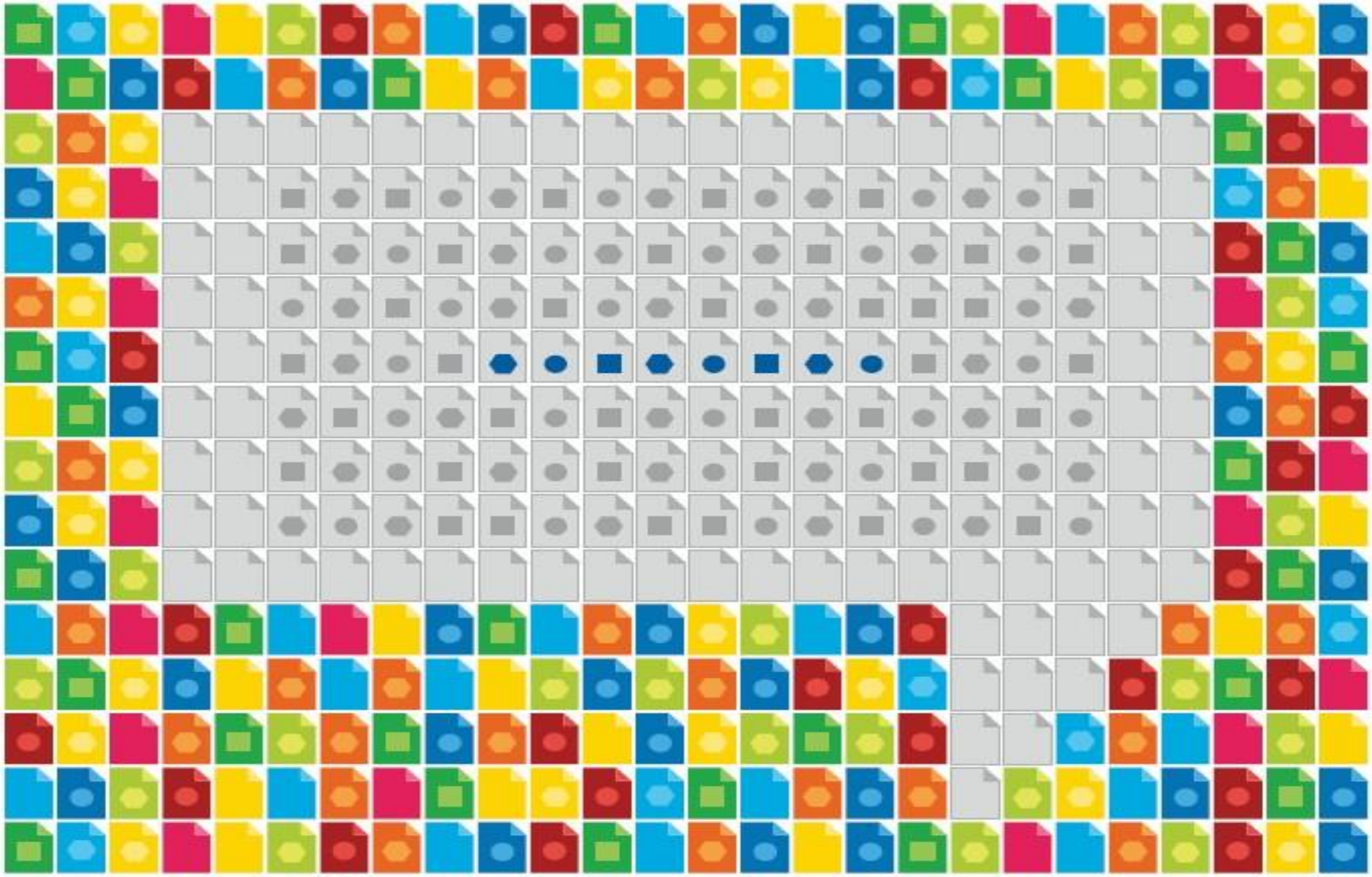
Knowledge Harvesting from Web Sources

Part 2: Knowledge Base Applications: Search, Ranking, Disambiguation

Gerhard Weikum
Max Planck Institute for Informatics
http://www.mpi-inf.mpg.de/~weikum/

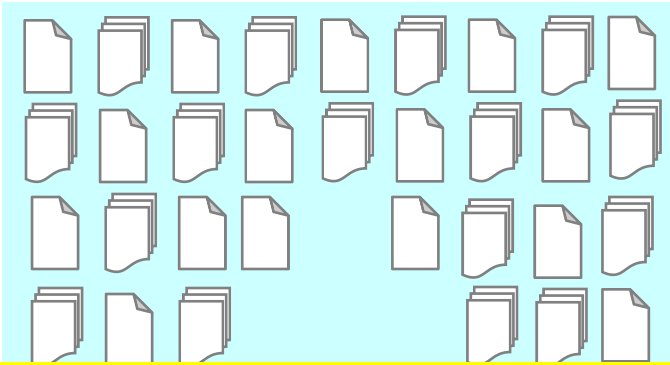
The Web Speaks to Us

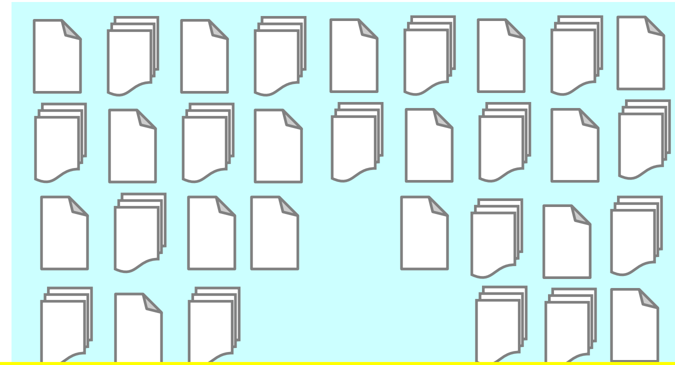


Source:
DB & IR methods for knowledge discovery. Communications of the ACM 52(4), 2009

- **Web 2011 contains more DB-style data than ever**
- **getting better at making structured content explicit: entities, classes (types), relationships**
- **but no (hope for) schema here!**

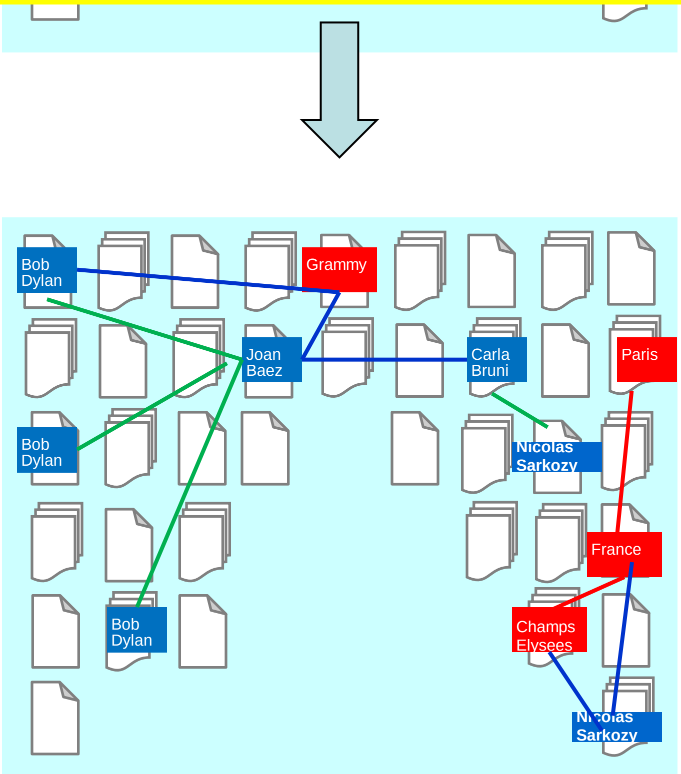
Structure Now!

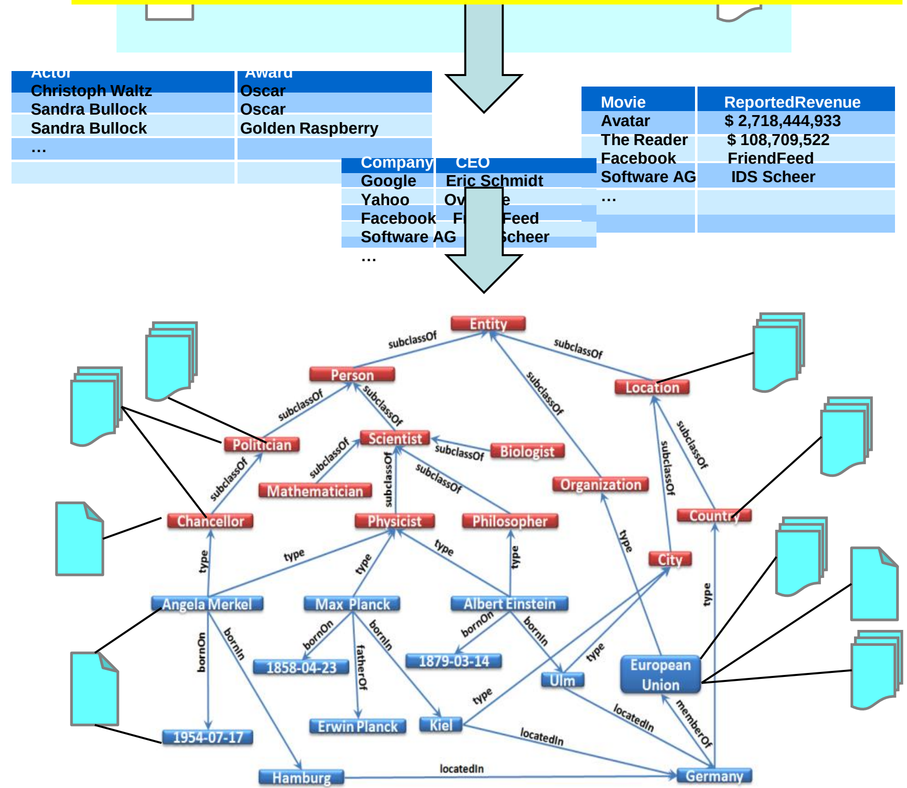




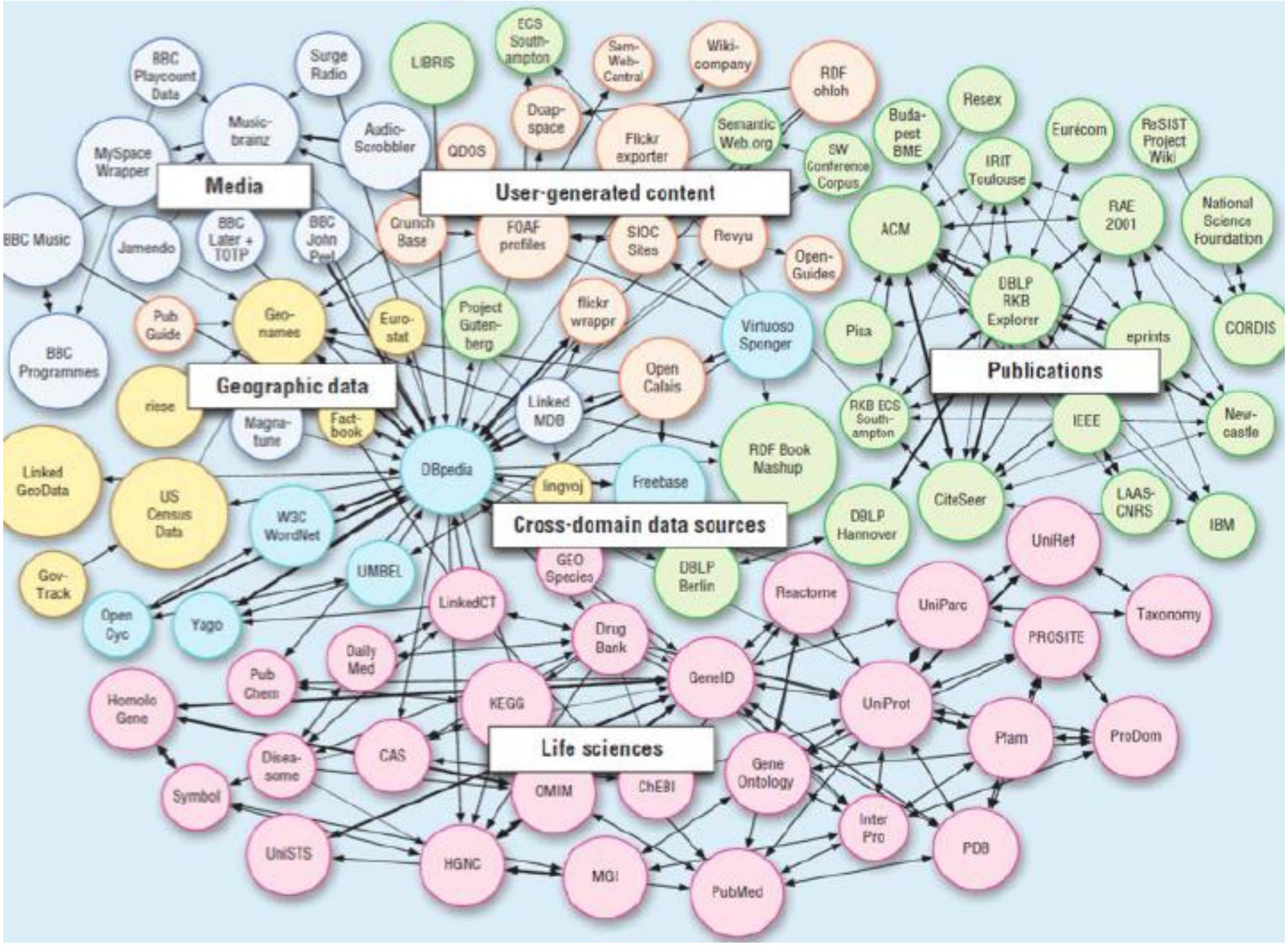
IE-enriched Web pages with embedded entities and facts

