How to know how much we know

Make knowledge bases completeness-aware

Luis Galárraga

September 10, 2017

Lab: INRIA Rennes Bretagne Atlantique
Keywords: Completeness, Knowledge Bases, Semantic Web

1 Motivation

Current knowledge bases (KBs) are highly incomplete. For example, as of 2015 the Wikidata knowledge base [4] knows the father of only 2% of the people in its records, even though everybody has a father in real life. As a second example, the DBpedia knowledge base [2] knows 160 out of the actual 199 Nobel laurates in Physics. The problem of incompleteness is aggravated by the open world assumption (OWA) made by KBs. The OWA states that if a piece of information is not explicitly asserted as true, it is not necessarily false. Under the OWA, for example, if a KB knew only the airport connections Amsterdam and London for the Rennes{Saint-Jacques Airport, the connections Rennes-Madrid and Rennes-Vienna cannot be deemed as false, but only as unknown (in fact Rennes-Madrid is provided by Iberia, whereas Rennes-Vienna does not exist). This incompleteness is a serious problem for both users and producers of semantic data. For users it means that queries on KBs cannot o er any completeness guarantee. If a user wants to generate a report with a rank of the most connected airports in Europe, she will not have any guarantees of the accuracy and completeness of the reported results. This happens because the knowledge base does not know how much it knows, i.e., it may miss some connections and be unaware of it. On the other hand, producers of semantic data cannot know which parts of the knowledge base should be populated with more information, since knowledge bases cannot distinguish between false and unknown information.

2 Topic

Since it is impossible to collect all the true information about the real-world, a natural solution is to annotate knowledge bases with *completeness statements*. These are

assertions about the completeness of parts of the KB. There have been some e orts to annotate KBs with completeness information [1], however all these works su er from a major limitation: they de ne completeness assertions for simple queries, e.g., lists with simple de nitions such as the \list of Nobel laurates". Still, a KB may be incomplete for this list and still complete for the (more complex to describe) list of Nobel laurates in Physics. In contrast, if the knowledge base knows the complete list of laurates for each of the categories of the Nobel Prize, it follows that the list of laurates is complete. Unfortunately, no storage engine nowadays supports such a reasoning, thus it is not possible to provide completeness guarantees for arbitrary queries currently.

This internship aims to tackle the aforementioned problem by developing algorithms for completeness-aware reasoning that can provide completeness guarantees for arbitrary queries on knowledge bases. These algorithms should exploit the existing completeness statements on simple queries to reason about the completeness of more complex queries. For further information about this topic, please refer to [3].

References

[1] Luis Galarraga, Simon Razniewski, Antoine Amarilli, and Fabian Suchanek. Predicting Completeness in Knowledge Bases. In *International Conference on Web Search and Data Mining*, 2017.

[2] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. v. Kleef, S. Auer, and C. Bizer. DBpedia - A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 6, 2015.

[3] Simon Razniewski Luis Galarraga, Katja Hose. Enabling Completeness-aware Querying in SPARQL. In *International Workshop in Web and Databases*, 2017. Available at http://luisgalarraga.de/docs/enabling-completeness.pdf.

[4] D. Vrandecic and M. Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 2014.