max planck institut
informatik

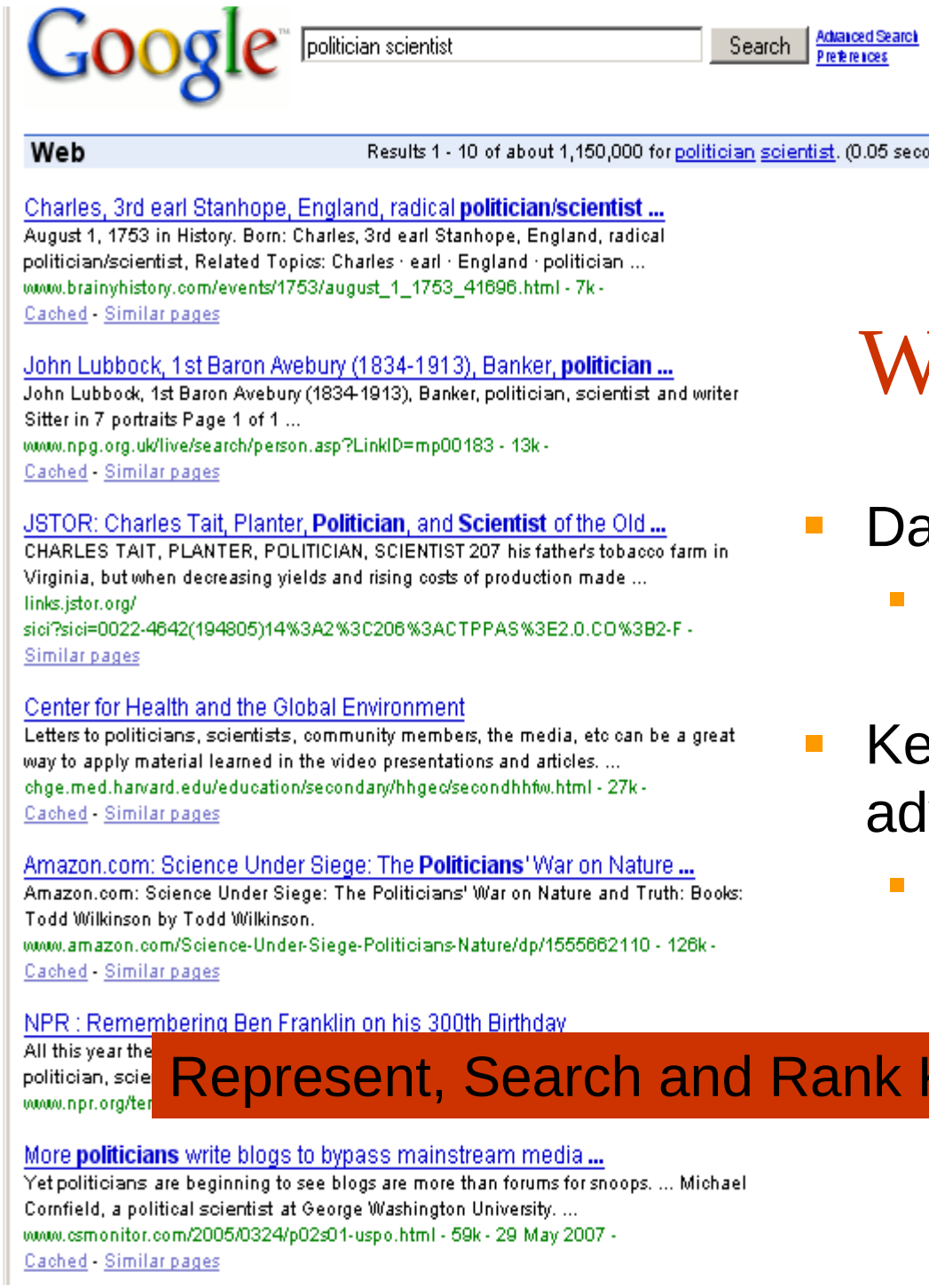
Language-Model based Ranking in Entity-Relationship Graphs

Shady Elbassuoni, Maya Ramanath, Gerhard Weikum
Max-Planck Institute for Informatics
Saarbrücken, Germany

max planck institut informatik

Motivation

- Politicians who are also scientists ?
- Academy Awarded movie directed by an Australian director?
- Fiction books written by a Nobel prize winner?
- Author of two papers in WWW 2008?



max planck institut informatik

What is lacking ?