Knowledge bases with facts from Web and IE witnesses



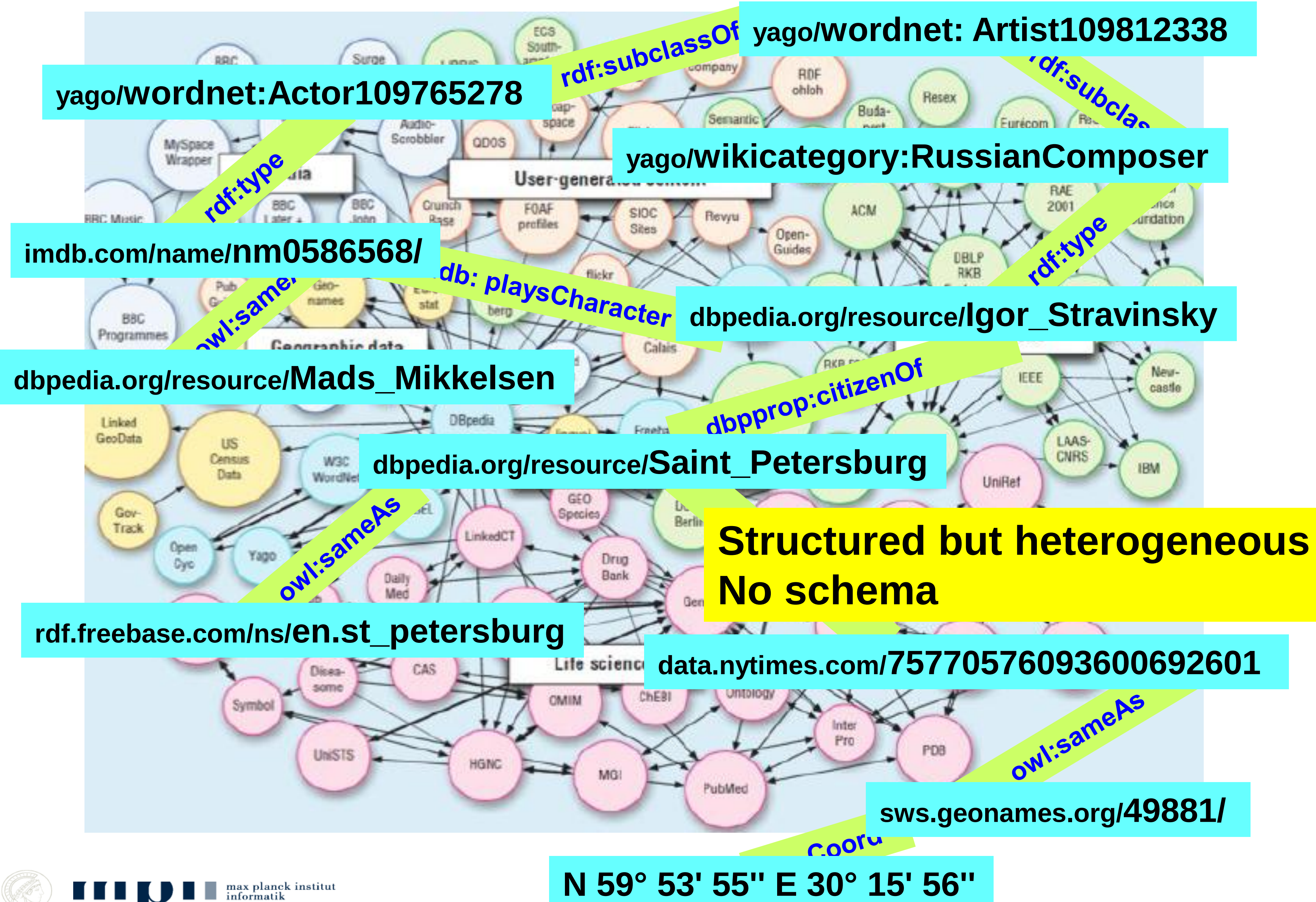


Distributed Structure: Linking Open Data



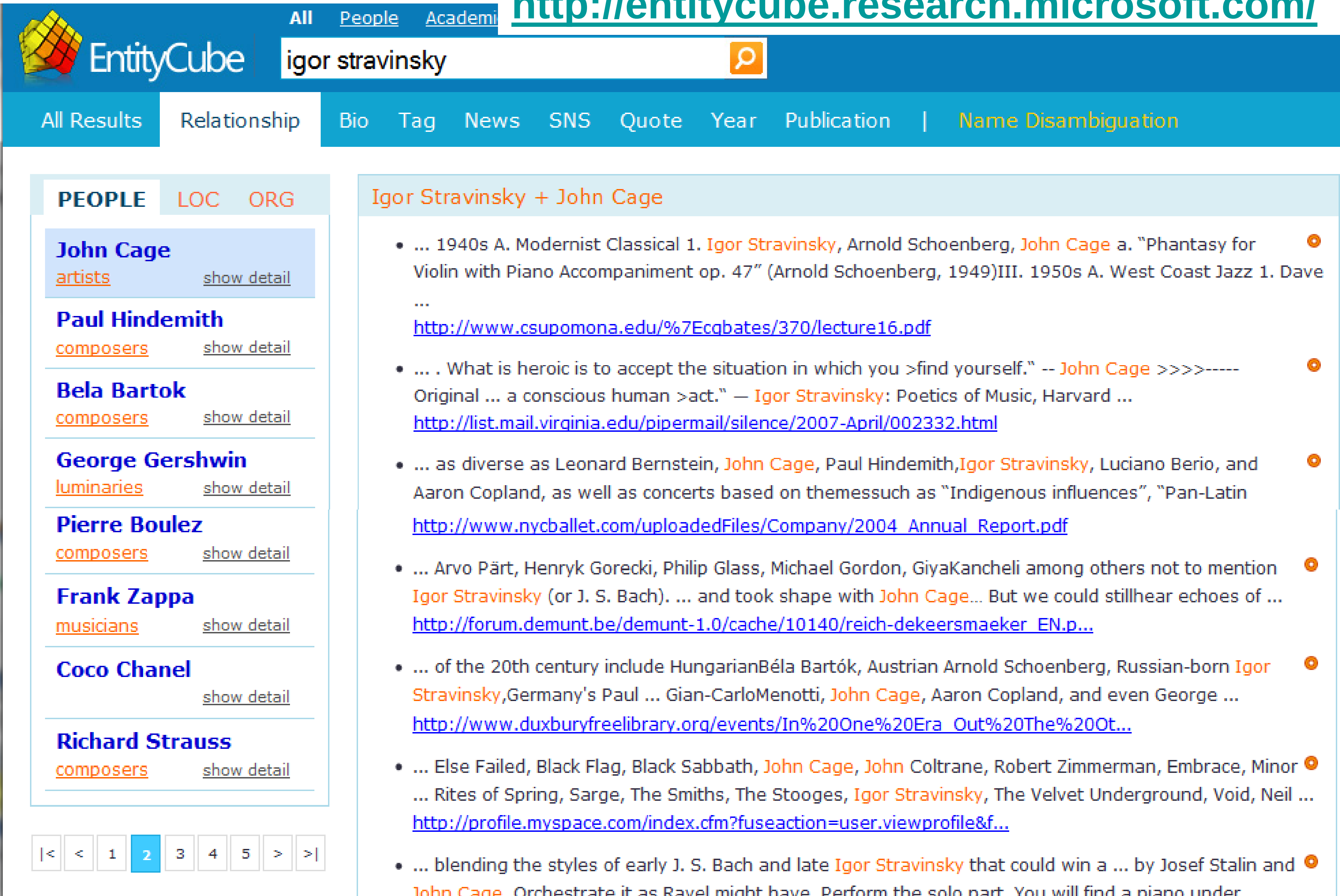
Source: Christian Bizer, Tom Heath, Tim Berners-Lee, Michael Hausenblas, WWW 2010 Workshop on Linked Data on the Web

Distributed Structure: Linking Open Data



Entity Search

http://entitycube.research.microsoft.com/



Semantic Search with Entities, Classes, Relationships

properties of entity

- US president when Barack Obama was born?
  Nobel laureate who outlived two world wars and all his children?

instances of classes

- Politicians who are also scientists?
  European composers who won the Oscar?

relationships

- FIFA 2010 finalists who played in a Champions League final?
  German soccer clubs that won against Real Madrid?

multiple entities

- Commonalities & relationships among:
  Max Planck, Angela Merkel, Jim Gray, Dalai Lama?

applications

- Enzymes that inhibit HIV?
  Antidepressants that interfere with blood-pressure drugs?
  German philosophers influenced by William of Ockham?

Outline

- ✓ Motivation
- ★ **Searching for Entities & Relations**
- ★ Informative Ranking
- ★ Entity-Name Disambiguation
- ★ Wrap-up

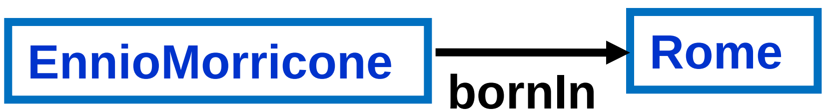
RDF: Structure, Diversity, No Schema

SPO triples (statements, facts):

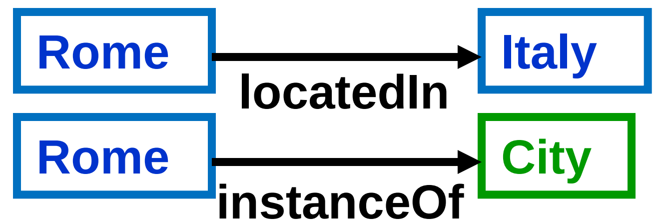
(EnnioMorricone, bornIn, Rome)
(Rome, locatedIn, Italy)
(JavierNavarrete, birthPlace, Teruel)
(Teruel, locatedIn, Spain)
(EnnioMorricone, composed, l'Arena)
(JavierNavarrete, composerOf, aTale)

(uri1, hasName, EnnioMorricone)
(uri1, bornIn, uri2)
(uri2, hasName, Rome)
(uri2, locatedIn, uri3)
…

bornIn (EnnioMorricone, Rome)



locatedIn(Rome, Italy)



- **SPO triples**: Subject – Property/Predicate – Object/Value)
- **pay-as-you-go**: schema-agnostic or schema later
- RDF triples form **fine-grained ER graph**
- popular for **Linked Data**, comp. biology (UniProt, KEGG, etc.)
- open-source engines: Jena, Sesame, RDF-3X, etc.

Facts about Facts

facts:

1: (EnnioMorricone, composed, l'Arena)

2: (JavierNavarrete, composerOf, aTale)

3: (Berlin, capitalOf, Germany)

4: (Madonna, marriedTo, GuyRitchie)

5: (NicolasSarkozy, marriedTo, CarlaBruni)

temporal facts:

6: (1, inYear, 1968)

7: (2, inYear, 2006)

8: (3, validFrom, 1990)

9: (4, validFrom, 22-Dec-2000)

10: (4, validUntil, Nov-2008)

11: (5, validFrom, 2-Feb-2008)

provenance:

12: (1, witness, http://www.last.fm/music/Ennio+Morricone/)

13: (1, confidence, 0.9)

14: (4, witness, http://en.wikipedia.org/wiki/Guy_Ritchie)

15: (4, witness, http://en.wikipedia.org/wiki/Madonna_(entertainer))

16: (10, witness, http://www.intouchweekly.com/2007/12/post_1.php)

17: (10, confidence, 0.1)

- **temporal annotations**, witnesses/sources, confidence, etc. can refer to **reified facts** via fact identifiers

  (approx. equiv. to RDF quadruples: Col × Sub × Prop × Obj)

SPARQL Query Language

SPJ combinations of **triple patterns**
(triples with S,P,O replaced by variable(s))

Select ?p, ?c Where {
?p instanceOf Composer .
?p bornIn ?t . ?t inCountry ?c . ?c locatedIn Europe .
?p hasWon ?a .?a Name AcademyAward . }

Semantics:
return all bindings to variables that match all triple patterns
(subgraphs in RDF graph that are isomorphic to query graph)

+ filter predicates, duplicate handling, RDFS types, etc.

Select Distinct ?c Where {
?p instanceOf Composer .
?p bornIn ?t . ?t inCountry ?c . ?c locatedIn Europe .
?p hasWon ?a .?a Name ?n .
?p bornOn ?b . Filter (?b > 1945) . Filter(regex(?n, "Academy") . }

Querying the Structured Web

Structure but no schema: SPARQL well suited

flexible subgraph matching

wildcards for properties (relaxed joins):

Select ?p, ?c Where {
?p instanceOf Composer .
?p ?r1 ?t . ?t ?r2 ?c . ?c isa Country . ?c locatedIn Europe . }

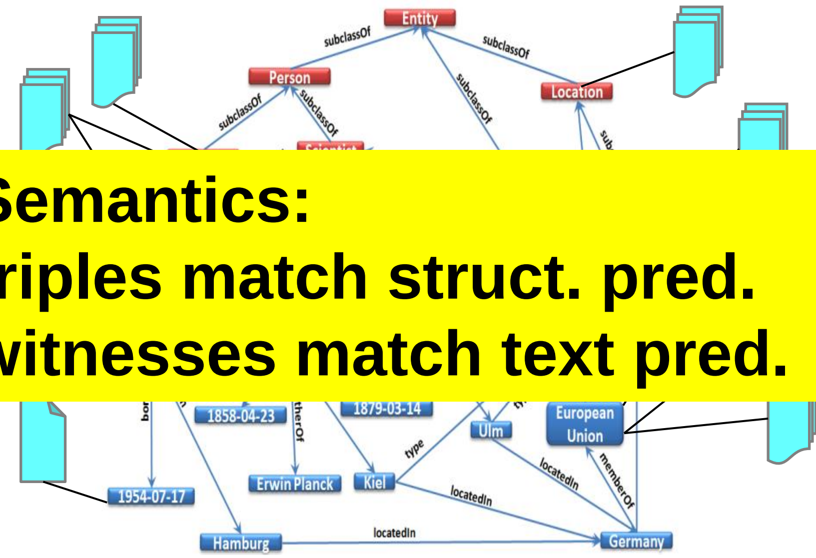
Extension: transitive paths [K. Anyanwu et al.: WWW'07]

Select ?p, ?c Where {
?p instanceOf Composer .
?p ??r ?c . ?c isa Country . ?c locatedIn Europe .
PathFilter(cost(??r) < 5) .
PathFilter (containsAny(??r,?t) . ?t isa City . }

Extension: regular expressions [G. Kasneci et al.: ICDE'08]

Select ?p, ?c Where {
?p instanceOf Composer .
?p (bornIn | livesIn | citizenOf) locatedIn* Europe . }

Querying Facts & Text



Problem: not everything is triplified

- **Consider witnesses/sources (provenance meta-facts)**
- **Allow text predicates with each triple pattern (à la XQ-FT)**

Semantics:
triples match struct. pred.
witnesses match text pred.

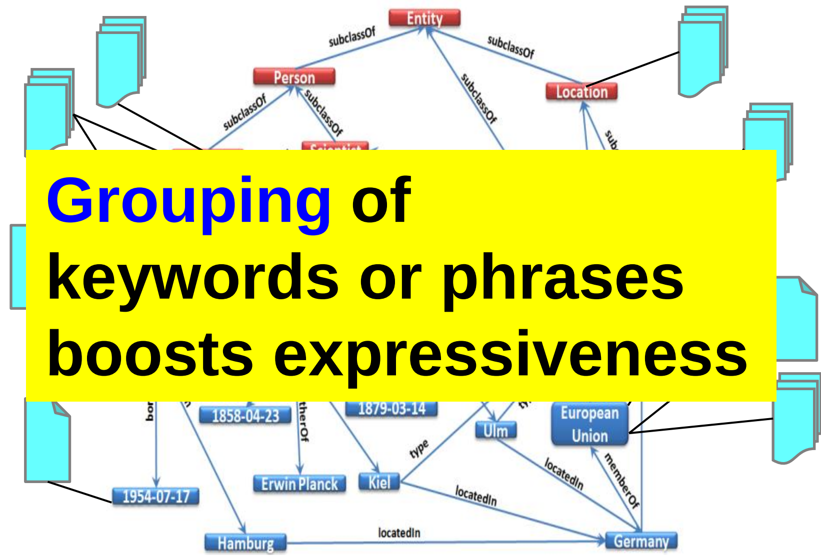
European composers who have won the Oscar, whose music appeared in dramatic western scenes, and who also wrote classical pieces ?

```
Select ?p Where {
?p instanceOf Composer .
?p bornIn ?t . ?t inCountry ?c . ?c locatedIn Europe .
?p hasWon ?a .?a Name AcademyAward .
?p contributedTo ?movie [western, gunfight, duel, sunset] .
?p composed ?music [classical, orchestra, cantata, opera] . }
```

Querying Facts & Text

Problem: not everything is triplified

- Consider **witnesses/sources** (provenance meta-facts)
- Allow **text predicates** with each triple pattern (à la XQ-FT)



Grouping of keywords or phrases boosts expressiveness

French politicians married to Italian singers?

Select ?p1, ?p2 Where {
?p1 instanceOf politician [France] .
?p2 instanceOf singer [Italy] .
?p1 marriedTo ?p2 . }

Select ?p1, ?p2 Where {
?p1 instanceOf ?c1 [France, politics] .
?p2 instanceOf ?c2 [Italy, singer] .
?p1 marriedTo ?p2 . }

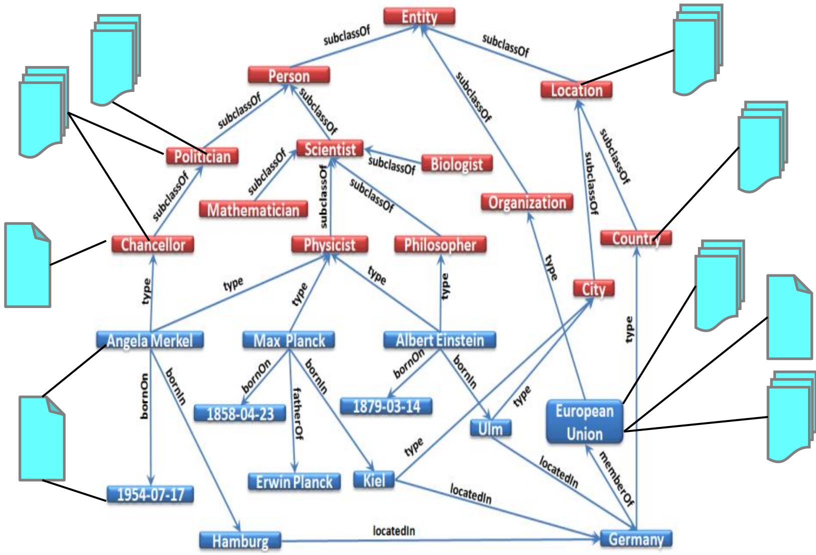
CS researchers whose advisors worked on the Manhattan project?

Select ?r, ?a Where {
?r instanceOf researcher [computer science] .
?a workedOn ?x [Manhattan project] .
?r hasAdvisor ?a . }

?r ?p1 ?c1 [computer] .
?a ?p2 ?x [Manhattan project] .
?r ?p3 ?a . }

Relatedness Queries

Schema-agnostic keyword search (on RDF, ER graph, relational DB) becomes a special case



Relationship between Angela Merkel, Jim Gray, Dalai Lama?

Select ??p1, ??p2, ??p3 Where {
AngelaMerkel ??p1 MaxPlanck.
MaxPlanck ??p2 DalaiLama .
DalaiLama ??p3 AngelaMerkel . }

Select ??p1, ??p2, ??p3 Where {
?e1 ?r1 ?c1 [“Angela Merkel“] .
?e2 ?r2 ?c2 [“Max Planck“] .
?e3 ?r3 ?c3 [“Dalai Lama“] .
?e1 ??p1 ?e2 .
?e2 ??p2 ?e3 .
?e3 ??p3 ?e1 . }

Querying Temporal Facts

[Y. Wang et al.: EDBT'10]
[O. Udrea et al.: TOCL'10

Problem: not all facts hold forever (e.g. CEOs, spouses, …)

- **Consider temporal scopes of reified facts**
- **Extend Sparql with temporal predicates**

1: (BayernMunich, hasWon, ChampionsLeague)
2: (BorussiaDortmund, hasWon, ChampionsLeague)
3: (1, validOn, 23May2001) 4: (1, validOn, 15May1974)
5: (2, validOn, 28May1997)
6: (OttmarHitzfeld, manages, BayernMunich)
7: (6, validSince, 1Jul1998) 8:(6, validUntil, 30Jun2004)

When did a German soccer club win the Champions League?

Select ?c, ?t Where { ?c isa soccerClub . ?c inCountry Germany .
?id1: ?c hasWon ChampionsLeague . ?id1 validOn ?t . }

Managers of German clubs who won the Champions League?

Select ?m Where { isa soccerClub . ?c inCountry Germany .
?id1: ?c hasWon ChampionsLeague . ?id1 validOn ?t .
?id2: ?m manages ?c . ?id2 validSince ?s . ?id2 validUntil ?u .
[?s,?u] overlaps [?t,?t] . }

Querying with Vague Temporal Scope

98 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 20

Problem: user‘s temporal interest is often imprecise

- Consider temporal phrases as text conditions
- Allow approximate matching and rank results wisely

German Champion League winners in the nineties?

Select ?c Where {
?c isa soccerClub . ?c inCountry Germany .
?c hasWon ChampionsLeague [nineties] . }

Soccer final winners in summer 2001?

