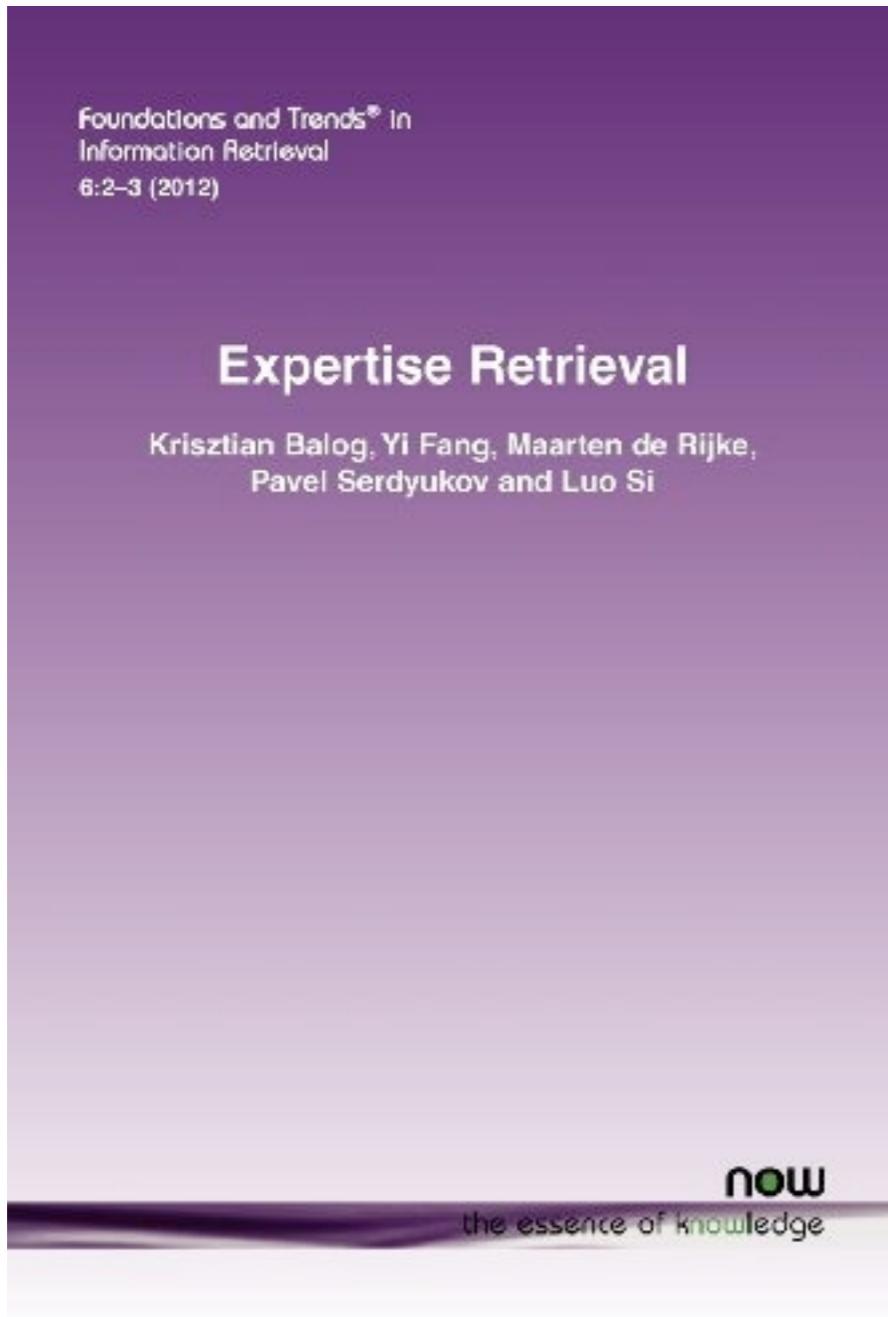
# **Open challenges**

- Combining text and structure
  - Knowledge bases and unstructured Web documents
- Query understanding and modeling
- UI/UX/Result presentation
  - How to interact with entities
- Hyperlocal
  - Siri/Google Now/...
  - Recommendations

# **Open challenges (2)**

- There is more to types than currently exploited
  - Multiple category systems, hierarchy, ...
- Entity retrieval is typically part of some more complex task
  - Buying a product, planning a vacation, etc.

# Follow-up reading



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

# References – Entity retrieval

The screenshot shows a Mendeley group page titled "Entity Linking and Retrieval – Tutorial at WWW 2013 and SIGIR 2013". The page displays three research papers:

- Analysis and Enhancement of Wikification for Microblogs with Context Expansion.** By Taylor Cassidy, Heng Ji, Lev-Arie Ratinov, Arkatz Zubaga, Hongzhao Huang. Published in COLING 2012 (2012).

Disambiguation to Wikipedia (D2W) is the task of linking mentions of concepts in text to their corresponding Wikipedia entries. Most previous work has focused on linking terms in formal texts (e.g. newswire) to Wikipedia. Linking terms in short...

Added 1 minute ago | 1 reader
- Microblog-genre noise and impact on semantic annotation accuracy** by Leon Derczynski, Diana Maynard, Niraj Aswani, Kalina Bontcheva in HT 2013 (2013).

Using semantic technologies for mining and intelligent information access to microblogs is a challenging, emerging research area. Unlike carefully authored news text and other longer content, tweets pose a number of new challenges, due to their...

Added 11 minutes ago
- Entity Disambiguation with Freebase** by Zicheng Zheng, Xiancse Si, Fangtao Li, Edward Y. Chang, Xiaoyan Zhu in WIAT 2013 (2013).

Using semantic technologies for mining and intelligent information access to microblogs is a challenging, emerging research area. Unlike carefully authored news text and other longer content, tweets pose a number of new challenges, due to their...

Added 11 minutes ago

The right sidebar shows "Top tags in this group" including entity linking, Wikipedia, TAC, commonness, SVM, graph, relatedness, naive bayes, pagerank, keyphraseness, Twitter, centrality, meta evaluation, NER, word sense disambiguation, random forests, Freebase, tagme, local, web.

<http://www.mendeley.com/groups/3339761/entity-linking-and-retrieval-tutorial-at-www-2013-and-sigir-2013/papers/added/0/tag/entity+retrieval/>