- Data is not knowledge
  - Extraction and Organization
- Keywords cannot express advanced user intentions
  - Expression of concepts, relationships, etc.

Represent, Search and Rank Knowledge on the Web

max planck institut informatik

Knowledge Representation, Searching and Ranking

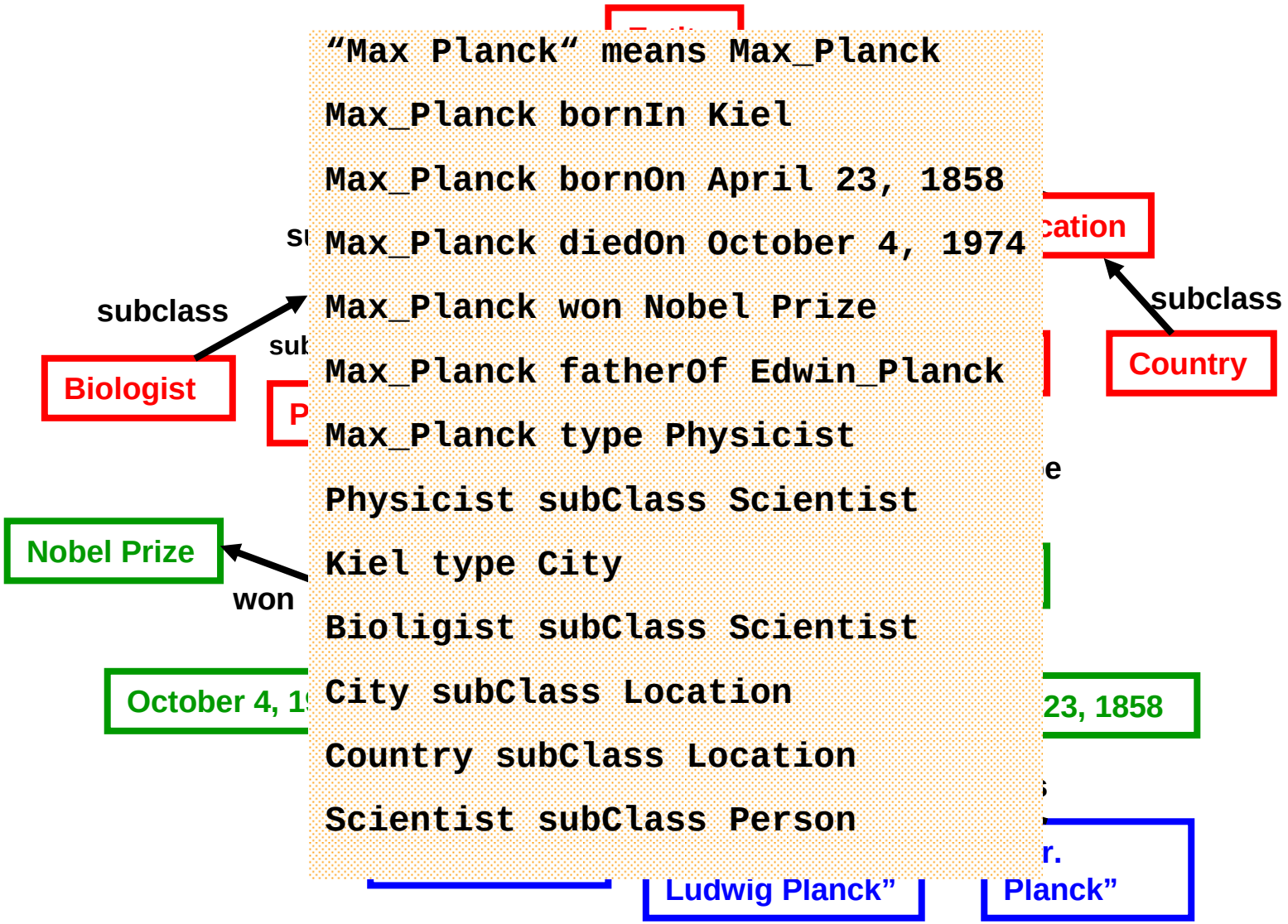
- Knowledge Representation
  - Entity-Relationship (ER) Graphs
    - Graphs of RDF-triples derived from web-sources
    - YAGO, FreeBase, DBPedia, DBLife, Libra, etc…
- Knowledge Searching
  - Expressive yet simple means of querying
    - Structured queries (as close to keywords as possible)
- Knowledge Ranking
  - IR-style ranking for structured queries over ER graphs