Select ?c Where {
?c isa soccerClub . ?id: ?c matchAgainst ?o [final] .
?id winner ?c [“summer 2001“] . }

Take-Home Message (Querying)

- Don‘t re-invent the wheel:

  SPARQL is there already, entity search is special case

- Extensions should be conceptually simple

  meta-fact & text predicates naturally embedded

- Ease of use (progammability) is crucial:

  doing well for API, UI is harder

Outline

✓ **Motivation**

✓ **Searching for Entities & Relations**

★ **Informative Ranking**

★ **Entity-Name Disambiguation**

★ **Wrap-up**

Ranking Criteria

Confidence:
Prefer results that are likely correct
- ➢ accuracy of info extraction
- ➢ trust in sources (authenticity, authority)

bornIn (Jim Gray, San Francisco) from „Jim Gray was born in San Francisco“ (en.wikipedia.org)

livesIn (Michael Jackson, Tibet) from „Fans believe Jacko hides in Tibet“ (www.michaeljacksonsightings.com)

Informativeness:
Prefer results with salient facts
Statistical estimation from:
- ➢ frequency in answer
- ➢ frequency on Web
- ➢ frequency in query log

q: Einstein isa ?
Einstein isa scientist
Einstein isa vegetarian

q: ?x isa vegetarian
Einstein isa vegetarian
Whocares isa vegetarian

Diversity:
Prefer variety of facts

E won … E discovered … E played …
E won … E won … E won … E won …

Conciseness:
Prefer results that are tightly connected
- ➢ size of answer graph
- ➢ cost of Steiner tree

Einstein won NobelPrize
Bohr won NobelPrize

Einstein isa vegetarian
Cruise isa vegetarian
Cruise born 1962 Bohr died 1962

Ranking Approaches

Confidence:

Prefer results that are likely correct
- accuracy of info extraction
- trust in sources (authenticity, authority)

empirical accuracy of IE
PR/HITS-style estimate of trust
combine into:
max { accuracy (f,s) * trust(s) | s $\in$ witnesses(f) }

Informativeness:

Prefer results with salient facts
Statistical LM with estimations from:
- frequency in answer
- frequency in corpus (e.g. Web)
- frequency in query log

PR/HITS-style entity/fact ranking
[V. Hristidis et al., S.Chakrabarti, …]

or

IR models: tf*idf … [K.Chang et al., …]
Statistical Language Models

Diversity:

Prefer variety of facts

Statistical Language Models

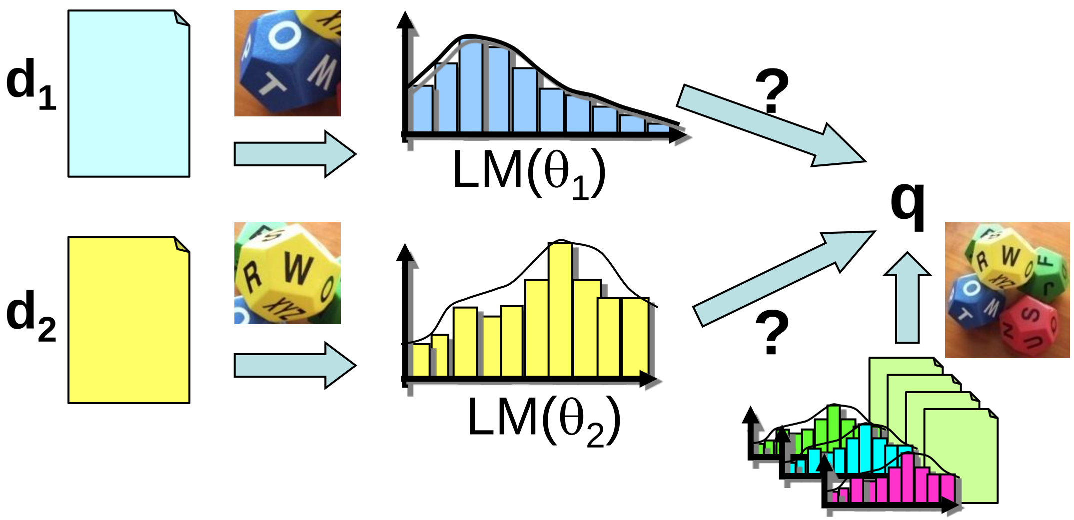
Conciseness:

Prefer results that are tightly connected
- size of answer graph
- cost of Steiner tree

graph algorithms (BANKS, STAR, …)
[J.X. Yu et al., S.Chakrabarti et al., B. Kimelfeld et al., G.Kasneci et al., …]

Statistical Language Models (LM's)

[Maron/Kuhns 1960, Ponte/Croft 1998, Hiemstra 1998, Lafferty/Zhai 2001]

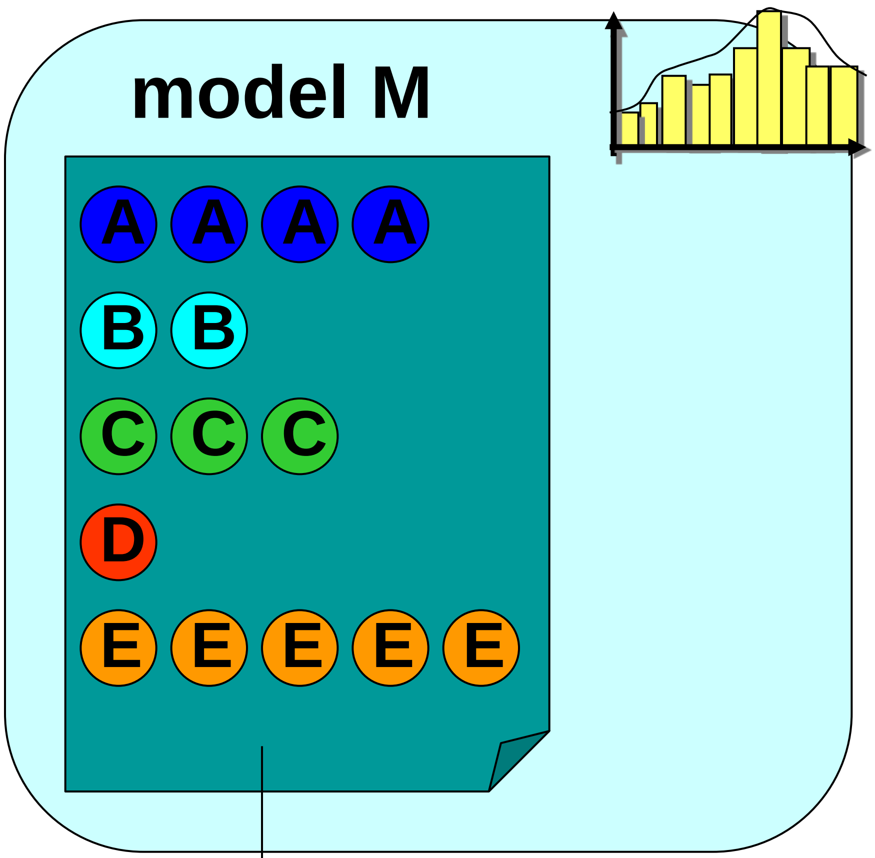


„God does not play dice"
(Albert Einstein)
„IR does" (anonymous)

„God rolls dice
in places where
you can't see them"
(Stephen Hawking)

- each doc d_i has LM: generative prob. distr. with params θ_i
- query q viewed as sample from $LM(\theta_1)$, $LM(\theta_2)$, …
- estimate likelihood $P[\ q \mid LM(\theta_i)\ ]$
  that q is sample of LM of doc d_i (q is „generated by" d_i)
- rank by descending likelihoods (best „explanation" of q)

LM: Doc as Model, Query as Sample

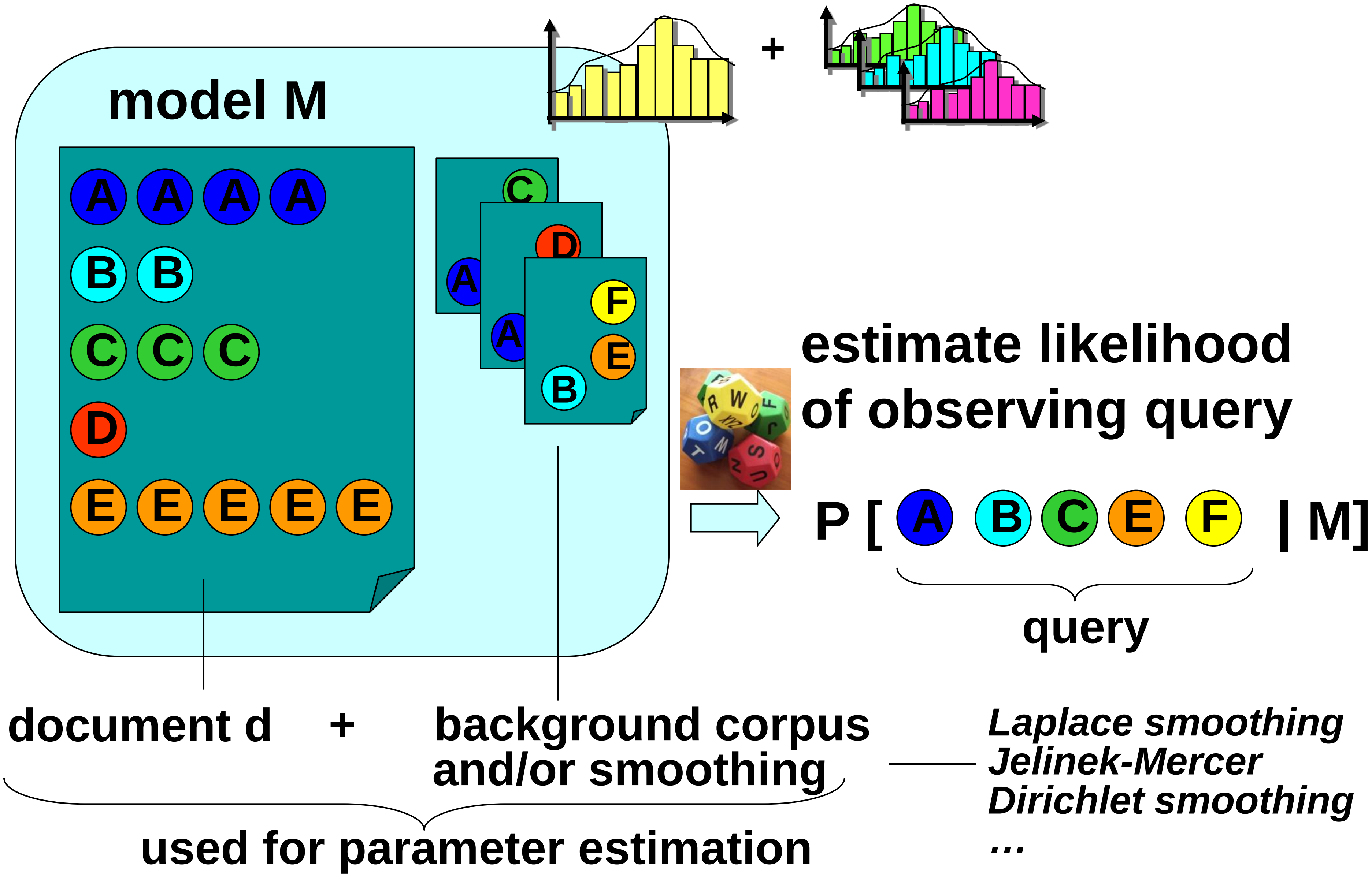


document d: sample of M
used for parameter estimation

estimate likelihood
of observing query

$$P\,[\underbrace{A\;A\;B\;C\;E\;E}_{\text{query}} \mid M]$$

LM: Need for Smoothing



Some LM Basics

$$s(d,q) = P[q \mid d] = \prod_i P[q_i \mid d]$$

independ. assumpt.

$$\sim \sum_i \log \frac{tf(i,d)}{\sum_k tf(k,d)}$$

simple MLE: overfitting

$$s(d,q) = \lambda P[q \mid d] + (1-\lambda) P[q]$$

mixture model for smoothing

$$\sim \sum_i \log \left(\lambda \frac{tf(i,d)}{\sum_k tf(k,d)} + (1-\lambda) \frac{df(i)}{\sum_k df(k)} \right)$$

P[q] est. from log or corpus

$$\sim \sum_i \log \left(1 + \frac{tf(i,d)}{\sum_k tf(k,d)} \cdot \frac{1-\lambda}{\lambda} \frac{\sum_k df(k)}{df(i)} \right)$$

rank by ascending "improbability"

$$\sim KL(q \mid d) = \sum_i P[i \mid q] \log \frac{P[i \mid q]}{P[i \mid d]}$$

KL divergence (Kullback-Leibler div.) aka. relative entropy

- Precompute per-keyword scores
- Store in postings of inverted index
- Score aggregration for (top-k) multi-keyword query

efficient implementation

Entity Search with LM Ranking

[Z. Nie et al.: WWW’07, H. Fang et al.: ECIR‘07, P. Serdyukov et al.: ECIR‘08, …]

query: keywords → answer: entities

$$s(e,q) = \lambda P[q \mid e] + (1-\lambda) P[q] \;\sim\; \prod \frac{P[q_i \mid e_i]}{P[q_i]} \;\sim\; KL\,(LM\,(q) \mid LM\,(e))$$

LM (entity e) = prob. distr. of words seen in context of e

query q: „*French player who won world championship*“

candidate entities:

e1: David Beckham

e2: Ruud van Nistelroy

e3: Ronaldinho

e4: Zinedine Zidane

e5: FC Barcelona

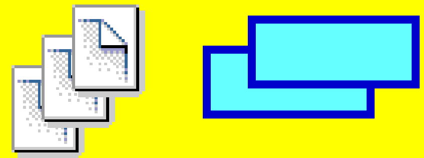
played for ManU, Real, LA Galaxy
David Beckham champions league
England lost match against France
married to spice girl …

Zizou champions league 2002
Real Madrid won final ...
Zinedine Zidane best player
France world cup 1998 ...

weighted by conf.

LM's: from Entities to Facts

Document / Entity LM's



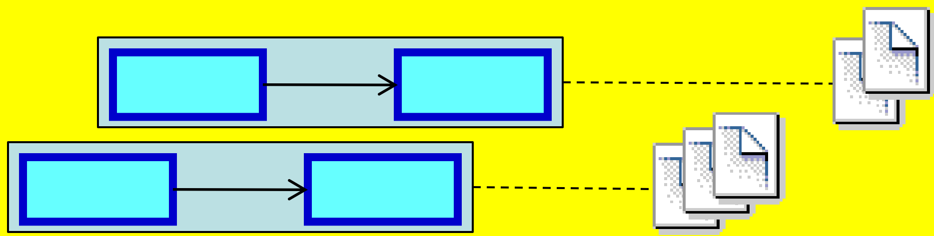
LM for doc/entity: prob. distr. of words

LM for query: (prob. distr. of) words

LM's: rich for docs/entities, super-sparse for queries

richer query LM with query expansion, etc.

Triple LM's



LM for facts: (degen. prob. distr. of) triple

LM for queries: (degen. prob. distr. of) triple pattern

LM's: apples and oranges

- *expand query variables by S,P,O values from DB/KB*
- *enhance with witness statistics*
- *query LM then is prob. distr. of triples !*

LM's for Triples and Triple Patterns

[G. Kasneci et al.: ICDE'08; S. Elbassuoni et al.: CIKM'09, ESWC'11]

triple patterns (queries q):

q: LM(q) + smoothing

q: Beckham p ManU	200/550
q: Beckham p Real	300/550
q: Beckham p Galaxy	20/550
q: Beckham p Milan	30/550

q: ?x p ASCannes

Zidane p ASCannes	20/30
Tidjani p ASCannes	10/30

q: ?x p ?y

Messi p FCBarcelon

Zidane p RealMadric

Kaka p ACMilan

...

q: Cruyff ?r FCBarcelona

Cruyff playedFor FCBarca	200/500
Cruyff playedAgainst FCBarca	50/500
Cruyff coached FCBarca	250/500

triples (facts f):

f1: Beckham p ManchesterU	200
f2: Beckham p RealMadrid	300
f3: Beckham p LAGalaxy	20
f4: Beckham p ACMilan	30
F5: Kaka p ACMilan	300
F6: Kaka p RealMadrid	150
f7: Zidane p ASCannes	20
f8: Zidane p Juventus	200
f14: Ribery p BayernMunich	100
f15: Drogba p Chelsea	150
f16: Casillas p RealMadrid	20

LM(q): {t → P [t | t matches q] ~ #witnesses(t)}

LM(answer f): {t → P [t | t matches f] ~ 1 for f}

smooth all LM's

rank results by ascending KL(LM(q)|LM(f))

witness statistics Σ: 2600

LM's for Composite Queries

q: Select ?x,?c Where { France ml ?x . ?x p ?c . ?c in UK . }

P [F ml Henry,
Henry p Arsenal,

$\sim \frac{2}{6}$

P [F ml Drogba,
Drogba p Chelsea,
Chelsea in UK]

$$\sim \frac{30}{650} \cdot \frac{150}{2600} \cdot \frac{140}{500}$$

- queries q with subqueries $q_1 \ldots q_n$
- results are **n-tuples of triples** $t_1 \ldots t_n$
- LM(q): $P[q_1 \ldots q_n] = \prod_i P[q_i]$
- LM(answer): $P[t_1 \ldots t_n] = \prod_i P[t_i]$
- $KL(LM(q)|LM(answer)) = \sum_i KL(LM(q_i)|LM(t_i))$

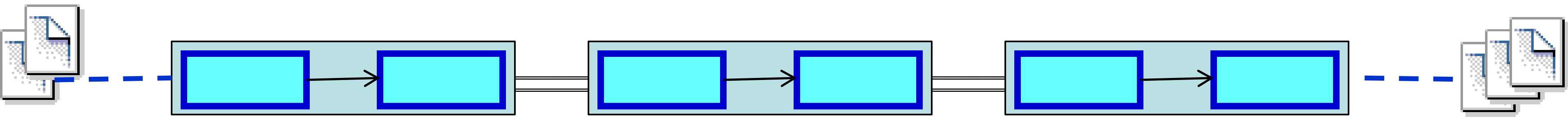
f21: F ml Zidane	200
f22: F ml Tidjani	20
f23: F ml Henry	200
f24: F ml Ribery	200
f25: F ml Drogba	30
f26: IC ml Drogba	100
f27 ALG ml Zidane	50

f1: Beckham p ManU	200
f7: Zidane p ASCannes	20
f8: Zidane p Juventus	200
f9: Zidane p RealMadrid	300
f10: Tidjani p ASCannes	10
f12: Henry p Arsenal	200
f13: Henry p FCBarca	150
f14: Ribery p Bayern	100
f15: Drogba p Chelsea	150

f31: ManU in UK	200
f32: Arsenal in UK	160
f33: Chelsea in UK	140

LM's for Keyword-Augmented Queries

q: Select ?x, ?c Where {
France ml ?x *[goalgetter, "top scorer"]* **.**
?x p ?c .
?c in UK *[champion, "cup winner", double]* **. }**



subqueries q_i with keywords $w_1 \ldots w_m$

results are still n-tuples of triples t_i

LM(q_i): $P[\text{triple } t_i \mid w_1 \ldots w_m] = \prod_k \beta P[t_i \mid w_k] + (1-\beta) P[t_i]$

LM(answer f_i) analogous

$KL(LM(q) | LM(\text{answer } f_i)) = \sum_i KL(LM(q_i) \mid LM(f_i))$

result ranking prefers (n-tuples of) triples whose witnesses score high on the subquery keywords

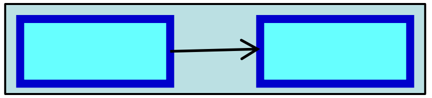
LM's for Temporal Phrases

[K.Berberich et al.: ECIR'10]

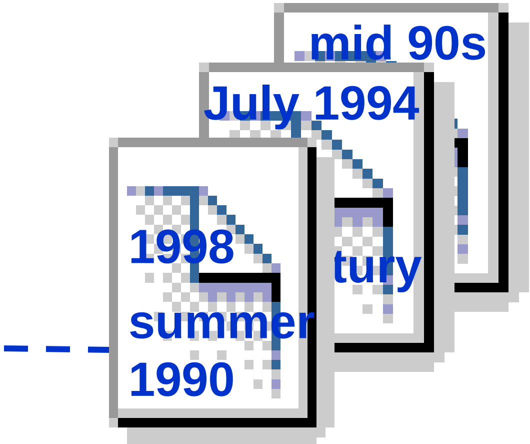
98 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 20

q: Select ?c Where {
?c instOf nationalTeam .
?c hasWon WorldCup *[nineties]* . }

"nineties" →
v = [1Jan90,31Dec99]



"mid 90s" →
x_j = [1Jan93,31Dec97]



normalize temp expr v in query

extract temp expr's x_j from witnesses & normalize

$P[q \mid f] \sim \ldots \sum_j P[v \mid x_j]$

$\sim \ldots \sum_j \sum \{ P[\,[b,e] \mid x_j] \mid \text{interval } [b,e] \in v \}$

$\sim \ldots \sum_j \text{overlap}(u,x_j) \,/\, \text{union}(u,x_j)$

- enhanced ranking
- efficiently computable
- plug into doc/entity/triples LM's

P[v=„nineties" | x_j = „mid 90s"] = 1/2
P[„nineties" | „summer 1990"] = 1/30
P[„nineties" | „last century"] = 1/10

Query Relaxation

q: … Where { France ml ?x . ?x p ?c . ?c in UK . }

$q^{(4)}$: … Where { France ml ?x . ?x p ?c . ?c in UK . }

[F ml Zidane,
Zidane p Real,

[IC ml Drogba,
Drogba p Chelsea,

[F resOf Drogba,
Drogba p Chelsea,

[IC ml Drogba,
Drogba p Chelsea,
Chelsea in UK]

$LM(q^*) = \lambda\ LM(q) + \lambda_1\ LM(q^{(1)}) + \lambda_2\ LM(q^{(2)}) + \dots$

replace **e** in q by $e^{(i)}$ in $q^{(i)}$:
precompute P:=LM (e ?p ?o)
and Q:=LM ($e^{(i)}$?p ?o)
set $\lambda_i \sim 1/2$ (KL (P|Q) + KL (Q|P))

replace r in q by $r^{(i)}$ in $q^{(i)}$ → LM (?s $r^{(i)}$?o)
replace e in q by ?x in $q^{(i)}$ → LM (?x r ?o)
…

LM‘s of e, r, ...
are prob. distr.‘s
of triples !

