But What Do We Actually Know?

Simon Razniewski[1], Fabian Suchanek[2] and Werner Nutt[1]

[1]Free University of Bozen-Bolzano, Italy [2]Télécom ParisTech University, France

Freie Universität Bozen
Libera Università di Bolzano
Free University of Bozen - Bolzano

TELECOM ParisTech

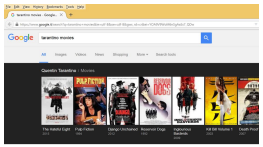
1. Overview

Knowledge bases usually have a high *precision*, while in contrast, little is known about their *recall* on specific topics. We envision that KBs become annotated with information about their recall on specific topics. We show what such annotations could look like and discuss techniques to obtain them.

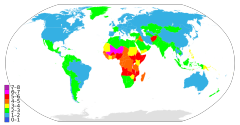
2. State of the art: KB recall is highly variant

167 out of 199 Nobel laureates in Physics are in DBpedia ☺

2 out of 2 children of Obama are in Wikidata ☺



36 out of 48 movies by Tarantino are shown in the Google Knowledge Graph ☺



According to YAGO, the average number of children per person is 0.02 ☹

The Google Knowledge Graph contains a predicate called ``Points of Interest''. It is not even clear how to measure its recall ☹

DBpedia contains currently only 6 out of 35 Dijkstra Prize winners ☹

3. Why unknown recall is a problem

Querying

```
SELECT ((COUNT(?child)) DIV
(COUNT(?person)) as ?childrenperperson)
WHERE {
        {?x dbr:children ?child}
        UNION
        {?person rdf:type dbo:Person}
```

```
Result: 0.02
```

→ United Nations might stop funding programs against overpopulation

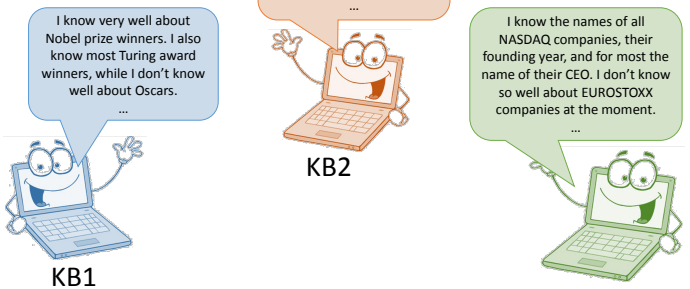
Distant supervision

"X buys diapers for Y"
→ hasChild(X, Y)?
→ Evaluation without negative facts is difficult
(same applies to **rule mining**)

KB debugging

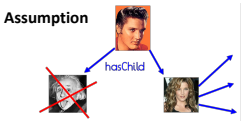
Recall information would make it possible to warn if someone adds another 5 children for Obama in Wikidata

Vision:
KBs should know what they know



4. How to obtain knowledge about recall?

A. Partial Completeness Assumption

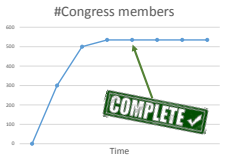


Assumption:
If we know r(x,y1),..., r(x,yn),
then all other r(x,z) are false.

B. Pattern Matching

Pattern: "X has two children"
Corpus: "Obama has two children"
KB: contains two children for Obama
→ complete(hasChild, Obama)

C. Growth Patterns over Time



D. Rule mining

$$\text{Complete(hasParent, X)} \rightarrow \text{Complete(hasChild, X)}$$

E. Popularity of Entities

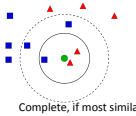
"Obama" About 170,000,000 results (0.76 seconds) → Maybe complete

"John Wilson Smith" 6 results (0.52 seconds) → Maybe incomplete

F. Class Membership

Complete, if most entities in the same class are complete

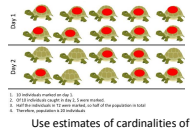
G. Similar Entities



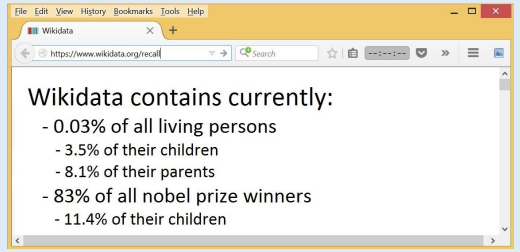
Complete, if most similar entities are complete

H. Crowdsourcing

I. Mark and recapture



Use estimates of cardinalities of subjects to predict recall



The ultimate vision: A comprehensive understanding of what KBs know.

Further reading

- Evaluation of the partial completeness assumption: L. Galárraga, C. Teflioudi, K. Hose, and F. M. Suchanek. AMIE: Association Rule Mining under Incomplete Evidence in Ontological Knowledge Bases *WWW*, 2013
- L. Galárraga, C. Teflioudi, K. Hose, and F. Suchanek. Fast Rule Mining in Ontological Knowledge Bases with AMIE+. *VLDB Journal*, 2015
- S. Razniewski, F. Korn, W. Nutt, and D. Srivastava. Identifying the extent of completeness of query answers over partially complete databases. *SIGMOD* 2015
- A tool for crowdsourced completeness annotations for Wikidata: http://cool-wd.inf.unibz.it/
- A crowdsourced completeness evaluation of Wikidata and YAGO: http://luisgalarraga.de/completeness-in-kbs

Acknowledgment

This work has been partially supported by the projects ``TQTK - The Quest to Know'', funded by the Free University of Bozen-Bolzano, and ``MAGIC'', funded by the Province of Bozen-Bolzano.