max planck institut informatik

Outline

- Motivation
- Knowledge Representation
  - Entity-Relationship Graphs
- Knowledge Searching
  - Query Language
  - Result Matching
- Knowledge Ranking
  - Language-Model based ranking for structured queries
- Preliminary Evaluation

Entity-Relationship Graphs



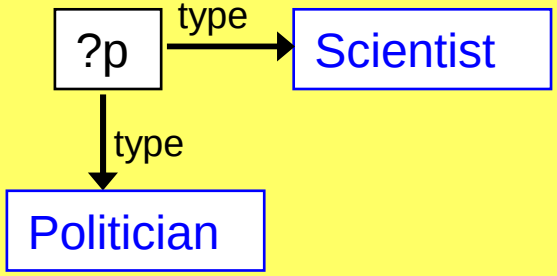
max planck institut informatik

Outline

- Motivation
- Knowledge Representation
  - Entity-Relationship Graphs
- Knowledge Searching
  - Query Language
  - Result Matching
- Knowledge Ranking
  - Language-Model based ranking for structured queries
- Preliminary Evaluation

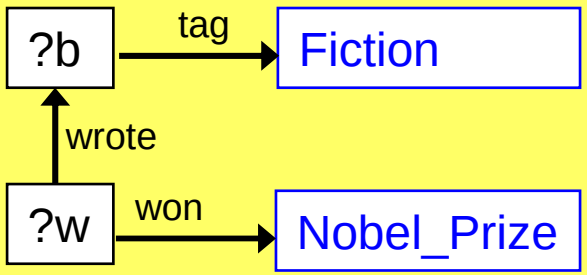
Query Language

- Politicians who are also scientists?



?p type Scientist; ?p type Politician

- Fiction books written by a Nobel prize winner?



?w wrote ?b; ?w won Nobel_Prize; ?b tag Fiction

max planck institut informatik

Result Matching

- Based on graph matching
  - Results are tuples that match the query triple-templates

```
?p type Scientist; ?p type Politician
```

```
Benjamin_Franklin type Scientist; Benjamin_Franklin type Politician

Alan_Greenspan type Scientist; Alan_Greenspan type Politician

Jan_Smuts type Scientist; Jan_Smuts type Politician

Paul_Wolfowitz type Scientist; Paul_Wolfowitz type Politician
```