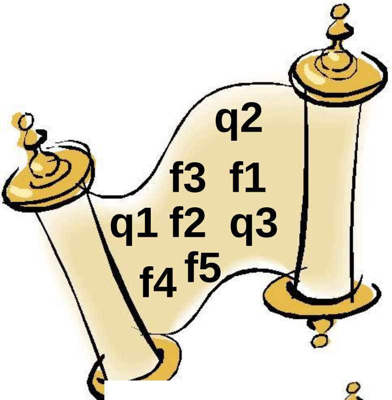
f21: F ml Zidane	200
f22: F ml Tidjani	20
F23: F ml Henry	200
F24: F ml Ribery	200
F26: IC ml Drogba	100
F27 ALG ml Zidane	50

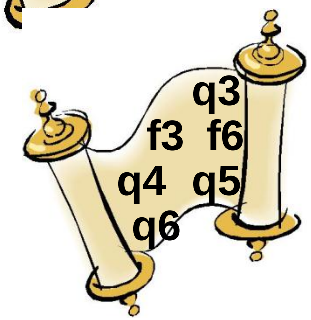
f1: Beckham p ManU	200
f7: Zidane p ASCannes	20
f9: Zidane p Real	300
f10: Tidjani p ASCannes	10
f12: Henry p Arsenal	200
f15: Drogba p Chelsea	150

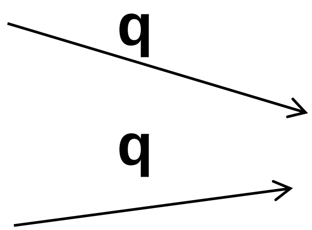
f32: Arsenal in UK	160
f33: Chelsea in UK	140

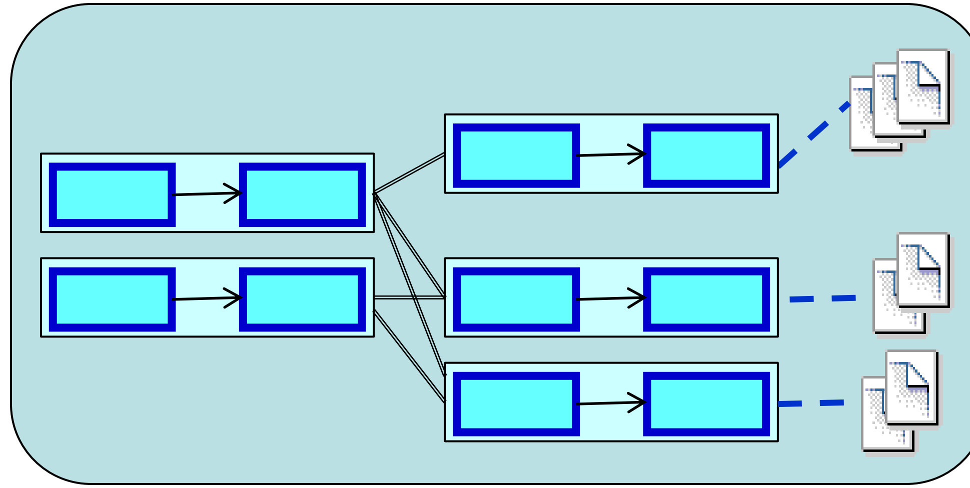
Result Personalization

[S. Elbassuoni et al.: PersDB'08]









same answer for everyone?

Personal histories of queries & clicked facts
→ LM(user u):
prob. distr. of triples !

$LM(q|u] = \mu\ LM(q) + (1-\mu)\ LM(u)$
then business as usual

u1 [classical music] → q: ?p from Europe . ?p hasWon AcademyAward

u2 [romantic comedy] → q: ?p from Europe . ?p hasWon AcademyAward

u3 [from Africa] → q: ?p isa SoccerPlayer . ?p hasWon ?a

Open issue: „insightful“ results (new to the user)

Result Diversification

[J. Carbonell, J. Goldstein: SIGIR‘98]

q: Select ?p, ?c Where { ?p isa SoccerPlayer . ?p playedFor ?c . }

1 Beckham, ManchesterU
2 Beckham, RealMadrid
3 Beckham, LAGalaxy
4 Beckham, ACMilan
5 Zidane, RealMadrid
6 Kaka, RealMadrid
7 Cristiano Ronaldo, RealMadrid
8 Raul, RealMadrid
9 van Nistelrooy, RealMadrid
10 Casillas, RealMadrid

1 Beckham, ManchesterU
2 Beckham, RealMadrid
3 Zidane, RealMadrid
4 Kaka, ACMilan
5 Cristiano Ronaldo, ManchesterU
6 Messi, FCBarcelona
7 Henry, Arsenal
8 Ribery, BayernMunich
9 Drogba, Chelsea
10 Luis Figo, Sporting Lissabon

rank results f_1 ... f_k by ascending

δ **KL(LM(q) | LM(f_i))** – (1–δ) KL(LM(f_i) | LM($\{f_1..f_k\} \setminus \{f_i\}$))

implemented by greedy re-ranking of f_i‘s in candidate pool

Take-Home Message (Ranking)

- **Don‘t re-invent the wheel:**

  LM‘s are elegant and expressive means for ranking

  consider both data & workload statistics

- **Extensions should be conceptually simple:**

  can capture informativeness, personalization,

  relaxation, diversity – all in same framework

- **Unified ranking model for complete query language:**

  still work to do

Outline

✓ **Motivation**

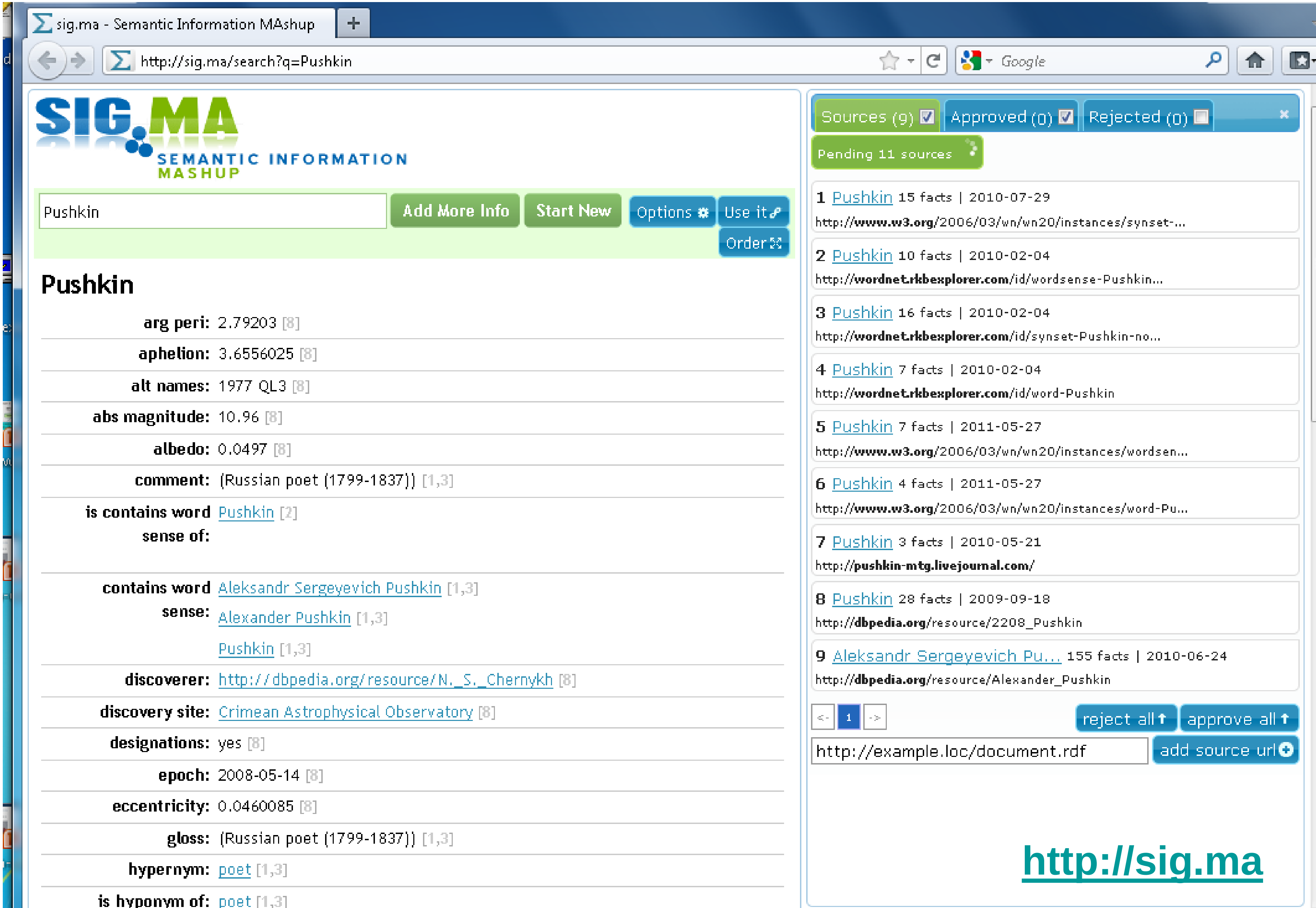
✓ **Searching for Entities & Relations**

✓ **Informative Ranking**

★ **Entity-Name Disambiguation**

★ **Wrap-up**

Entity Diversity



http://sig.ma

Entity-Name Ambiguity

<sameAs>

interlinking the Web of Data

The Web of Data has many equivalent URIs.

This service helps you to find co-references between different data sets.

Enter a known URI, or use Sindice to search first.

Search results from Sindice, with co-references applie

Sindice "2208 Pushkin"

<sameAs> {
1. http://dbpedia.org/resource/2208_Pushkin
2. http://rdf.freebase.com/ns/guid.9202a8c04000641f8000000007d408a8

rdf+xml · n3 · json · text

Sindice "Pushkin"

<sameAs> {
1. http://wordnet.rkbexplorer.com/id/synset-Pushkin-noun-1
2. http://www.w3.org/2006/03/wn/wn20/instances/synset-Pushkin-noun-1

rdf+xml · n3 · json · text

About: 2208 Pushkin

An Entity of Type : Thing, from Named Graph : http://dbpedia.org, within Data Space : dbpedia.org

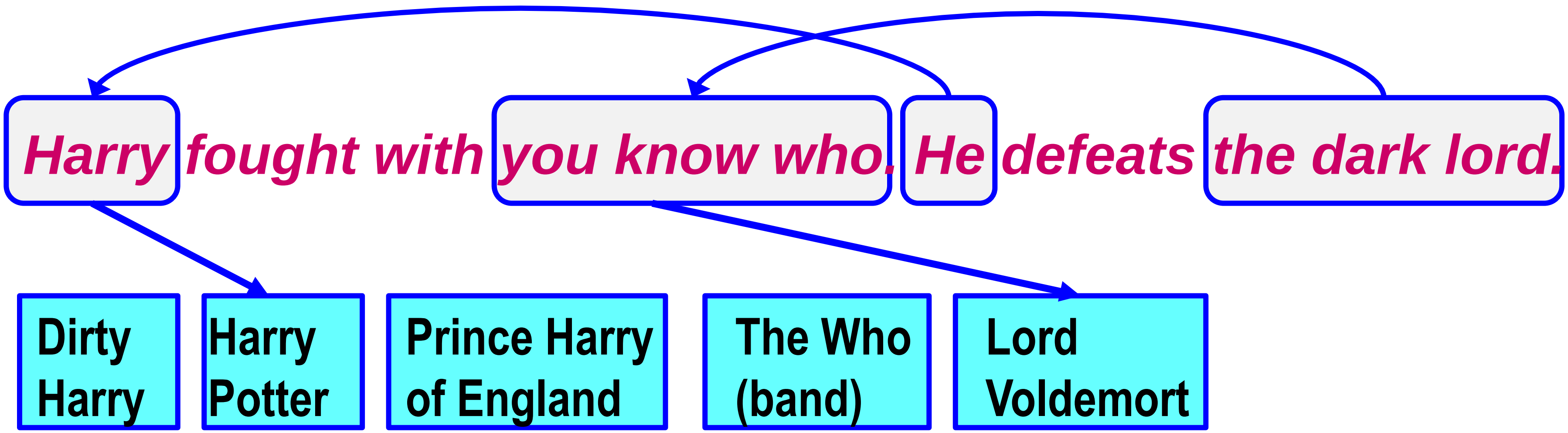
DBpedia

2208 Pushkin (1977 QL3) is an outer main-belt asteroid discovered on August 22, 1977 by N. S.

Property	Value
http://dbpedia.org/meta/editlink	http://en.wikipedia.org/w/index.php?title=2208_Pushkin&action=edit
http://dbpedia.org/meta/pageid	16477278 (xsd:integer)
http://dbpedia.org/meta/revision	http://en.wikipedia.org/w/index.php?title=2208_Pushkin&oldid=438390754
dbpedia-owl:abstract	▪ 2208 Pushkin è un asteroide della fascia principale del diametro medio di circa 38,31 km. Scoperto nel 1977, presenta un'orbita caratterizzata da un semiasse maggiore pari a 3,4921755 UA e da un'eccentricità di 0,0468605, inclinata di 5,41668° rispetto all'eclittica. ▪ 2208 Pushkin (1977 QL3) – planetoida z grupy pasa głównego asteroid, okrążająca Słońce w ciągu 6,53 lat, w średniej odległości 3,49 j.a. Odkryta 22 sierpnia 1977 roku. ▪ Pushkin (asteróide 2208) é um asteróide da cintura principal com um diâmetro de 38,31 quilómetros, a 3,3281379 UA. Possui uma excentricidade de 0,0470008 e um período orbital de 2 383,75 dias. Pushkin tem uma velocidade orbital média de 15,93816614 km/s e uma inclinação de 5,41668º. Este asteróide foi descoberto em 22 de Agosto de 1977 por Nikolai Chernykh. ▪ 2208 Pushkin (1977 QL3) is a Outer Main-belt Asteroid discovered on August 22, 1977 by N. S. Chernykh at the Crimean Astrophysical Observatory. It was named after the leading writer of Russia, Pushkin.
dbpprop:absMagnitude	10.96 (xsd:double)
dbpprop:abstract_live	2208 Pushkin (1977 QL3) is an outer main-belt asteroid discovered on August 22, 1977 by N. S. Chernykh at the Crimean Astrophysical Observatory. It was named after the leading writer of Russia, Pushkin.
dbpprop:albedo	0.0497 (xsd:double)
dbpprop:altNames	1977 QL3
dbpprop:aphelion	3.6556025 (xsd:double)
dbpprop:argPeri	2.79203 (xsd:double)
dbpprop:ascNode	79.48005 (xsd:double)
dbpprop:bgcolour	FFFFC0
dbpprop:comment_live	2208 Pushkin (1977 QL3) is an outer main-belt asteroid discovered on August 22, 1977 by N. S.

http://sameas.org

Named-Entity Disambiguation



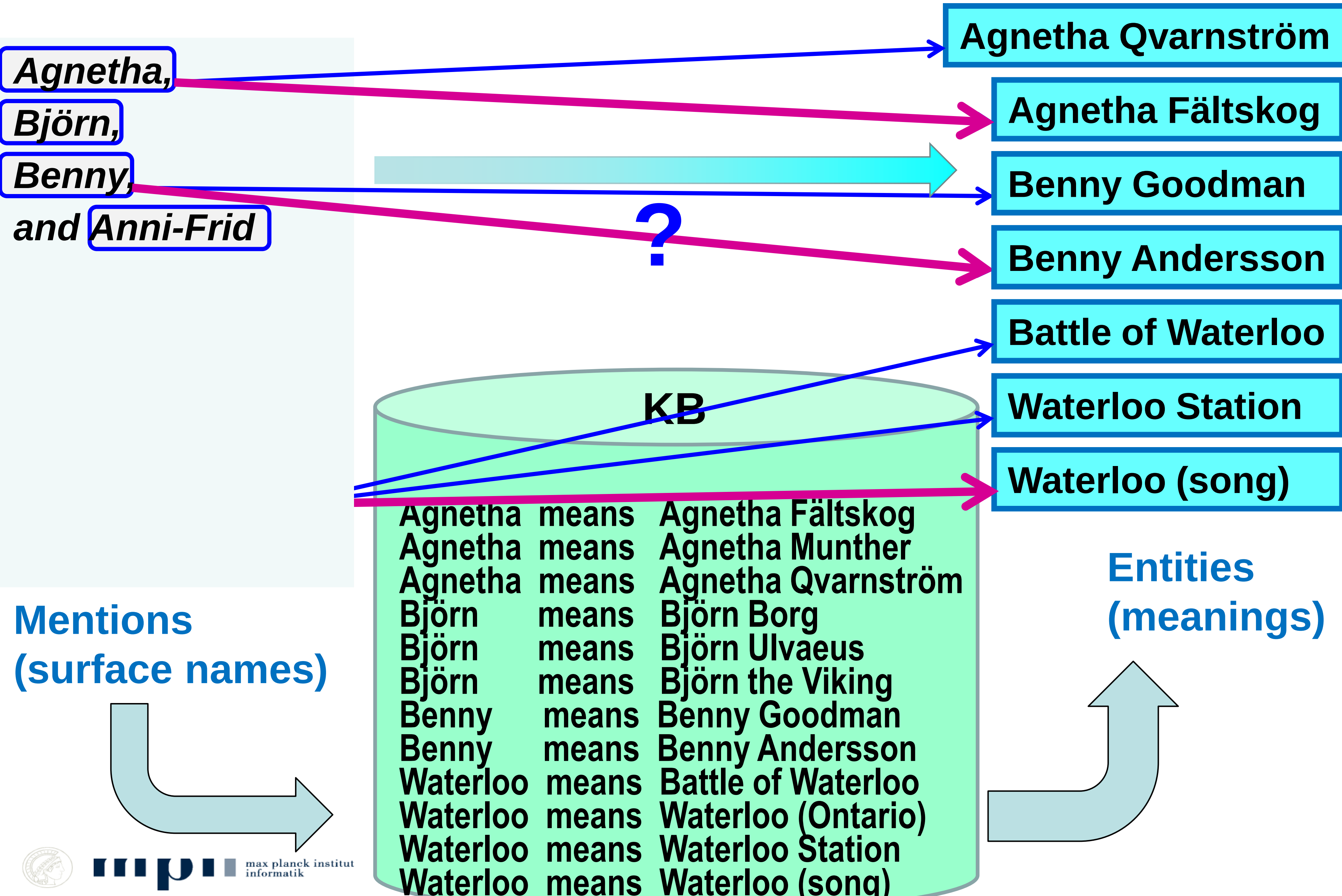
Three NLP tasks:

1) **named-entity detection: segment & label by HMM or CRF (e.g. Stanford NER tagger)**

2) **co-reference resolution: link to preceding NP (trained classifier over linguistic features)**

3) **named-entity disambiguation: map each mention (name) to canonical entity (entry in KB)**

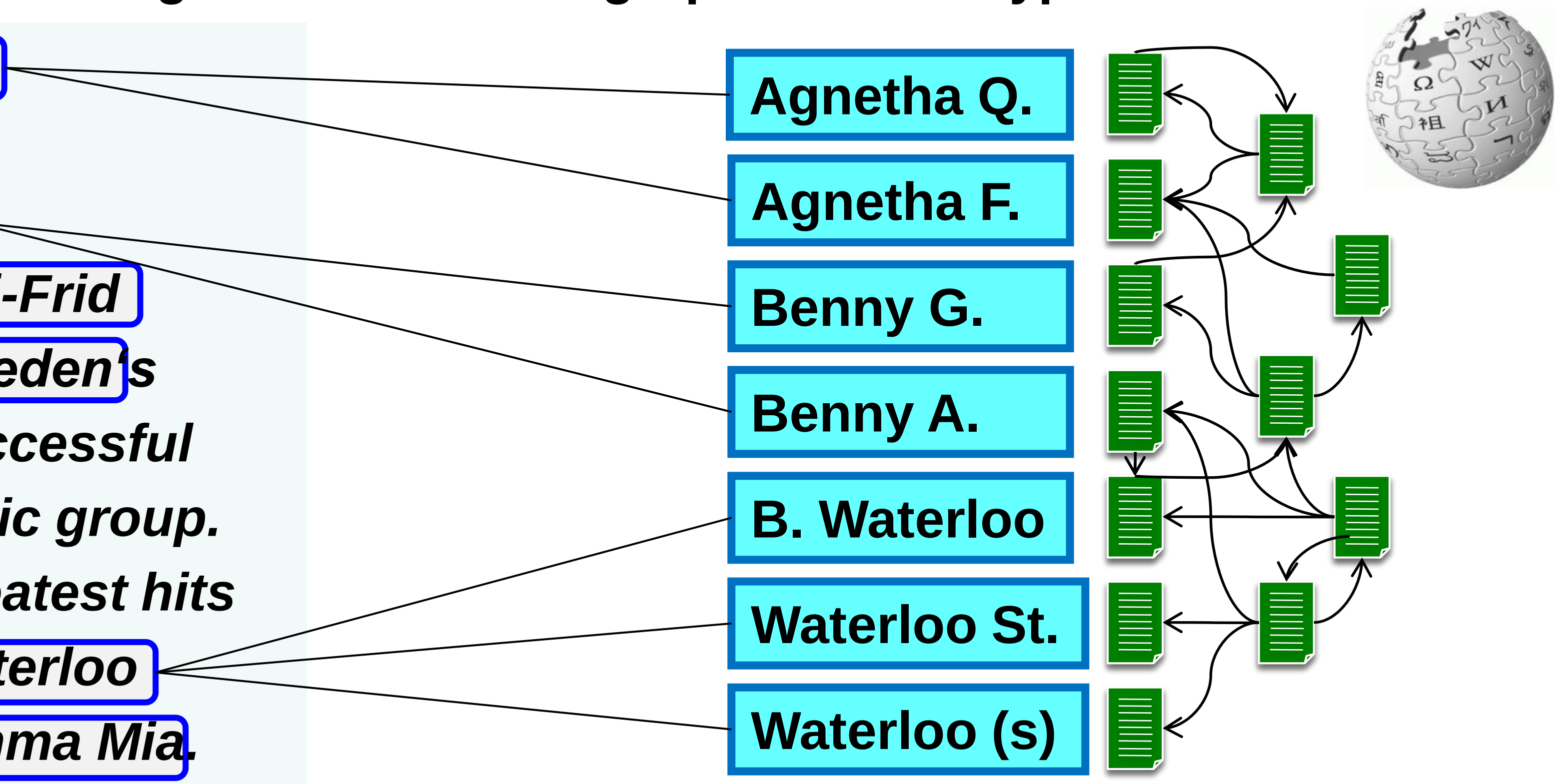
Mentions, Meanings, Mappings



Mention-Entity Graph

weighted undirected graph with two types of nodes

Agnetha, Björn, Benny, and Anni-Frid were Sweden's most successful pop music group. Their greatest hits were Waterloo and Mamma Mia.

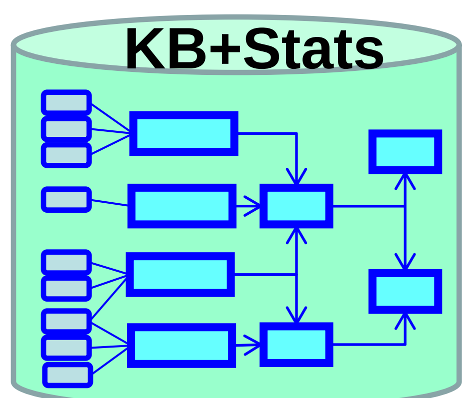


Popularity (m,e):

- freq(m,e|m)
- length(e)
- #links(e)

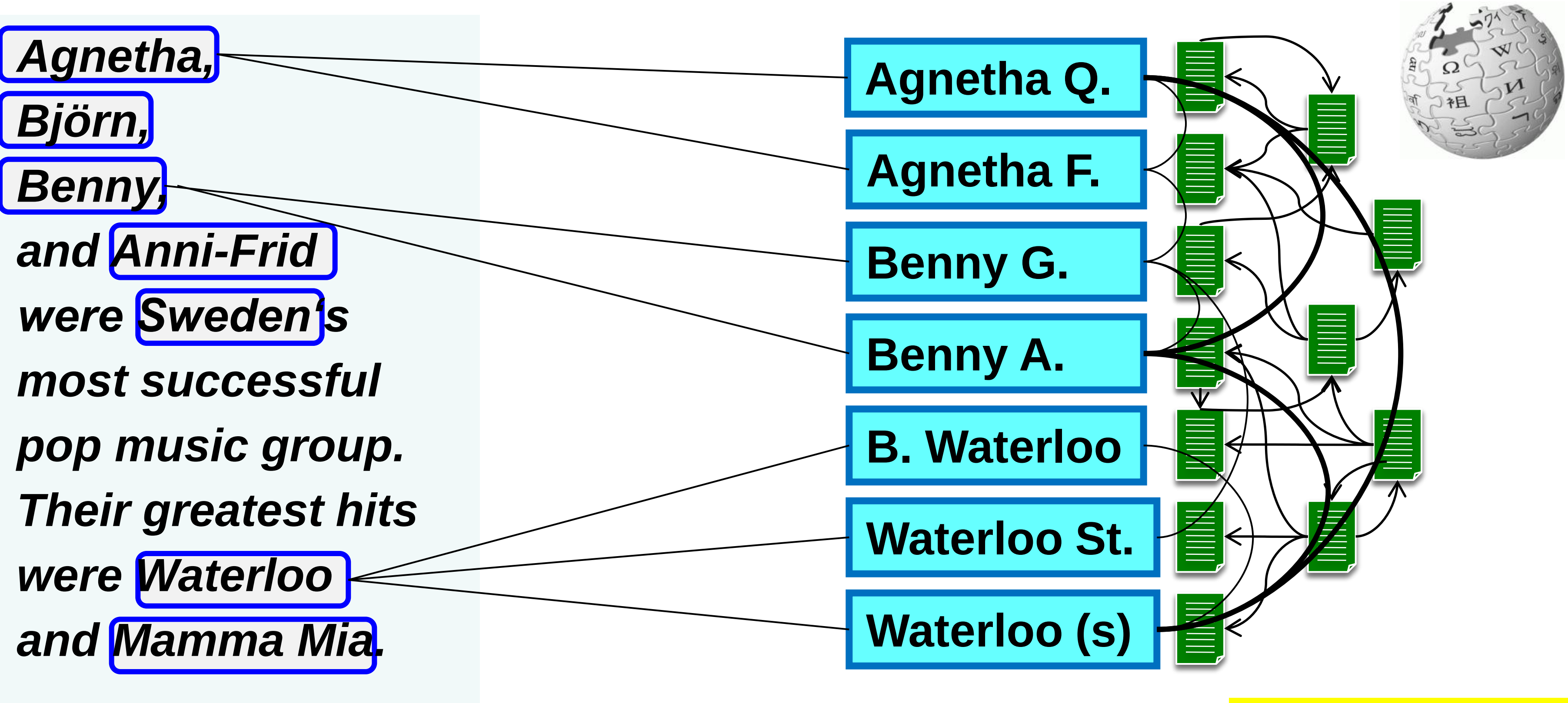
Similarity (m,e):

- cos/Dice/KL (context(m), context(e))



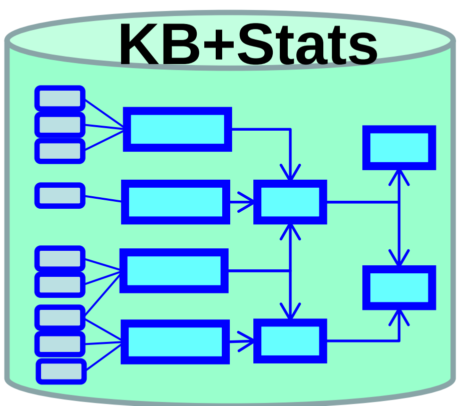
Mention-Entity Graph

weighted undirected graph with two types of nodes



Popularity (m,e):
- freq(m,e|m)
- length(e)
- #links(e)

Similarity (m,e):
- cos/Dice/KL (context(m), context(e))



Coherence (e,e'):
- dist(types)
- overlap(links)
- overlap (anchor words)

Mention-Entity Graph

weighted undirected graph with two types of nodes

Agnetha, Björn, Benny, and Anni-Frid were Sweden's most successful pop music group. Their greatest hits were Waterloo and Mamma Mia.

- Agnetha Q.
- Agnetha F.
- Benny G.
- Benny A.
- B. Waterloo
- Waterloo St.
- Waterloo (s)

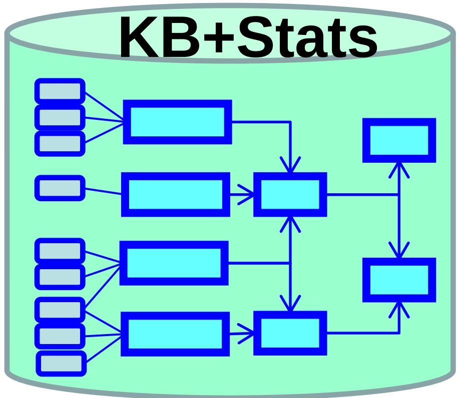
Swedish female singers
people from Jönköping
singers
musicians

Swedish songwriters
people from Stockholm
composers
musicians

ABBA songs
#1 chart singles
songs
artifacts

Popularity (m,e):
- freq(m,e|m)
- length(e)
- #links(e)

Similarity (m,e):
- cos/Dice/KL (context(m), context(e))



Coherence (e,e'):
- dist(types)
- overlap(links)
- overlap (anchor words)

Mention-Entity Graph

weighted undirected graph with two types of nodes

Agnetha, Björn, Benny, and Anni-Frid were Sweden's most successful pop music group. Their greatest hits were Waterloo and Mamma Mia.

- Agnetha Q.
- Agnetha F.
- Benny G.
- Benny A.
- B. Waterloo
- Waterloo St.
- Waterloo (s)

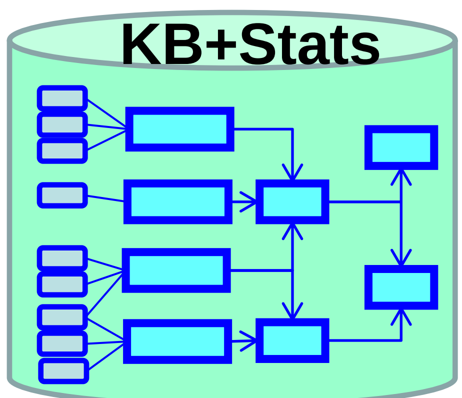
http://.../wiki/ABBA
http://.../wiki/Anni-Frid_Lyngstad
http://.../wiki/Jönköping
http://.../wiki/Eurovision_Song_Cor

http://.../wiki/ABBA
http://.../wiki/Anni-Frid_Lyngstad
http://.../wiki/Mamma_Mia!
http://.../wiki/Agnetha_Fältskog

http://.../wiki/ABBA
http://.../wiki/Eurovision_Song_Cor
http://.../wiki/Mamma_Mia!

Popularity (m,e):
- freq(m,e|m)
- length(e)
- #links(e)

Similarity (m,e):
- cos/Dice/KL (context(m), context(e))



Coherence (e,e'):
- dist(types)
- overlap(links)
- overlap (anchor words)

Mention-Entity Graph

weighted undirected graph with two types of nodes

Agnetha, Björn, Benny, and Anni-Frid were Sweden's most successful pop music group. Their greatest hits were Waterloo and Mamma Mia.

- Agnetha Q.
- Agnetha F.
- Benny G.
- Benny A.
- B. Waterloo
- Waterloo St.
- Waterloo (s)

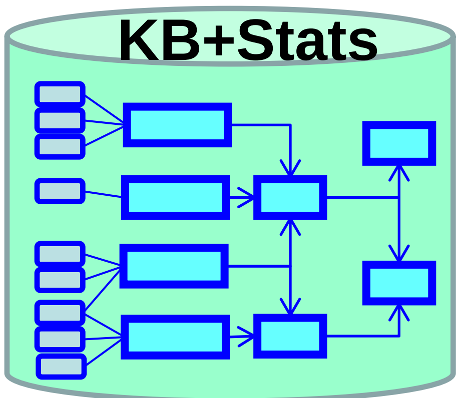
pop group ABBA
best-selling music artist in history
Melodifestivalen
The Winner Takes It All

pop group ABBA
Grammy Award nomination
Melodifestivalen
Mamma Mia!

Agnetha Fältskog
Benny Andersson
number-one single in Norway
Mamma Mia!