Ranking is required

max planck institut informatik

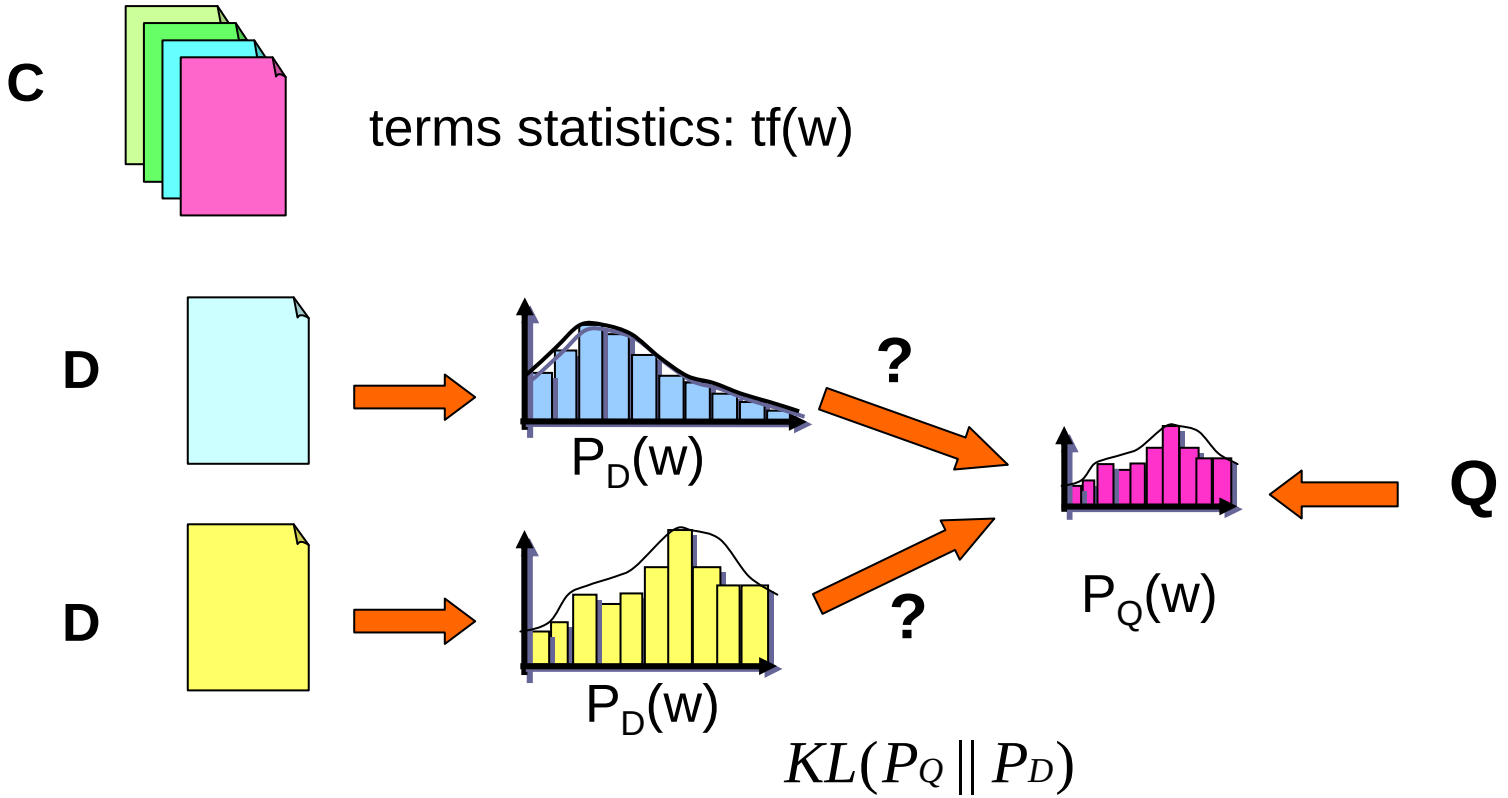
Outline

- Motivation
- Knowledge Representation
  - Entity-Relationship Graphs
- Knowledge Searching
  - Query Language
  - Result Matching
- Knowledge Ranking
  - Language-Model based ranking for structured queries
- Preliminary Evaluation