Popularity (m,e):
- freq(m,e|m)
- length(e)
- #links(e)

Similarity (m,e):
- cos/Dice/KL (context(m), context(e))



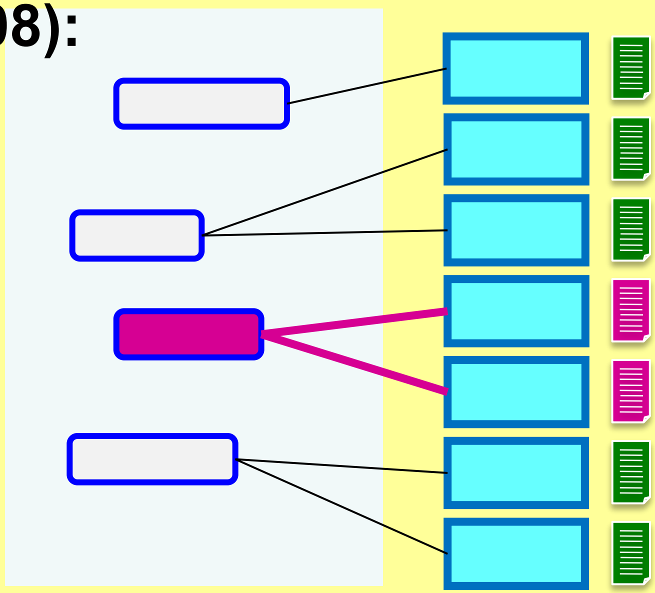
Coherence (e,e'):
- dist(types)
- overlap(links)
- overlap (anchor words)

Different Approaches

Combine Popularity, Similarity, and Coherence Features

(Cucerzan: EMNLP'07, Milne/Witten: CIKM'08):

- for **sim (context(m), context(e))**:
  consider **surrounding mentions**
  and their **candidate entities**
- use their types, links, anchors
  as **features of context(m)**
- set m-e edge weights accordingly
- use greedy methods for solution

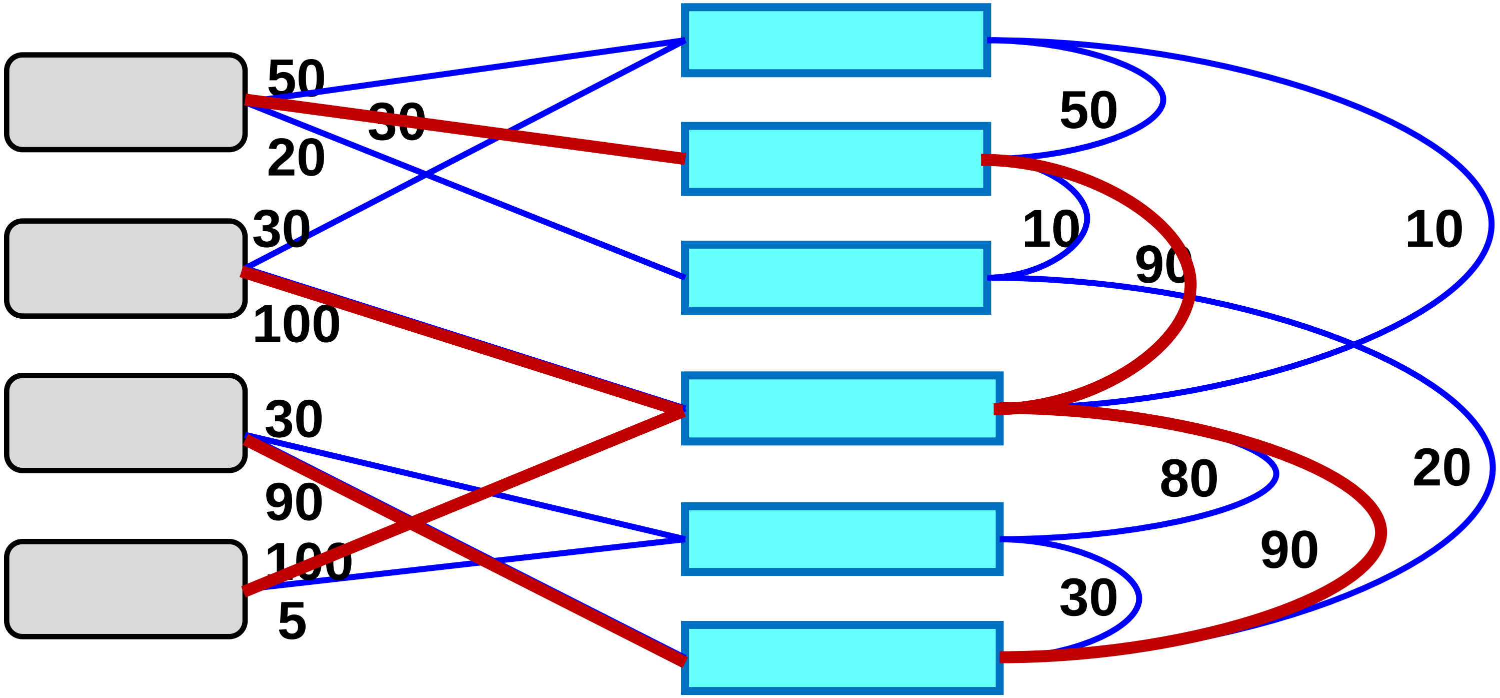


Collective Learning with Prob. Factor Graphs

(Chakrabarti et al.: KDD'09):

- model **P[m|e]** by similarity and **P[e1|e2]** by coherence
- consider **likelihood** of **P[m1 … mk | e1 … ek]**
- **factorize** by all **m-e pairs** and **e1-e2 pairs**
- use hill-climbing, LP, etc. for solution

Joint Mapping



- Build mention-entity graph or joint-inference factor graph from knowledge and statistics in KB
- Compute high-likelihood mapping (ML or MAP) or dense subgraph such that:
  each m is connected to exactly one e (or at most one e)

K. Kulkarni et al.: Collective Annotation of Wikipedia Entities in Web Text, KDD'09
J. Hoffart et al.: Robust Disambiguation of Named Entities in Text, EMNLP'11

Mention-Entity Similarity Edges

Precompute characteristic keyphrases q for each entity e: anchor texts or noun phrases in e page with high PMI:

$$weight(q,e) = \log \frac{freq(q,e)}{freq(q)\, freq(e)}$$

„Eurovision song contest“

Match keyphrase q of candidate e in context of mention m

$$score(q \mid e) \sim \frac{\#\ matching\ words}{length\ of\ cover(q)} \left(\frac{\sum_{w \in cover(q)} weight(w \mid e)}{\sum_{w \in q} weight(w \mid e)} \right)^{1+\gamma}$$

Extent of partial matches **Weight of matched words**

Benny won the Grand Prix Eurovision de la Chanson, a contest for pop songs

Compute overall similarity of context(m) and candidate e

$$score(e \mid m) \sim \sum_{\substack{q \in keyphrases(e) \\ in\ context(m)}} score(q)\, dist(cover(q), m)^{-\alpha}$$

Entity-Entity Coherence Edges

Precompute overlap of incoming links for entities e1 and e2

$$mw\text{-}coh(e1, e2) \sim 1 - \frac{\log \max(in(e1, e2)) - \log(in(e1) \cap in(e2))}{\log |E| - \log \min(in(e1), in(e2))}$$

Alternatively compute overlap of anchor texts for e1 and e2

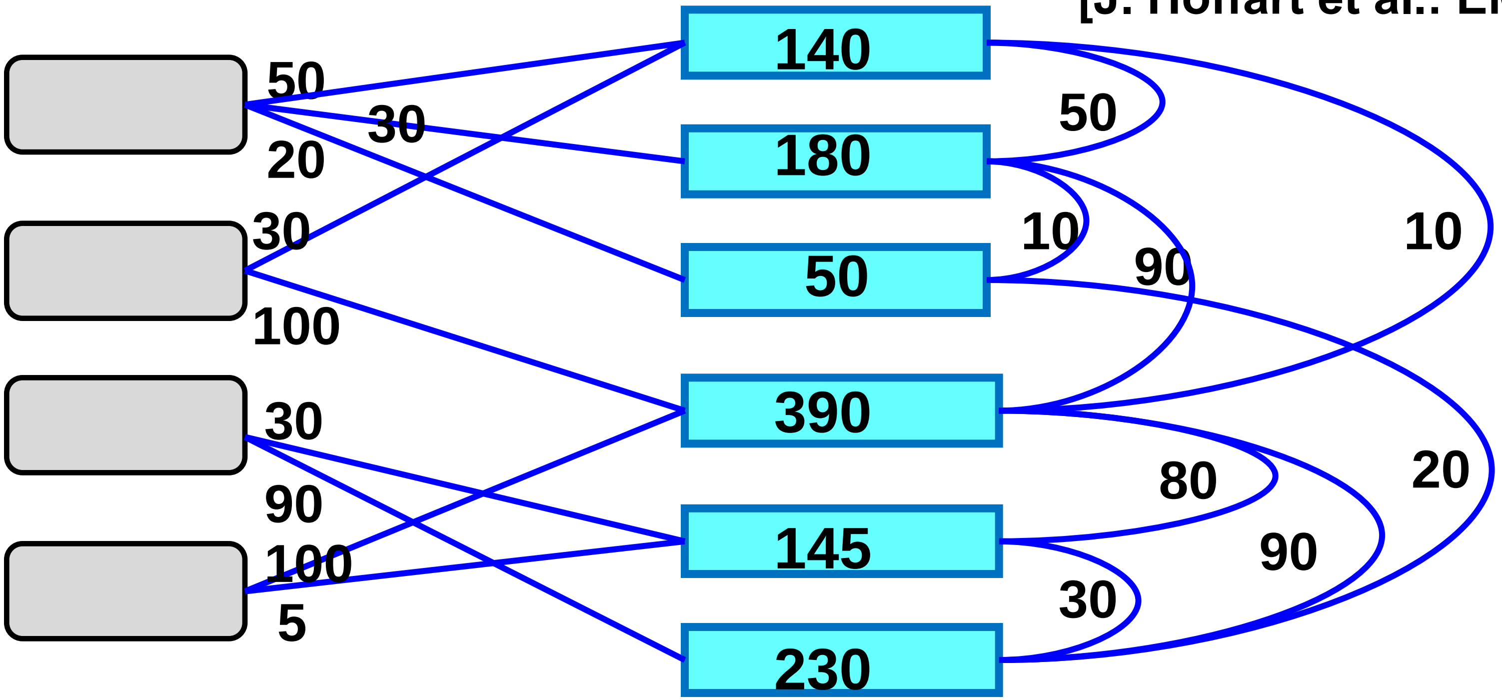
$$ngram\text{-}coh(e1, e2) \sim \frac{|ngrams(e1) \cap ngrams(e2)|}{|ngrams(e1) \cup ngrams(e2)|}$$

or overlap of keyphrases, or similarity of bag-of-words, or …
optionally filtered by words or n-grams in entire input text

Optionally combine with type distance of e1 and e2
(e.g., Jaccard index for type instances)
and other (precomputed) measures

Coherence Graph Algorithm

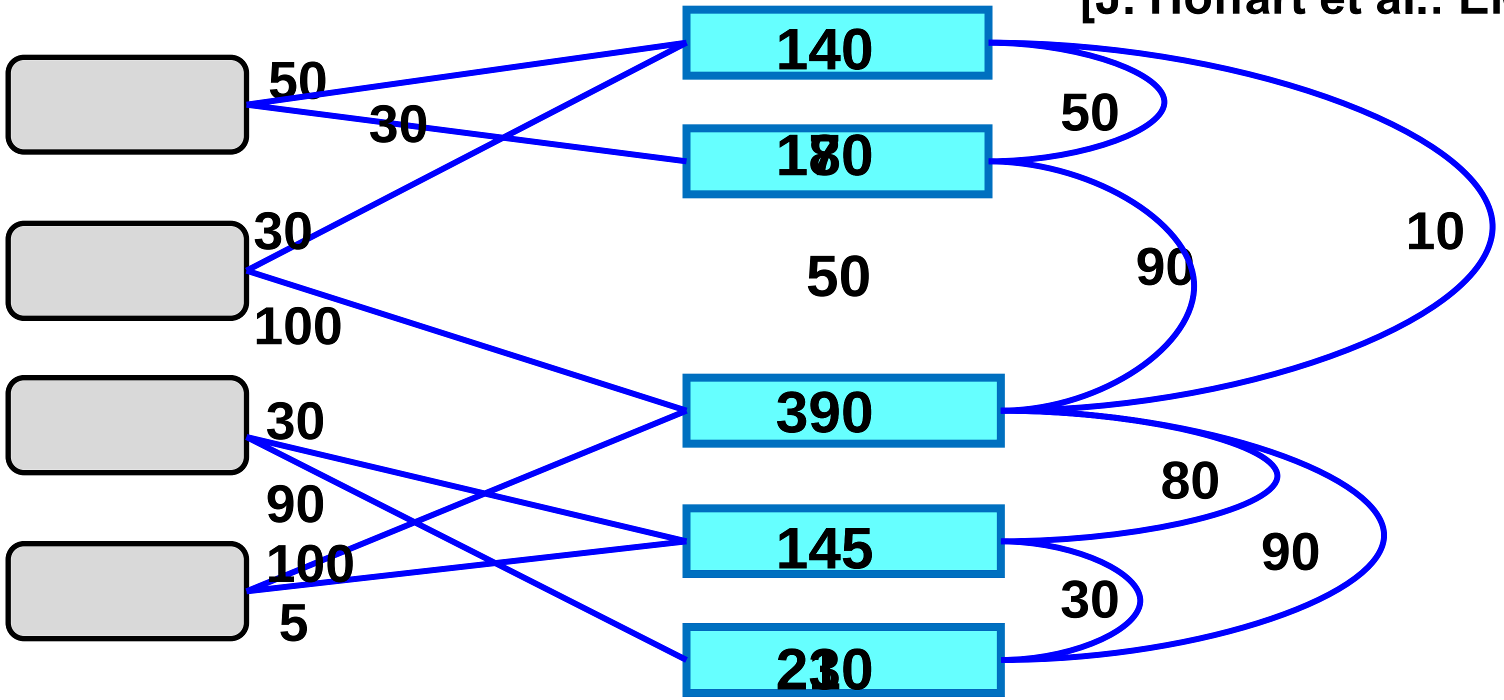
[J. Hoffart et al.: EMNLP'11]



- Compute dense subgraph to
  maximize min weighted degree among entity nodes
  such that:
  each m is connected to exactly one e (or at most one e)
- Greedy approximation:
  iteratively remove weakest entity and its edges
- Keep alternative solutions, then use local/randomized search

Coherence Graph Algorithm

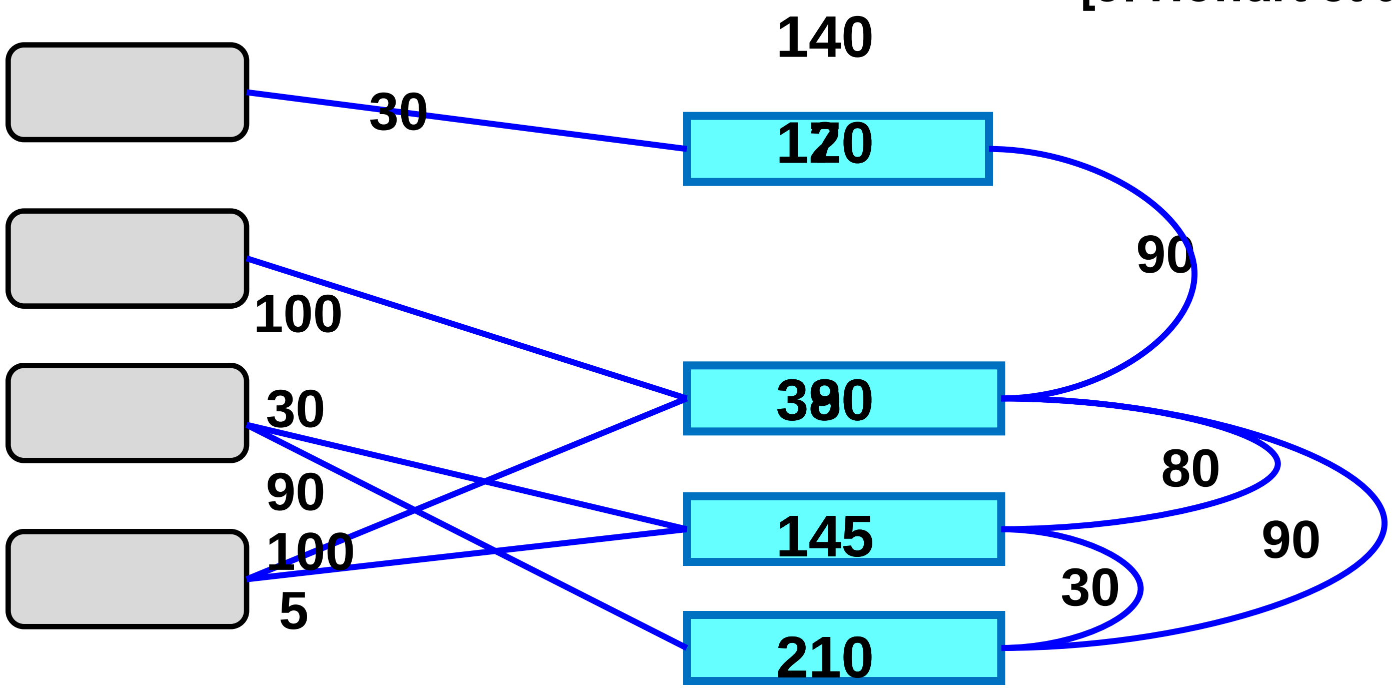
[J. Hoffart et al.: EMNLP'11]



- Compute **dense subgraph** to
  maximize **min weighted degree** among entity nodes
  such that:
  each m is **connected to exactly one** e (or **at most one** e)
- **Greedy** approximation:
  iteratively remove weakest entity and its edges
- Keep alternative solutions, then use local/randomized search

Coherence Graph Algorithm

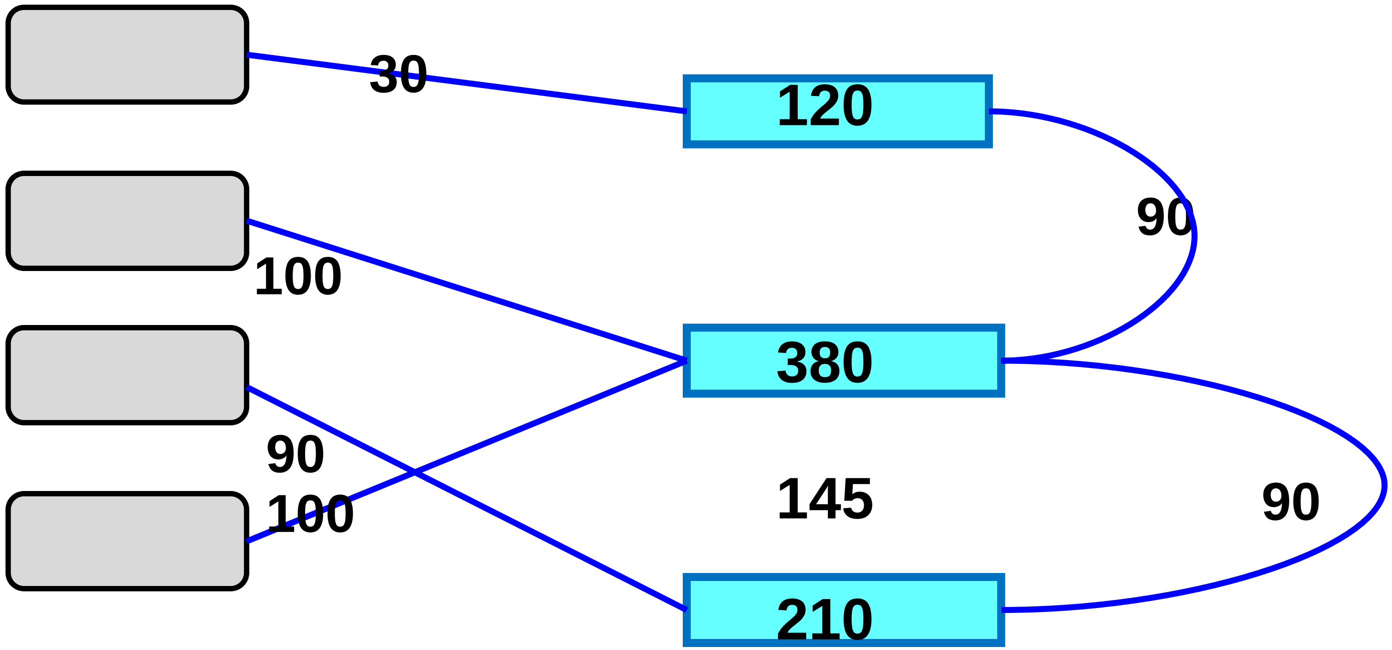
[J. Hoffart et al.: EMNLP'11]



- Compute dense subgraph to
    maximize min weighted degree among entity nodes
  such that:
    each m is connected to exactly one e (or at most one e)
- Greedy approximation:
    iteratively remove weakest entity and its edges
- Keep alternative solutions, then use local/randomized search

Coherence Graph Algorithm

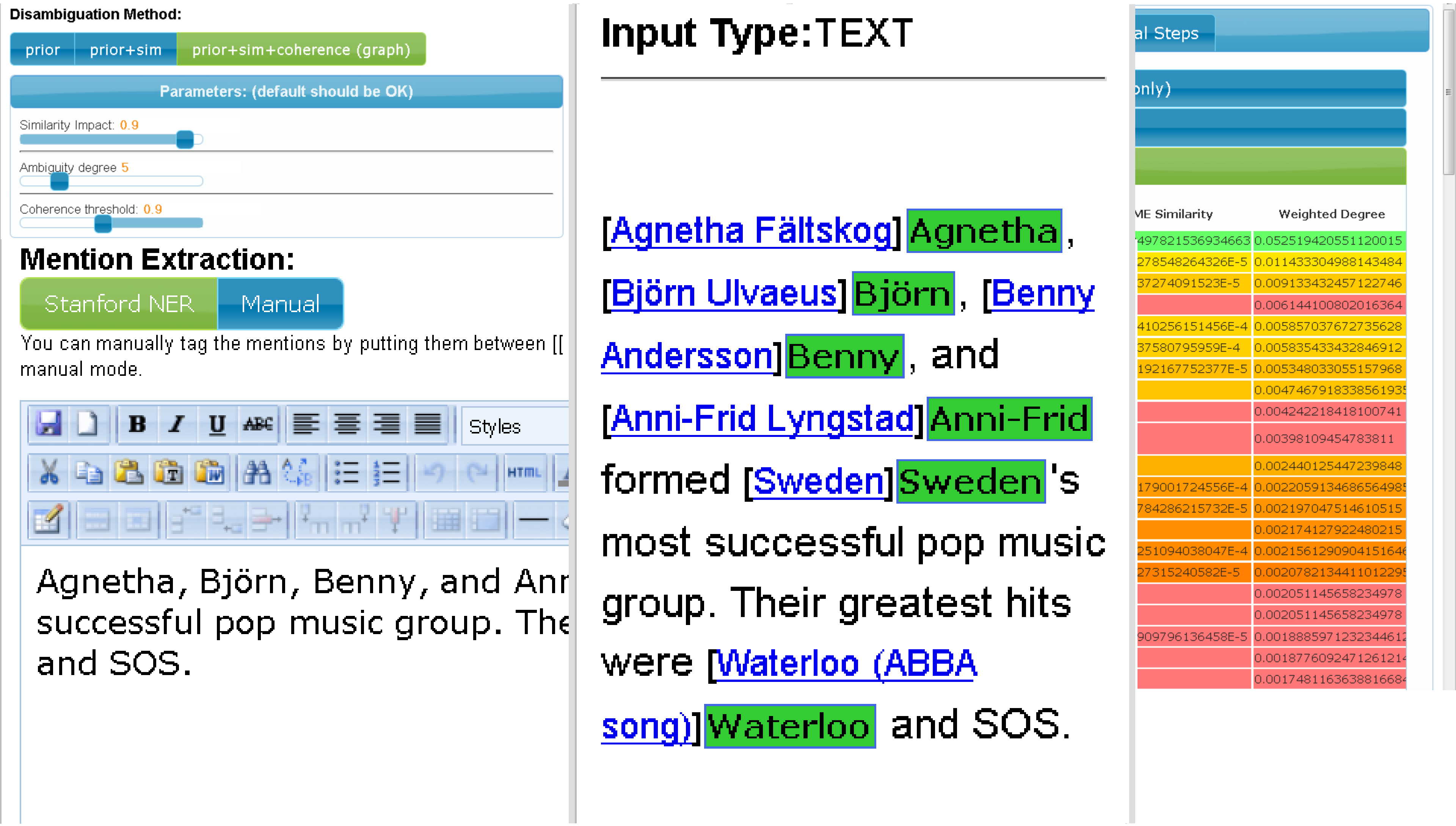
[J. Hoffart et al.: EMNLP‘11]



- Compute dense subgraph to
  maximize min weighted degree among entity nodes
  such that:
  each m is connected to exactly one e (or at most one e)
- Greedy approximation:
  iteratively remove weakest entity and its edges
- Keep alternative solutions, then use local/randomized search

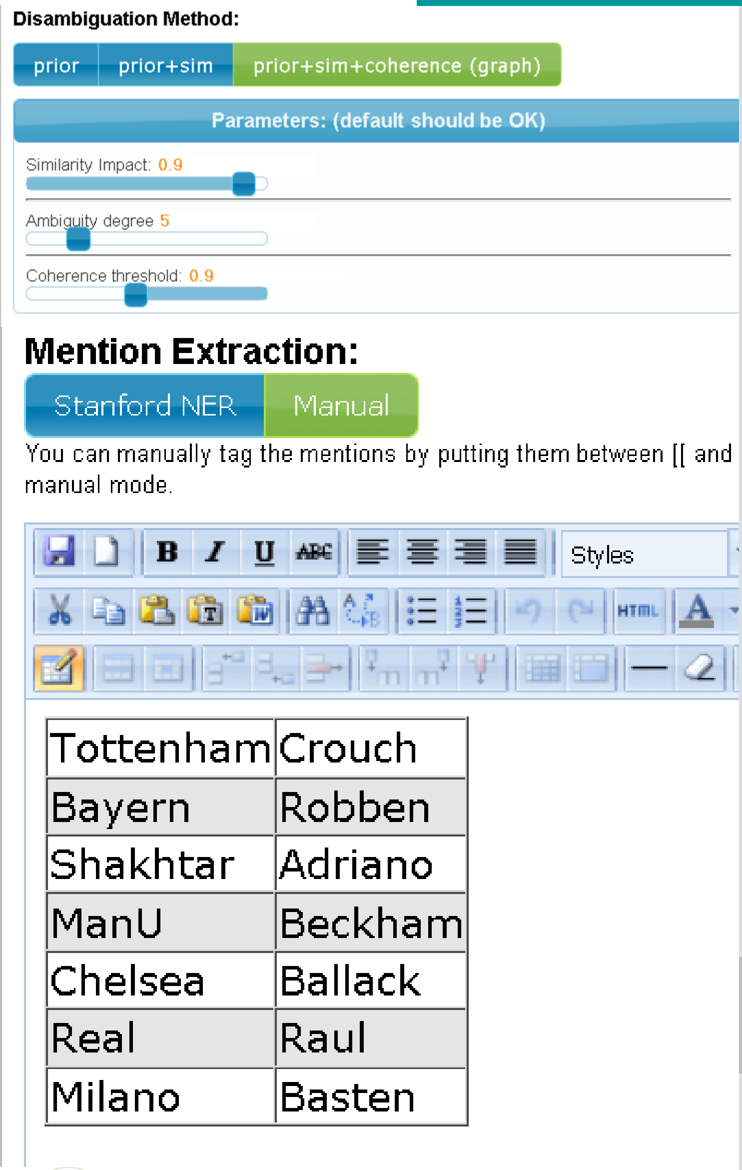
AIDA Accurate Online Disambiguation

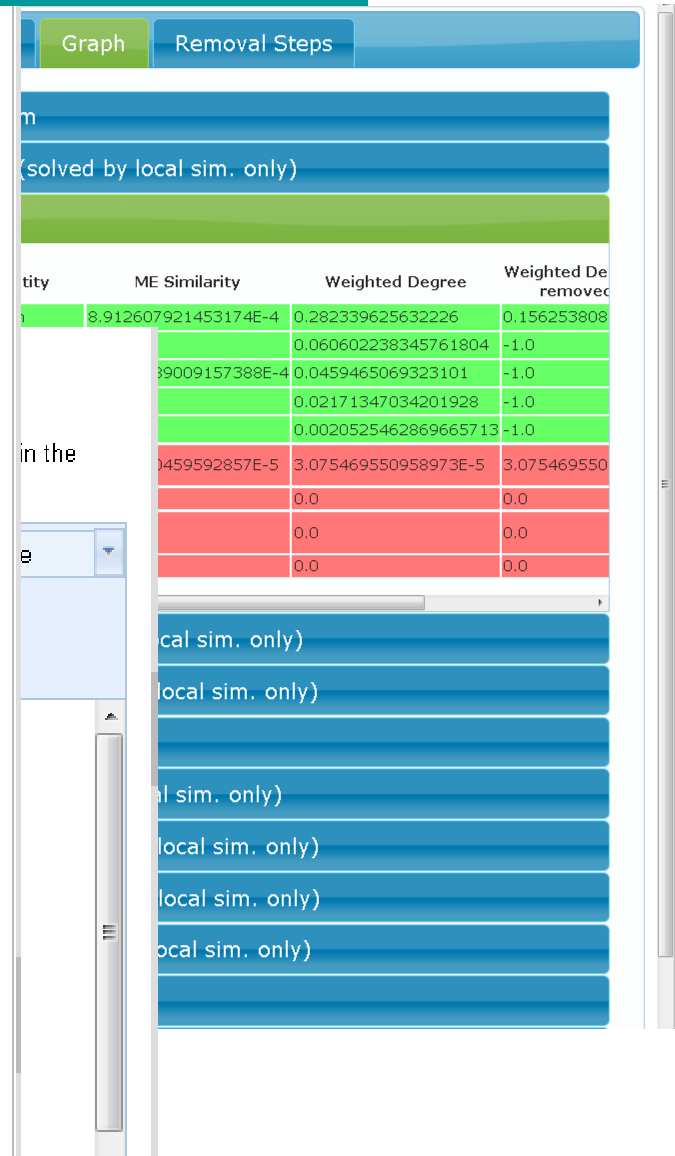
http://www.mpi-inf.mpg.de/yago-naga/aida/



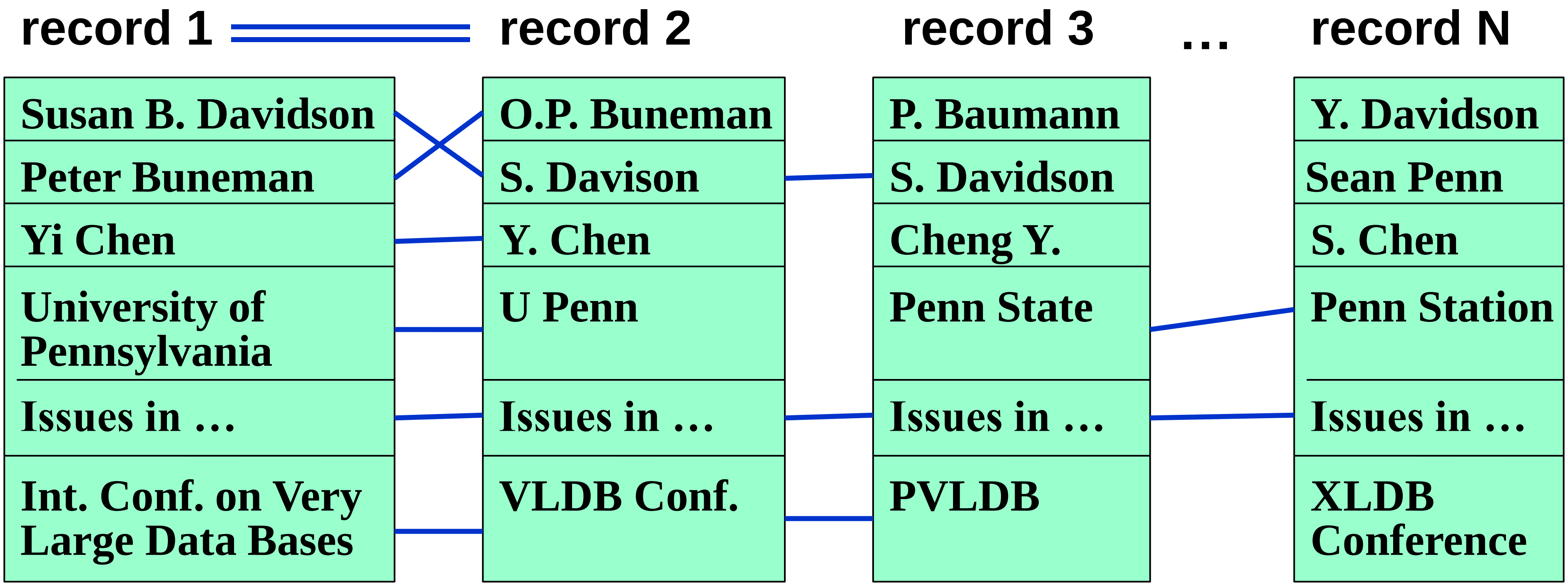
AIDA Accurate Online Disambiguation

http://www.mpi-inf.mpg.de/yago-naga/aida/





Record Linkage (Entity Resolution)



Find equivalence classes of entities, and records, based on:

- similarity of values (edit distance, n-gram overlap, etc.)
- joint agreement of linkage

→ similarity joins, grouping/clustering, collective learning, etc.

Halbert L. Dunn: Record Linkage. American Journal of Public Health. 1946
H.B. Newcombe et al.: Automatic Linkage of Vital Records. Science, 1959.

Record Linkage (Entity Resolution)

record 1	record 2	record 3	…	record N
Susan B. Davidson	O.P. Buneman	P. Baumann		Y. Davidson
Peter Buneman	S. Davison	S. Davidson		Sean Penn
Yi Chen	Y. Chen	Cheng Y.		S. Chen
University of Pennsylvania	U Penn	Penn State		Penn Station
Issues in …				
Int. Conf. on Large Data B				

Find equiv

- simila
- joint a

→ similari

prob. / uncertain rules:

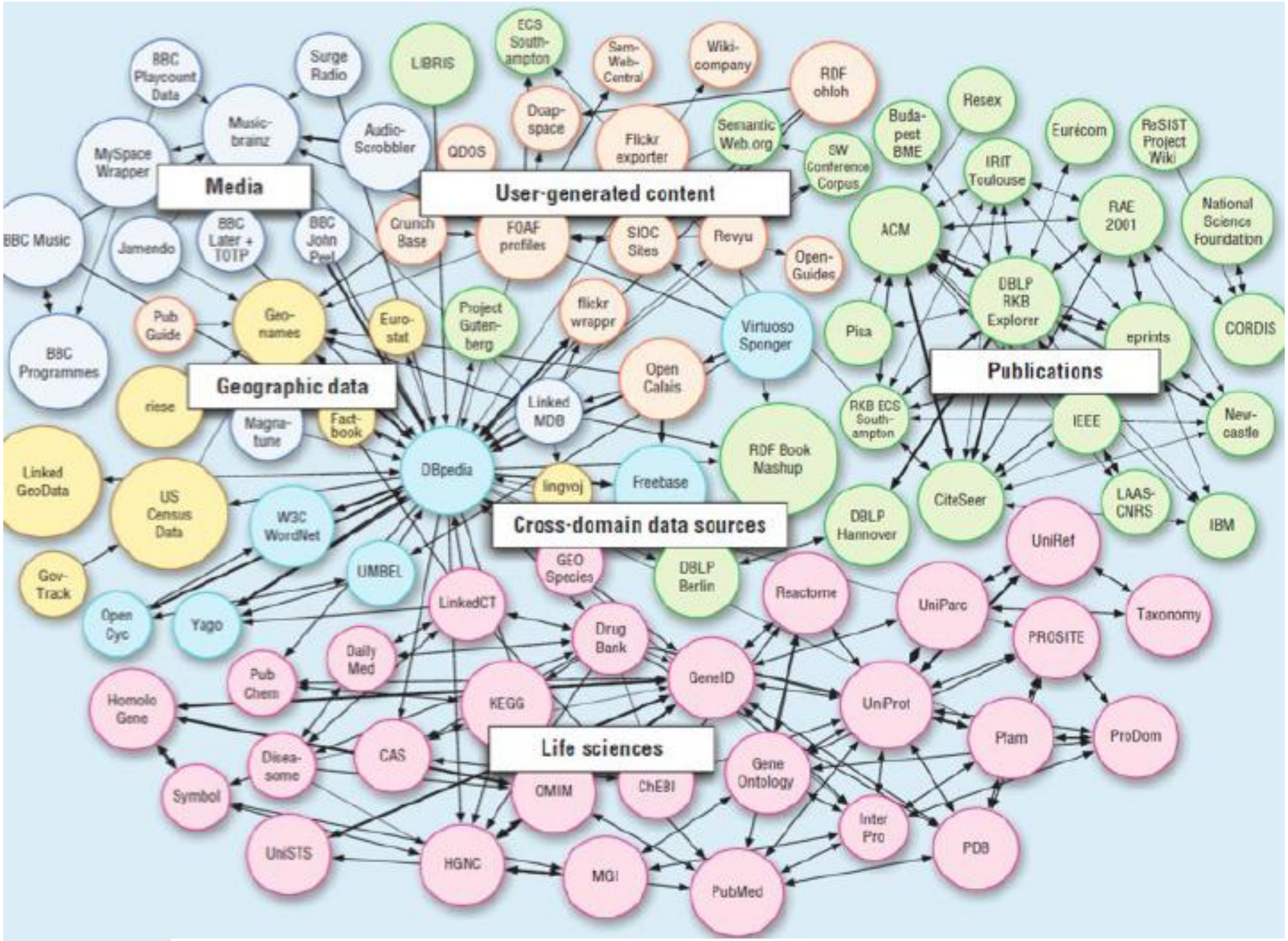
sameTitle(x,y) ∧ sameAuths(x,y) ∧ sameVenue(x,y) ⇒ sameAs(x,y)
sameTitle(x,y) ∧ sameAuths(x,y) ∧ sameAffil(x,y) ⇒ sameAs(x,y)
overlapAuths(x,y) ∧ sameAffil(x,y) ⇒ sameAuths(x,y)
sameAs(rec1.auth1, rec2.auth1) [0.2]
sameAs(rec1.auth1, rec2.auth2) [0.9]
…

- specify in Markov Logic or as factor graph
- generate MRF (or …) and solve by MCMC (or …)

(Singla/Domingos: ICDM'06,
Hall/Sutton/McCallum:KDD'08)

Halbert L. Dunn: Record Linkage. American Journal of Public Health. 1946
H.B. Newcombe et al.: Automatic Linkage of Vital Records. Science, 1959.