Language Model Ranking

- Results, Query: probability distributions over some common vocabulary
  - Construct Query Language Model P_Q
  - Construct Result Language Model P_D
- Measure how close the distributions are
  - “Closeness” gives measure of “relevance”
  - Rank results based on KL-Divergence between P_Q and P_D

Language Models in Textual Retrieval



max planck institut informatik

Language Models in ER Graphs

```
t1: Benjamin_Franklin type Scientist
t2: Benjamin_Franklin type Politician
t3: Paul_Wolfowitz type Scientist
t4: Paul_Wolfowitz type Politician
t5: Alan_Greenspan type Scientist
t6: Alan_Greenspan type Politician
t7: Jan_Smuts type Scientist
t8: Jan_Smuts type Politician
t9: Alan_Greenspan citizen American
t10: Benjamin_Franklin citizen American
```

200
250
300
150
60
100
40
50
150
400

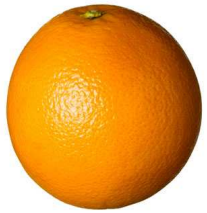
```
T1: Benjamin_Franklin type Scientist;
    Benjamin_Franklin type Politician

T2: Alan_Greenspan type Scientist;
    Alan_Greenspan type Politician
```

Q

```
?p type Scientist;
?p type Politician
```

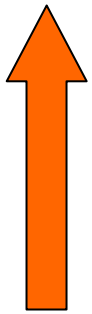
max planck institut informatik

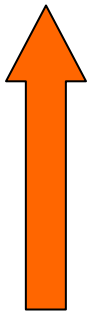
Terms in ER Graphs



Triples

Triple templates





```
T1: Benjamin_Franklin type Scientist;
    Benjamin_Franklin type Politician

T2: Alan_Greenspan type Scientist;
    Alan_Greenspan type Politician
```

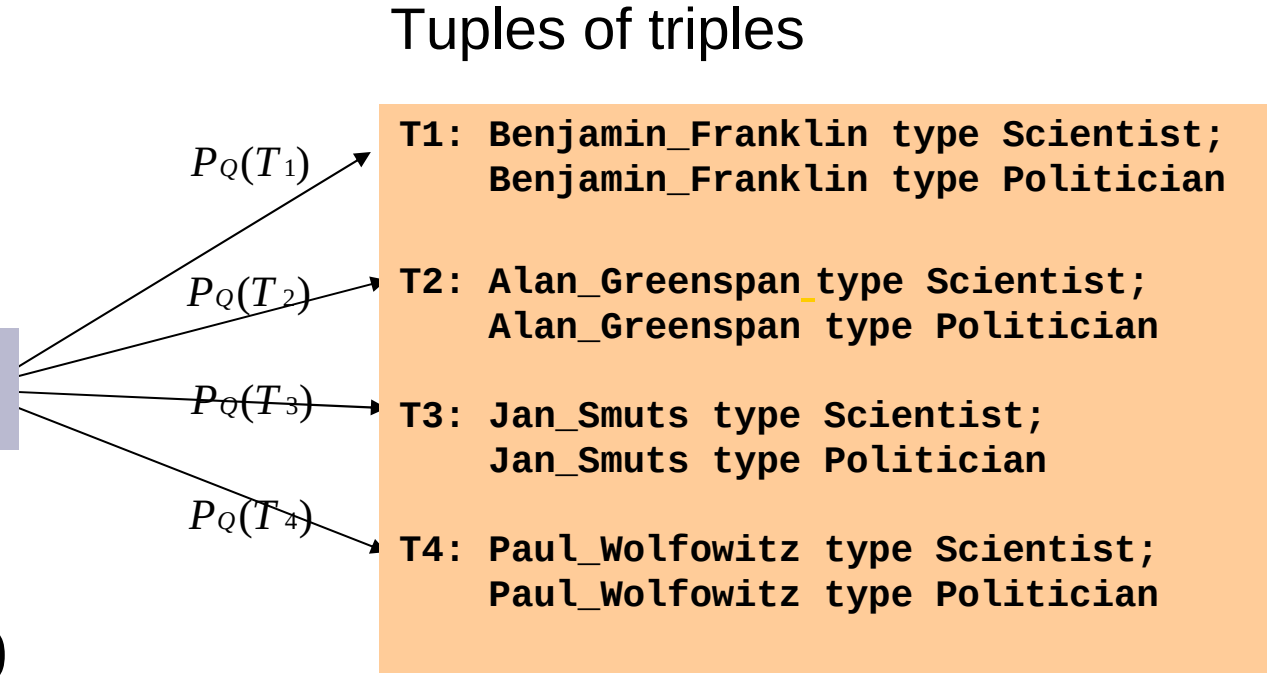
Q

```
?p type Scientist;
?p type Politician
```

Query Language Model

Q

```
?p type Scientist;
?p type Politician
```



$$P_Q(T) = \prod_{i=1}^{k} P_Q(t_i)$$

P_Q(Benjamin_Franklin type Scientist; Benjamin_Franklin type Politician)

=

P_Q(Benjamin_Franklin type Scientist) . P_Q(Benjamin_Franklin type Politician)

Estimating Query LM

Q

```
?p type Scientist;
?p type Politician
```

$$P_Q(t_i) = \frac{c(t_i)}{\sum_j c(t_j)}$$

t_i	$C(t_i)$	$P_Q(t_i)$
Benjamin_Franklin type Scientist	200	200/600
Alan_Greenspan type Scientist	60	60/600
Jan_Smuts type Scientist	40	40/600
Paul_Wolfowitz type Scientist	300	300/600
	600	

max planck institut informatik

Result Language Model

Results
(Tuples of triples)

```
T1: Benjamin_Franklin type Scientist;
    Benjamin_Franklin type Politician

T2: Alan_Greenspan type Scientist;
    Alan_Greenspan type Politician
```

$$P_D(T) = \alpha P(T \mid D) + (1-\alpha) P(T \mid C)$$

$$P(T \mid C) = \prod_{i=1}^{k} P(t_i \mid C)$$

Estimating Result LM

$$P(t_i \mid C) = \frac{c(t_i)}{\sum_j c(t_j \mid C)}$$

	$C(t_i)$	$P(t_i \mid C)$
t1: Benjamin_Franklin type Scientist	200	200/1700
t2: Benjamin_Franklin type Politician	250	250/1700
t3: Paul_Wolfowitz type Scientist	300	300/1700
t4: Paul_Wolfowitz type Politician	150	150/1700
t5: Alan_Greenspan type Scientist	60	60/1700
t6: Alan_Greenspan type Politician	100	100/1700
t7: Jan_Smuts type Scientist	40	40/1700
t8: Jan_Smuts type Politician	50	50/1700
t9: Alan_Greenspan citizen American	150	150/1700
t10: Benjamin_Franklin citizen American	400	400/1700
	1700	

max planck institut informatik

Example Results

IMDB Queries	|Q|	Top result
Movies with genre comedy and two actors who acted in it.	3	Sideways, Paul Giamatti, Sandra Oh
Actor who acted in a family movie in 1995 and who also acts in comedies.	5	Tom Hanks, Toy Story, You've Got Mail

LibraryThing Queries	|Q|	Top result
A fiction book with tag Magic and its author	3	Stephanie Meyer, New Moon
Two books written by the same novelist	3	Cormac McCarthy, The Road, No Country for Old Men

max planck institut informatik

Conclusions

- Knowledge Representation, Searching and Ranking
- Language Model based Ranking in ER graphs
  - Estimate Query and Result LMs
  - Relevance determined by KL-divergence between the two estimated models
- Evaluation
  - Preliminary, but promising

max planck institut
informatik

THINK KEYWORDS, THINK AGAIN.