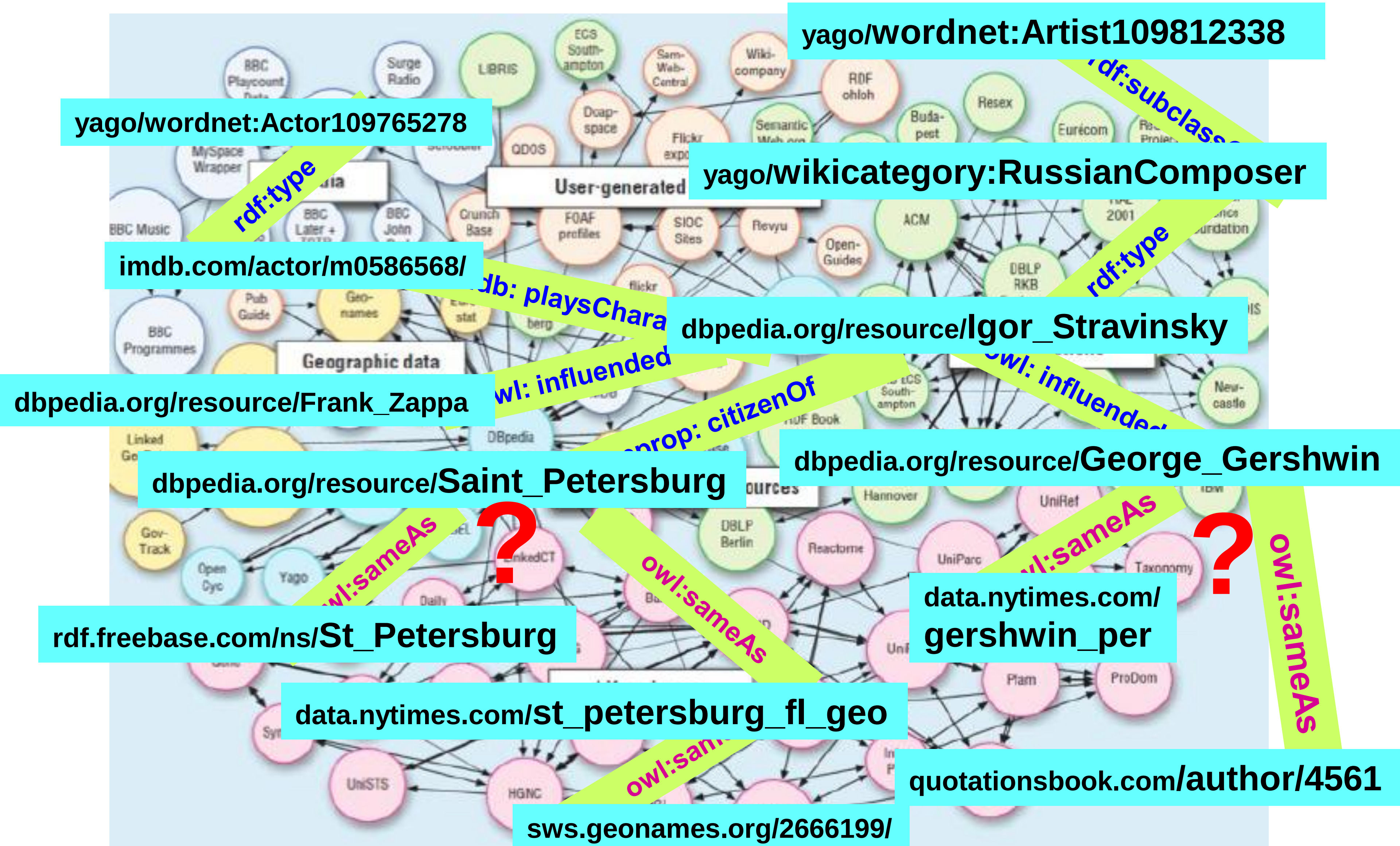
Linked Data: Record Linkage at Web Scale



Source: Christian Bizer, Tom Heath, Tim Berners-Lee, Michael Hausenblas, WWW 2010 Workshop on Linked Data on the Web

linkeddata.org

Linked Data: Record Linkage at Web Scale



need referential data quality for Linked Data: automatic & dynamic !

Open Problems

- More efficient graph algorithms (multicore, etc.)
- Allow mentions of unknown entities, mapped to null
- Short and difficult texts:
  - tweets, headlines, etc.
  - fictional texts: novels, song lyrics, etc.
  - incoherent texts
- Disambiguation beyond entity names:
  - coreferences: pronouns, paraphrases, etc.
  - common nouns, verbal phrases (general WSD)

Outline

✓ **Motivation**

✓ **Searching for Entities & Relations**

✓ **Informative Ranking**

✓ **Entity-Name Disambiguation**

★ **Wrap-up**

KB Applications: Achievements & Challenges

Search for Entities and Relations

good success story on entities, problems left:

- coverage beyond celebrities (long tail)
- complex queries → Sparql awareness & extensions
- reconcile Web with Web of Data

Ranking

good progress on ER language models, challenges left:

- better understanding of time & space
- mapping names & phrases to entities & relations
- efficiency & scalability

Entity-Name Disambiguation

good progress, challenges left:

- entities in the long tail, newly emerging entities
- robustness on short & difficult inputs
- apply at Web scale: tables, lists, text+RDFa, LOD, etc.

UI: Structured Keyword Search [Ilyas et al. Sigmod'10]

Need to map (groups of) keywords onto entities & relationships based on name-entity similarities/probabilities

q: Champions League finals with Real Madrid

UEFA Champions League

League of Wrestling Champions

final match

final exam

Real Madrid C.F.

q: German football clubs that won (a match) against Real

soccer

American football

social club

night club

sports team

Real Madrid C.F.

Real Zaragoza

Brazilian Real

Familia Real

Real Number

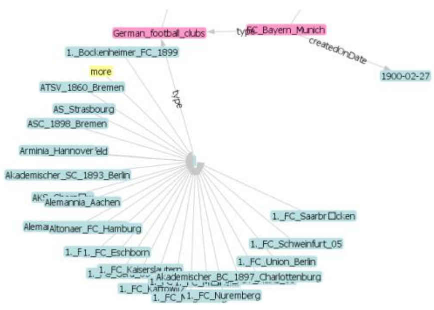
more ambiguity, more sophisticated relation
→ more candidates → combinatorial complexity

UI: Natural Language Questions

translate question into Sparql query:

- **dependency parsing to decompose question**
- **mapping of question units onto entities, classes, relations**

Which German soccer clubs have won against Real Madrid ?



German soccer teams that won against Real Madrid — 5 items

Item Name	Image	Description	Date Of Birth	Place Of Birth	Club
Cristiano Ronaldo		**Cristiano Ronaldo** dos Santos Aveiro, OIH, (Portuguese pronunciation: [kriʃˈtiɐnu ʁuˈnaɫdu]; born 5 February 1985), commonly known as **Cristiano Ronaldo**, ...	5 February, 1985	Funchal, Madeira, Portugal	Real Madrid
Lionel Messi		**Lionel** Andrés **Messi** (Spanish pronunciation: [ljoˈnel anˈdres ˈmesi]; born 24 June 1987) is an Argentine footballer who currently plays for Barcelona and the ...	24 June 1987	Rosario, Argentina	Barcelona
Steven Gerrard		**Steven** George **Gerrard**, MBE (pronounced /ˈdʒɛrərd/; born 30 May 1980), is an English footballer who plays for English Premier League club Liverpool and the ...	30/05/1980	Whiston, England	Liverpool
Frank Lampard		**Frank** James **Lampard** (born 20 June 1978) is an English footballer who currently plays as an attacking midfielder for Chelsea and at international level for ...	June 20, 1978	Romford, England	Chelsea
Juninho		This is a Portuguese name; the family name is Ribeiro Reis. António Augusto Ribeiro Reis Jr. (born 30 January 1975 in Recife), commonly known as **Juninho** or ...	January 30th 1975	Brazil	Kawasaki Fro

map results into tabular or visual presentation or speech

UI: Natural Language Questions

translate question into Sparql query:

- **dependency parsing to decompose question**
- **mapping of question units onto entities, classes, relations**

?c German soccer clubs ?r

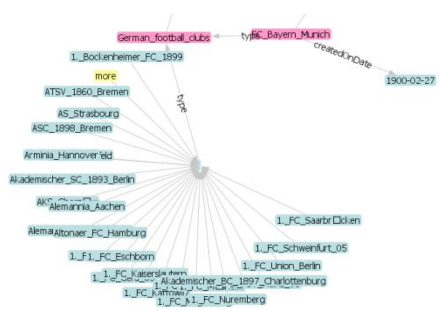
?o Real Madrid ?

Which ?c

?r have won against ?o

Select ?c Where { ?c isa GermanSoccerClub . c? wonAgainst RealMadrid . }

?id: ?c playedAgainst RealMadrid .
?c isWinnerOf ?id .}



map results into tabular or visual presentation or speech

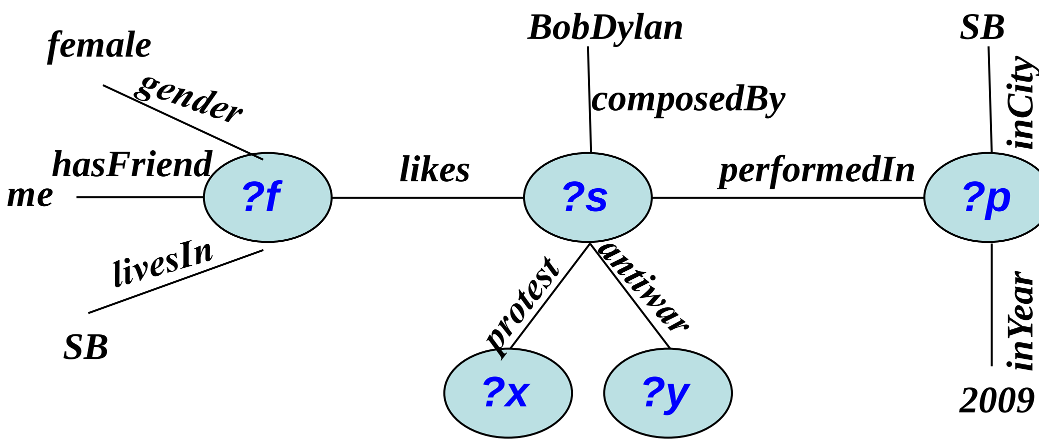
Performance/Scalability: Open Issues

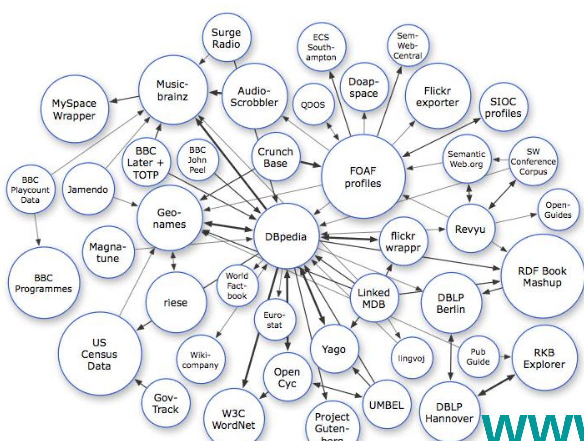
- Many joins between triples and inverted lists
- Query relaxation very expensive
- Entity disambiguation within QP
- Top-k processing with early termination
- Everything distributed (LOD cloud)

Efficient& Scalable RDF Query Processing

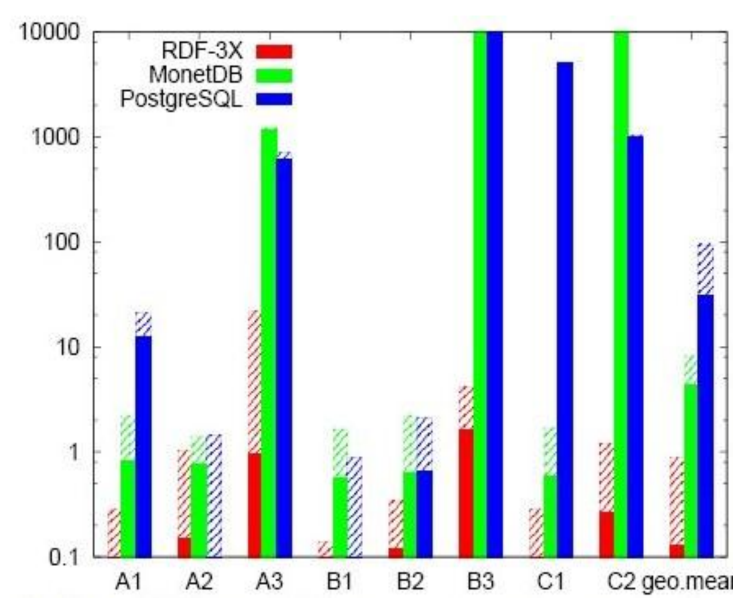
RDF-3X engine [T. Neumann et al.: VLDB'08, SIGMOD'09, VLDBJ'10]:

- no-tuning RISC, versioning, online updates, transactions
- aggressive **indexing** (all SPO permutations & projections)
- fast DP-based **join-order optimization** for 20-30 joins
- aggressive **sideways information passing**





As of September 2008



www.mpi-inf.mpg.de/~neumann/rdf3x/

Overall Take-Home

Web of Data

→ RDF: entities, relationships, structure, no schema

Querying

→ extended SPARQL:
W3C, tripleX patterns, schema-free joins

Ranking

→ Language Models: from docs to triples,
composable, relaxable, temporal, individual

Disambiguation

→ coherence graph: powerful model, harness KB
and statistics, robustness & efficiency challenge

- **Structure, No Schema**
- **SPARQL Extensions**
- **Language Models**
- **Entity Coherence**

Recommended Readings: Search & Ranking

- G. Kasneci, F. Suchanek, G. Ifrim: NAGA: Searching and Ranking Knowledge. ICDE 2008
- S. Elbassuoni, M. Ramanath, G. Weikum: Query Relaxation for Entity-Relationship Search. ESWC 2011
- S. Elbassuoni, M. Ramanath, et al.: Language-model-based ranking for queries on RDF-graphs. CIKM 2009
- Z. Nie, Y. Ma, S. Shi, J.-R. Wen, W.-Y. Ma: Web Object Retrieval. WWW 2007
- H. Bast, A. Chitea, F. Suchanek, I. Weber: ESTER: efficient search on text, entities, and relations. SIGIR 2007
- J. Pound, P. Mika, H. Zaragoza: Ad-hoc object retrieval in the web of data. WWW 2010
- J. Pound, I.F. Ilyas, G.E. Weddell: Expressive and flexible access to web-extracted data: a keyword-based structured query language. SIGMOD 2010
- O. Udrea, D. Recupero, V.S. Subrahmanian: Annotated RDF. ACM Trans. Comput. Log. 11(2), 2010
- V. Hristidis, H. Hwang, Y. Papakonstantinou: Authority-based keyword search in databases. TODS 33(1), 2008
- T. Cheng, X. Yan, K. Chang: EntityRank: Searching Entities Directly and Holistically. VLDB 2007
- D. Vallet, H. Zaragoza: Inferring the Most Important Types of a Query: a Semantic Approach. SIGIR 2008
- R. Blanco, H. Zaragoza: Finding support sentences for entities. SIGIR 2010
- P. Serdyukov, D. Hiemstra: Modeling Documents as Mixtures of Persons for Expert Finding. ECIR 2008
- R. Kaptein, P. Serdyukov, A.P. de Vries, J. Kamps: Entity ranking using Wikipedia as a pivot. CIKM 2010
- H. Fang, C. Zhai: Probabilistic Models for Expert Finding. ECIR 2007
- D. Petkova, W.B. Croft: Hierarchical Language Models for Expert Finding in Enterprise Corpora. ICTAI 2006
- M. Pasca: Towards Temporal Web Search. SAC 2008
- K. Berberich, S.J. Bedathur, O. Alonso, G. Weikum: A Language Modeling Approach for Temporal Information Needs. ECIR 2010: 13-25
- C. Zhai: Statistical Language Models for Information Retrieval. Morgan&Claypool, 2008
- B.C. Ooi (Ed.): Special Issue on Keyword Search, IEEE Data Eng. Bull. 33(1), 2010
- T. Neumann, G. Weikum: The RDF-3X engine for scalable management of RDF data. VLDB J. 19(1), 2010
- S. Ceri, M. Brambilla (Eds.): Search Computing: Challenges and Directions, Springer, 2010

Recommended Readings: Disambiguation

- J. Hoffart, M. A. Yosef, I. Bordino, et al.: Robust Disambiguation of Named Entities in Text. EMNLP 2011
- R.C. Bunescu, M. Pasca: Using Encyclopedic Knowledge for Named entity Disambiguation. EACL 2006
- S. Cucerzan: Large-Scale Named Entity Disambiguation Based on Wikipedia Data. EMNLP 2007
- D.N. Milne, I.H. Witten: Learning to link with wikipedia. CIKM 2008
- S. Kulkarni, A. Singh, G. Ramakrishnan, S. Chakrabarti: Collective annotation of Wikipedia entities in web text. KDD 2009
- G. Limaye, S. Sarawagi, S. Chakrabarti: Annotating and Searching Web Tables Using Entities, Types and Relationships. PVLDB 3(1), 2010
- S. Rüd, M. Ciaramita, J. Müller, H. Schütze: Piggyback: Using Search Engines for Robust Cross-Domain Named Entity Recognition. ACL 2011
- S. Singh, A. Subramanya, F.C.N. Pereira, A. McCallum: Large-Scale Cross-Document Coreference Using Distributed Inference and Hierarchical Models. ACL 2011
- L. Ratinov, D. Roth, D. Downey, M. Anderson: Local and Global Algorithms for Disambiguation to Wikipedia. ACL 2011
- R. Navigli: Word sense disambiguation: A survey. ACM Comput. Surv. 41(2), 2009
- T. Heath, C. Bizer: Linked Data: Evolving the Web into a Global Data Space. Morgan&Claypool, 2011
- F. Naumann, M. Herschel: An Introduction to Duplicate Detection. Morgan&Claypool, 2010
- H. Köpcke, A. Thor, E. Rahm: Learning-Based Approaches for Matching Web Data Entities. IEEE Internet Computing 14(4): 23-31 (2010)

Thank You!

Thank You!

living knowledge

SEVENTH FRAMEWORK PROGRAMME

M2CI CLUSTER OF EXCELLENCE

Google

DFG Deutsche Forschungsgemeinschaft

mpi max planck institut informatik